

(Erweiterte) Funktionalgrößen Meßmethoden sind auch in Verbesserungsprojekten anwendbar

Ton Dekkers (ins Dt. von Oliver Meyer)

Sogeti Nederland B.V. (Sogeti Deutschland GmbH)

ton.dekkers@sogeti.nl

Abstract:

Betrachtet man die Verteilung der IT Kosten, stellt man fest, dass der größte Teil des Budgets für Wartungs- und Verbesserungsprojekte zugewiesen wird. Neuentwicklungen umfassen 30-50% der IT Kosten. Methoden zur Messung des Funktionsumfangs werden meistens nur bei Neuentwicklungen angewendet. Mit kleinen Erweiterungen bekannter Meßmethoden wie der Function Point Analyse oder COSMIC Full Function Points kann fast jede IT Aktivität analysiert werden. Während der vergangenen 10 Jahre hat Sogeti die erweiterte Meßmethode basierend auf der Function Point Analyse sehr erfolgreich in zahlreichen Projekten angewendet.

Schlüsselbegriffe

COSMIC Full Function Points, Function Point Analyse, Wartungsprojekte

1 Abgrenzungen

Vor einer detaillierten Beschreibung ist es notwendig die Grenzen / Beschränkungen der Funktionalgrößen Meßmethode zu erläutern. Die wichtigste Einschränkung hängt von der Bestimmung der Größeneinteilung ab. Wenn das Ergebnis der Größeneinteilung zur Abschätzung genutzt wird, ist es notwendig, dass die Anzahl von Function Points (FP) oder COSMIC Functional Size Units (CFSU) über einem Minimalwert liegt. Für eine verlässliche Abschätzung eines (neuen) Entwicklungsprojektes sollten mehr als 200 FP oder 100 CFSU betrachtet werden. Ähnliches gilt für Verbesserungs- und Wartungsprojekte. Werden die erweiterten Methoden genutzt sollte ein Projekt mindestens 100 Maintenance Function Points (MFP) oder 60 Maintenance COSMIC Functional Size Units (MCFSU) haben.

In der Praxis bedeutet das, daß der Einsatz von erweiterte methoden (nur) in einer Organisation sinnvoll ist, in der mit Releases gearbeitet wird. Wenn jeder Änderungswunsch sofort umgesetzt wird, ist es besser (Experten) Einschätzungen einzuholen.

2 Anwendbarkeit

Die oben erwähnte Anwendung von (erweiterten) Funktionalumfangmessungen findet Anwendung in der Budgetierung. Eine Abschätzung ist im einfachsten Fall das Produkt aus den Performance Indikatoren Produktivität (Aufwand pro Einheit) und Kosten pro Einheit. Für das Release Management kann die Bestimmung des Funktionalumfangs entscheidend sein. Ist erst einmal über das Gesamtbudget entschieden worden, lässt sich der Umfang der realisierbaren Funktionalität leicht bestimmen. Anders gesagt wird einfach das Budget durch die beschriebenen Performance Indikatoren geteilt um den Funktionsumfang zu bestimmen. Ist der Umfang der benötigten Funktionalität (Verbesserungen) eines Releases bekannt, kann dieser mit der kalkulierten Größe verglichen werden. Falls notwendig können die Entscheider bestimmte Funktionalitäten anhand einer Priorisierung auswählen oder versuchen zusätzliches Budget aufzutreiben.

Schließlich darf das Messen der Performance nicht vergessen werden. Nicht nur die Messung der Performance der eigenen Organisation ist wichtig, auch die des Zulieferers. Diese ist insbesondere zu betrachten, wenn es um Wartung geht, da der Trend in diesem Bereich zum Outsourcing geht.

Es ist zu beachten, dass die Messung des Funktionalumfangs lediglich eine Hilfe für das managen eines (Verbesserungs-)Projektes ist. Die auf der FPA oder der CFFP basierenden Abschätzungen sollten kritisch betrachtet werden und immer im Zusammenhang mit den projektspezifischen Umständen beurteilt werden. Dies wird in dem Meßmethodenmodell (Abb. 1) dargestellt. Der Umfang der Software und die

