

III. Auswertung

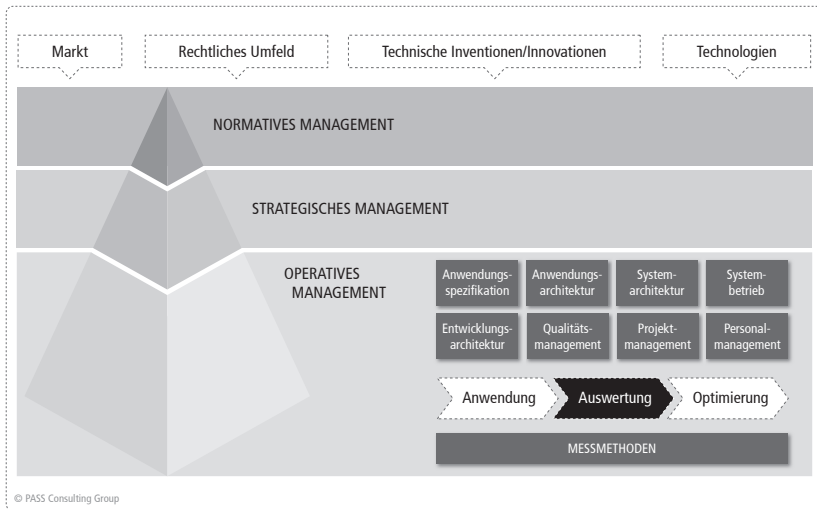


Abbildung 8: Auswertung der Kennzahlen und Ursachenanalyse

Dieses Kapitel beschreibt die Auswertung der im vorhergehenden Schritt erhobenen Kennzahlen und eine Analyse der Ursachen – als Voraussetzung für eine zielgerichtete Verbesserung.

Entwicklung der Produktivität im Zeitverlauf

Ein erster Schritt zur Auswertung der zuvor beschriebenen Kennzahlen ist die Analyse ihrer Veränderungen im zeitlichen Verlauf. Anhaltende Verschlechterungen sind meist ein Indikator für verborgene Chancen, um nach einer Analyse Verbesserungsmaßnahmen zu finden und in der Zukunft Zeit und Kosten sparen sowie die Qualität verbessern zu können. Der Verlauf der KPIs nach Durchführung einer Verbesserungsmaßnahme ist wiederum ein Indikator für deren Effektivität und Nachhaltigkeit.

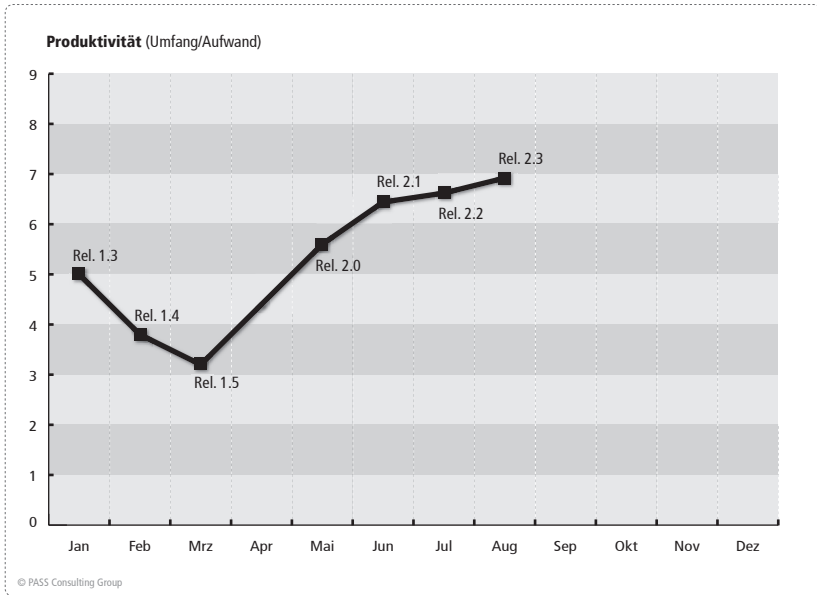


Abbildung 9: Entwicklung der Produktivität über mehrere Monate (Beispiel)

Abbildung 9 zeigt die grafische Darstellung der Entwicklungsproduktivität, die nach jedem fertiggestellten und in Produktion gesetzten Release eines beispielhaft betrachteten Systems gemessen wurde. Das Diagramm zeigt deutlich, dass seit Jahresbeginn die Produktivität zunächst immer geringer wird. In den Monaten April und Mai wurde längere Zeit an einem neuen Major Release gearbeitet, das im Mai als Release 2.0 in Produktion gesetzt wurde und in dem Verbesserungsmaßnahmen nachhaltig wirksam wurden.