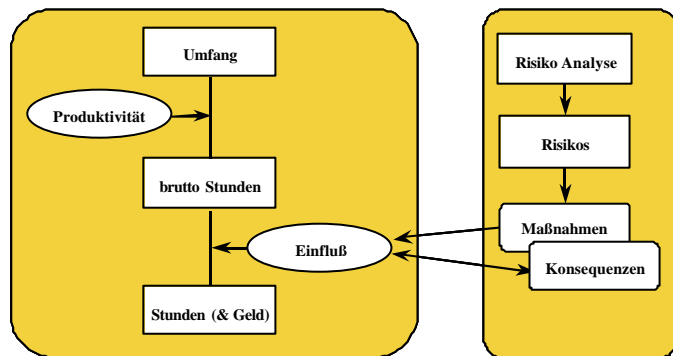


Abb. 1 - Meßmethodenmodell

Performance der Systementwicklung sind die entscheidenden Punkte wenn es um die Vorhersage von Arbeitsaufwand und Kosten pro Einheit für eine Applikation oder Projekt geht. Sind einmal Umfang und Entwicklungsmethoden bekannt, werden auch die Risiken sichtbar. Diese Risiken müssen analysiert werden ebenso der Einfluss und die Eintrittswahrscheinlichkeit und die damit verbundenen Kosten der Messung sind Kriterien die in der Messung umgesetzt werden müssen. Alle Komponenten fließen in das Projektbudget und die Planung mit ein. Es ist zu empfehlen die Abschätzungen anhand von organisationsinternen Performance Raten zu machen, sollten diese nicht gegeben sein kann auch auf Performance Raten Dritter zurückgegriffen werden. Sehr hilfreich sind die Projekt Übergabe Raten der International Software Benchmarking Standards Group (ISBSG) [3]. Diese gemeinnützige Organisation stellt Raten zur Verfügung die auf

einer Projektdatenbank mit über 2000 Projekten verschiedener Länder, Branchen und unterschiedlichen Entwicklungsumgebungen basieren.

3 Genereller Ablauf

Die Bestimmung des Funktionalumfangs ist eine objektive Methode um die Größe eines Informationssystems zu bestimmen. Der Umfang wird in Einheiten gemessen. Ähnliches gilt für die Abschätzung von Verbesserungsumfängen. Abhängig von den Änderungsnachfragen werden die beteiligten Funktionen identifiziert. Wird die FPA verwendet, korrespondieren die Funktionen mit ihren Elementarfunktionen, während bei Verwendung der CFFP die Funktionen als Funktionsprozesse beschrieben werden. Sind alle Funktionen identifiziert (und gemessen) ist die Größe des Releases bekannt. Mit Hilfe der Einfluss Analyse wird der Einfluss der RfC (Request for Change – Änderungsversuch) auf die Funktionen bestimmt und darauf basierend eine Verbesserungsart den Funktionen zugewiesen. Die Anzahl der Einheiten die einer Verbesserung zugewiesen werden, werden mit dem Faktor entsprechend dieser Art multipliziert. Wenn die Resultate pro Art addiert werden, ist die gewichtete Größe erreicht.

4 Wartung (Definitionen)

Es gibt zwei Ansichtspunkte bei der Wartung: Den technologischen Aspekt des Systems und den Anwenderwert des System. Das System muss die technologischen Bedingungen der Organisation erfüllen. Anwenderwert heißt, dass die Software die tägliche Arbeit der User sinnvoll unterstützt. Wartung (respektive Verbesserung) kann als Optimierung des technischen Status und des Anwenderwertes in kontrollierter Weise definiert werden.

In der Literatur findet man eine Unterscheidung der folgenden Kategorien:

- Korrigierende Wartung
- Produktpflege
- Anpassungen.

Es existieren verschiedene Definitionen für die Kategorien, so dass es notwendig ist, zunächst zu klären, was mit den Kategorien gemeint ist. Die am weitesten verbreitete Definition kommt von Lientz und Swanson [4]:

Korrigierende Wartung beinhaltet alle Aktivitäten, die zur Behebung der Softwarefehler angewendet werden. Darunter fallen nicht nur funktionale sondern auch Designfehler. Perfekte Wartung verbessert die Qualität der operativen Software. Das beinhaltet auch die Erweiterung existierender Funktionalitäten, basierend auf Requests for Change, vorgeschlagen und vereinbart vom Enduser und vom Auftraggeber. Anpassungen der Software beinhalten völlig neue Funktionalitäten, die sich aus neu definierten Anforderungen ergeben haben.

4.1 Abschätzung korrigierende Wartung

Fehler, die das gesamte System lahm legen oder das korrekte Arbeiten verhindern sind sofort zu beheben. Ein Abwarten einer Abschätzung auf dem formalen Entscheidungsweg ist normalerweise nicht gewünscht und nicht akzeptabel. Das Auftreten von Fehlern ist normalerweise unvorhergesehen und stört den normalen Ablauf der Aktivitäten. Sicherlich ist es wichtig, dass der Umfang der zu reparierenden Funktionalität abgeschätzt wird, wichtiger ist jedoch zunächst die eigentliche Fehlerbehebung. Durch Abschätzung von Korrekturmaßnahmen bekommt man einen guten Einblick in zwei fundamentale Performanceindikatoren: Qualität (Fehler pro Betriebssystem pro Einheit) und mittlere Reparaturdauer (Dauer pro Betriebssystem/ Fehlerumgebung). Sind diese Indikatoren nicht verfügbar, kann einfach die Dauer für die korrigierende Wartung angenommen werden. Ein Risikofaktor ist bestimmt durch den Quotienten vom Umfang des Betriebssystems und Dauer der korrigierenden Wartung. Ändert sich das Portfolio, ein neues System kommt hinzu oder existierende Systeme werden ersetzt, wird der benötigte Aufwand für korrigierende Maßnahmen neu berechnet. Nur wenn dieser Mehraufwand mit einbezogen wird kann ein realistischer Zeitplan und Abgabetermin bestimmt werden.

$$\text{Risikofaktor} = \text{Dauer} / \text{Größe des Betriebssystems}$$

$$\text{Aufwand} = (\text{neue}) \text{Größe des BS} * \text{Risikofaktor}$$

Durch die Bestimmung eines Risikofaktors für jedes BS kann die Kapazitätenplanung verfeinert werden. Es kann z.B. eine Unterscheidung zwischen neuen (jünger als ein Jahr in Betrieb), mittelalten und alten (länger als 5 Jahre in Betrieb) Systemen gemacht werden. Erfahrungsgemäß ist ein System weniger fehlerbehaftet, wenn es schon ein Jahr in Betrieb ist. Mit der Zeit beeinflussen Veränderungen die Struktur und Stabilität von älteren Systemen und erzeugen so ein steigende Anzahl von Fehlern. Hier jedoch ein Rat: Versuchen Sie nicht, diese Unterscheidung bei der ersten Abschätzung zu machen!

Diese "Methode" zur Abschätzung der korrigierenden Maßnahmen kann sowohl für die FPA als auch für die CFFP angewendet werden. Zur Abschätzung des Umfangs der reparierten Funktionalität, wird die Methode der Produktpflegemessung verwendet.

4.2 Abschätzung von Produktpflege und Anpassungen mit FPA

Schritt 1: Bestimmung des Releaseumfangs

Die durch Wartung beeinflusste Funktionalität muss zunächst in der so genannten Diagnosephase bestimmt werden. Basierend auf den BfC werden alle elementaren Funktionen (Transaktionsarten: EI - External Input / Externe Eingabe), EO - External Output / Externe Ausgabe, EQ - External Enquiry / Externe Abfrage) und

Logical Files (ILF - Internal Logical Files / Interne Geschäftsentität , EIF - External Interface Files / Externe Geschäftsentität) identifiziert, die von den Veränderungen in irgendeiner Art und Weise betroffen sind.

Ebenso muss der Dominoeffekt berücksichtigt werden: eine Veränderung eines ILF beeinflusst alle Elementarfunktionen, die diese nutzen (lesen, updaten). Des Weiteren sind alle Elementarfunktionen betroffen die untereinander Beziehungen haben (z.B. Schnittstellenübergreifende Funktionen). Der Releaseumfang setzt sich aus der Summe aller identifizierten Funktionen und Dateien zusammen. Hier wird der Umfang in Function Points wiedergegeben.