Bei Messungen der Weiterentwicklungsproduktivität unterschiedlich großer Releases derselben Anwendung stellt man in der Praxis oft fest, dass die Messgenauigkeit scheinbar proportional zum Entwicklungsumfang ist: Je geringer der

Entwicklungsumfang, umso zweifelhafter erscheint die Genauigkeit einer einzelnen Messung. Die Ursache dafür ist die in der Praxis oft sehr unterschiedliche Komplexität der zu implementierenden Anforderungen, die von funktionsorientierten Messmethoden nicht vollständig berücksichtigt wird, da sich diese stärker an der Anzahl von – die Systemgrenzen überschreitenden – Datenelementen oder -strukturen orientieren. Unbestreitbar ist jedoch eine Korrelation zwischen der Komplexität und dem Entwicklungsaufwand.

Als Beispiel mag das neue Release einer Anwendung dienen, in dem nur die Ausgabe einiger Reports umgesetzt wurde. Die Reports bestehen aus einer großen Menge von Datenelementen, welche die Systemgrenzen überschreiten und somit in die Zählung von funktionsorientierten Messmethoden eingehen. Nehmen wir an, alle Berichtsdaten werden durch eine einfache Abfrage aus der Datenbank gelesen. Der Aufwand für dieses Release ist gering und der funktionale Umfang groß, wodurch eine hohe Produktivität errechnet wird. Als zweites Beispiel mag nun die Implementierung eines komplexen Algorithmus mit Ausgabe des Ergebnisses in einem einzigen Dialogfeld dienen. Funktionsorientierte Umfangsmetriken ergeben hier einen geringen Entwicklungsumfang, da für den Anwender nur ein Wert sichtbar ist. Der Aufwand zur Implementierung ist jedoch hoch. Rechnerisch ergibt dies eine geringe Produktivität.

Bei einem geringen Entwicklungsumfang besteht das Risiko, dass die wenigen Anforderungen nahezu vollständig eine gegenüber dem Durchschnitt entweder zu niedrige oder zu hohe Komplexität haben und so auch die für dieses Release gemessene Produktivität einseitig beeinflusst ist. Bei einem größeren Entwicklungsumfang nivellieren sich diese Unterschiede meist. In der Praxis hat es sich daher bewährt, die Weiterentwicklungsproduktivität eines Systems durch Zusammenfassung von Umfang und Aufwand aller Releases der letzten n Monate zu berechnen:

$$\bar{p} = \frac{\Sigma \text{Umfang}}{\Sigma \text{Aufwand}}$$

Verbesserungen oder Verschlechterungen der Produktivität sind erkennbar, wenn sie nachhaltig genug sind und sich in Relation zum Gesamtumfang auch entsprechend auswirken. Tabelle 2 zeigt dies beispielhaft anhand des Zeitverlaufs, der in Abbildung 9 grafisch dargestellt ist. Aufgeführt sind der Entwicklungsumfang U_n und der Aufwand A_n für jedes einzelne Release.

		U_n (DIP)	A_n (PT)	\bar{p} (DIP/PT)
Rel 1.0	Okt	125	23	
Rel 1.1	Nov	23	2	
Rel 1.2	Dez	540	123	
Rel 1.3	Jan	125	15	5,0
Rel 1.4	Feb	410	149	3,8
Rel 1.5	Mrz	90	77	3,2
Rel 2.0	Mai	1.210	79	5,6
Rel 2.1	Jun	435	115	6,4
Rel 2.2	Jul	140	75	6,6
Rel 2.3	Aug	995	132	6,9

Tabelle 2: Weiterentwicklungsproduktivität durch Zusammenfassung eines Zeitraums von 4 Monaten (Beispiel)