Schritt 2: Bestimmung der Verbesserungsarten

Grundsätzlich kann eine Abschätzung für eine Neuentwicklung durch Produkt aus Umfang der Applikation (in FP's) und Produktivitätsrate (Zeit pro FP) angegeben werden. Die Produktivitätsrate korreliert mit einem sich wiederholenden Prozess, die Aktivitäten werden so bestimmt, dass sie den beschlossenen Standards und Tools genügen. Leider ist die Wartung jedoch etwas Grundverschiedenes wie eine Neuentwicklung. Im Prinzip sind die Aktivitäten die gleichen, jedoch hängt die Ausführung der Aktivitäten von dem Einfluss den der RfC auf die identifizierte Funktionalität hat ab. Die Abschätzung muss den Einfluss der verschiedenen Arten der Verbesserung aufzeigen und berücksichtigen. Basierend auf Erfahrungswerte und gesundem Menschenverstand können folgende Arten der Verbesserung identifiziert werden:

- ändern der Funktionalität (niedriger, durchschnittlicher und hoher Komplexität);
- erweitern der Funktionalität (neu);
- entfernen der Funktionalität;
- testen der Funktionalität

Die Zuordnung der betroffenen Funktionalitäten zu den Verbesserungsarten wird während der Einflussanalysephase vorgenommen. Diese Zuordnung ist offensichtlich und bedarf keiner weiteren Erklärung. Bei den Änderungen unterscheidet man jedoch zwischen Änderungen mit niedriger, durchschnittlicher und hoher Komplexität, wobei die Differenzierung recht subjektiv ist.

Die hier beschriebene Methode erlaubt dem Anwender zu entscheiden wie die Klassifizierung vorgenommen werden soll. In den Niederlanden werden 3 Klassifizierungen genutzt:

- Rein basierend auf dem eigenen Verständnis über die Auswirkungen;
- Basierend auf eine Matrix, in der die Anzahl betroffener logischer Dateien und Datenelementen , die Klassifizierung bestimmen.
- Basierend auf eine Matrix, in der die relative Aufteilung betroffener logischer Dateien und Datenelementen, die Klassifizierung bestimmen.

Die beiden letzteren bieten eine gewisse Objektivität, so dass es ratsam ist eine davon auszuwählen. Sogeti hat 1992 [5] eine Matrix aufgestellt und verwendet diese immer noch für sich selbst und für Kunden. In einer Veröffentlichung von 1997 wurde von der NESMA eine Matrix mit relativen Klassifizierungen beschrieben, jedoch ergaben Untersuchungen von Sogeti bessere Ergebnisse mit dem eigenen Ansatz, so dass kein Änderungsbedarf besteht. Gemeinsame Grundlage für beide Ansätze sind die Prinzipien der FPA. Die Komplexität der Elementarfunktionen, die von der FPA identifiziert werden, lässt sich den Klassen "niedrig", "mittel" und "hoch" zuordnen. Die gleiche Klassifizierung wird auch bei Änderungen angewendet.

Der nächste Schritt ist ziemlich naheliegend, verwenden der gleichen Variablen für die Klassifizierung der Änderungen, wie sie auch bei der Klassifizierung der Funktionen verwendet werden. Was sind Änderungen? In dem beschriebenen Ansatz gehören alle Elemente dazu, die anders sind als zuvor:

$$\text{Gesamtanzahl der Änderungen} = \text{Anzahl neuer Elemente} + \text{Anzahl geänderter Elemente} + \text{Anzahl entfernter Elemente}.$$

Zur Bestimmung der Änderungen der Dateien (ILF, EIF) wird die Anzahl der Änderungen identifiziert, die in Zusammenhang mit den record types / Datensuggruppe (ΔRT) und den Data Element Types / Datenelementen (ΔDET) stehen. Für die Transaction types (EI, EO, EQ) ist die Anzahl der Änderungen der file types referenced / benutzte Dateien (ΔFTR) und die Data Element Types (ΔDET) relevant.

Bewertung basierend auf der Anzahl der Änderungen

Nachdem die Anzahl der Änderungen bestimmt wurde, wird der Einflussfaktor aus der Matrix ausgelesen. Es wurde nur eine Matrix für transaction types (EI, EO, EQ) definiert.

ΔDET	0 – 1	2 – 5	> 5
ΔFTR			
0	N	N	M
1 – 2	N	M	H
> 2	M	H	H

Eine ähnliche Matrix wird für logical files (ILF, EIF) benutzt. In diesem Fall hat ΔRT keinen Einfluss auf den Aufwand, so dass nur die erste Zeile genutzt wird.

ΔDET	0 – 1	2 – 5	> 5
--------------	-------	-------	-----

-	N	N	M
---	---	---	---

Verändert ein logical file den Typ verändert, wird wie folgt vorgegangen:

EIF -> ILF der Einfluss-Faktor wird auf "niedrig" für die neue Größe des ILF gesetzt, bei Veränderung in umgekehrter Richtung, ILF -> EIF geschieht nichts.