Die je Monat errechnete durchschnittliche Produktivität \bar{p} ergibt sich aus der Summe von Umfang und Aufwand aller Releases aus den letzten 4 Monaten. Für Release 1.3 errechnet sich die Produktivität somit wie folgt:

$$\bar{p} = \frac{125 + 23 + 540 + 125}{23 + 2 + 123 + 15} = 5$$

Dieser Verlauf erweist sich als robust gegenüber „Ausreißern“ wie Release 1.1, das bei einer Einzelbetrachtung eine Produktivität von 11,5 hätte (ergibt sich aus Umfang 23 DIP dividiert durch Aufwand 2 PT), oder Release 2.2 mit einer isoliert gemessenen Produktivität von 1,9 (ergibt sich aus Umfang 140 DIP dividiert durch Aufwand 75 PT). Erst durch die Zusammenfassung mehrerer Einzelmessungen wird die Nachhaltigkeit der Verbesserung deutlich, die sich ab Release 2.0 eingestellt hat. Durch eine Betrachtung der einzelnen Messungen ist dies nicht erkennbar.

Der in diesem Beispiel gewählte Zeitraum von 4 Monaten für die Zusammenfassung der Weiterentwicklungen reicht aus, um den beschriebenen Effekt der Nivellierung von Unschärfen bei geringem Entwicklungsumfang verdeutlichen zu können. Je nach Releasezyklen kann die Zusammenfassung größerer Zeiträume wie dem eines vollständigen Jahres ($n = 12$) sinnvoll sein.

Interne Benchmarks

Neben der Aussagekraft des Produktivitätsverlaufs bei einem einzelnen System kann auch die Gegenüberstellung verschiedener Systeme oder Organisationseinheiten von Nutzen sein, wie sie in Abbildung 10 beispielhaft dargestellt ist.

Grundlage für die Vergleichbarkeit der Messwerte verschiedener Systeme ist, dass sie unabhängig von technologischen Aspekten sind und sich nur an den Anwendungsfällen orientieren. Dies ist bei Verwendung funktionsorientierter Umfangsmetriken wie der Function Point-Analyse, der COSMIC-Methode oder der Data Interaction Point-Methode grundsätzlich gegeben. Von Vorteil ist die Gegenüber-

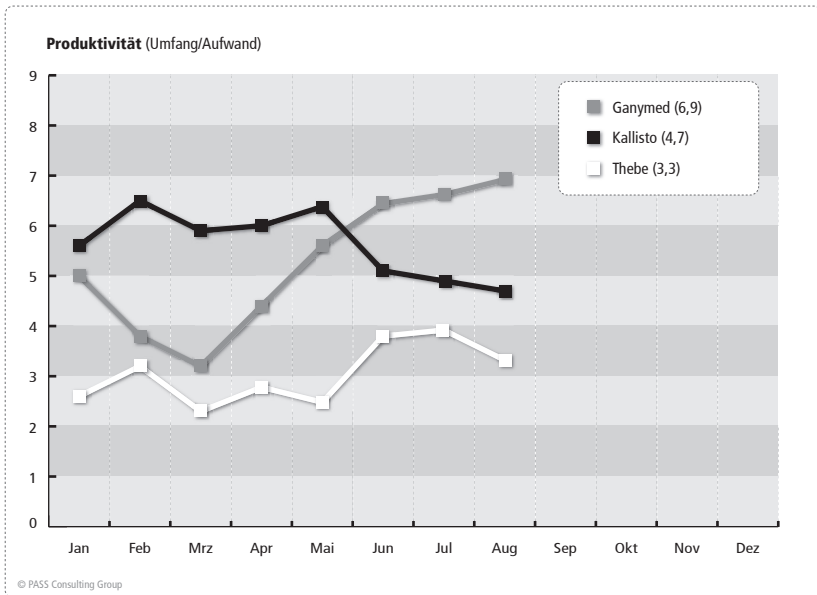


Abbildung 10: Interner Produktivitätsbenchmark (Beispiel)

stellung mehrerer Systeme oder IT-Shops der selben Organisation, was man auch als internen Benchmark bezeichnet, um zu erkennen, welches die produktivsten Teams bzw. Organisationseinheiten, die „Best in Class“, sind, von denen andere lernen können. Dies erfordert eine Analyse der Erfolgsfaktoren und eine Prüfung, ob bzw. wie diese auf andere Shops übertragbar sind. Versteht es eine Organisation, durch Transparenz und einen offenen Umgang mit validen Messungen ohne Schuldzuweisungen und Bestrafungen einen sportlichen Wettbewerb der Verbesserung zu fördern, ist das Ergebnis neben Kostenreduktionen auch eine durch den Markt wahrnehmbare bessere Planungssicherheit und Verlässlichkeit von Terminaussagen.