Beispiel: Die Personalabteilung benötigt eine Geburtstagsliste aller Angestellten in der Datei Angestellte (Vorname, Nachname, Abteilung, Büro und Geburtsdatum) sortiert nach dem Geburtsdatum. Die Informationen über Abteilung und Büro kommen aus der Datei Abteilung.

Berechnung: Transaktionstyp - External Output / Externe Ausgabe, Anzahl benutzte Dateien (FTR) - 2, Anzahl Datenelementen (DET) - 6, Umfang - 5 Function Points;

RfC: Füge Telefonnummer und email zu der Liste. Beide Werte kommen aus der Datei Angestellter.

Berechnung: Transaktionstyp - External Output, Anzahl FTR - 2, Anzahl DET - 8, Umfang - 5 Function Points; Δ FTR - 0, Δ DET - 2, Verbesserungstyp = niedrig.

Bewertung basierend relativen Änderungen

Diese Bewertung ist in "FPA für Softwareverbesserungen" beschrieben, es wird er oben erläuterte Ansatz genutzt. Der einzige Unterschied ist die Bestimmung der Variablen für die Bewertung des Einflusses.

$\% \text{ FTR} = (\Delta \text{ FTR} / \text{Anzahl FTR vorher}) \times 100\%$ $\% \text{ DET} = (\Delta \text{ DET} / \text{Anzahl DET's vorher}) \times 100\%$

Für diese Methode gibt es eine Matrix für die logical files,

	% DET	< 33.3%	33.3% - 66.7%	66.8% - 100%	> 100%
-		I	II	III	IV

.... und eine für Transaktionsarten (EI, EO, EQ).

	% DET	< 66.8 %	66.8% - 100%	> 100%
% FTR				
< 33.3%		I	II	III
33.3% - 66.7%		II	III	IV
66.8% - 100%		III	IV	V
> 100%		IV	V	VI

Änderung des Datentyps (EIF <-> ILF) bekommt einen Einflussfaktor von 0,4. Die Anzahl FP's die zur Bestimmung des MFP gebraucht wird entspricht der Anzahl FP's der ILF (alt/neu).

Beispiel: Geburtstagsliste

RfC: Zufügen von Telefonnummer und Email zu dieser Liste. Beides ist verfügbar in der Tabelle EMPLOYEE.

Berechnung: Transaktionstyp - External Output, Anzahl FTR - 2, Anzahl DET - 8, Umfang - 5 Funktionspunkte; % FTR - 0, % DET - 66,7%, Verbesserungstyp - I.

Schritt 3: Umfangbestimmung eines Verbesserungsprojektes

Der Umfang wird hergeleitet in FP's für jedes Element durch die Zuordnung der Verbesserungsarten zu den betroffenen Funktionalitäten (logical files und transaction types). Die Verbesserungsart korrespondiert mit dem Einflussfaktor.

Das Ziel des Einflussfaktors ist die Definition der Korrelation zwischen Änderungsaufwand und Aufwand für eine Neuentwicklung. Die Werte der verschiedenen Einflussfaktoren sind empirisch bestimmt worden. In zehn Jahren Anwendungserfahrung ergab sich keine Notwendigkeit die Werte zu ändern. Diese Erfahrungen beziehen sich dabei nur auf den eigenen Ansatz. In der folgenden Tabelle werden die Einflussfaktoren der beiden Ansätze verglichen.

	Verbesserungsart		Einflussfaktor	
	Sogeti	NESMA	Sogeti	NESMA
Neu	Neu	Neu	1,00	1,00
Änderung	niedrig	I	0,25	0,25
	mittel	II	0,50	0,50
	hoch	III	0,75	0,75
	ersetzen	IV	1,10	1,00
		V		1,25
VI		1,50		
entfernen	entfernen	entfernen	0,10	0,40
Test (ausschließlich)	Test (ausschließlich)	-	0,10	-

Der Einflussfaktor dient der Berechnung des Umfangs in MFP für jedes betroffene Element. Die Summe bzw. der Umfang für jedes Element zeigt die Gesamtgröße des Verbesserungsprojektes.

Beispiel Geburtstagsliste:

Berechnung: aktuelle Größe - 5 FP's, Verbesserungsart - Low, Einflussfaktor - 0,25, Entwicklungsgröße - 1,25

Beispiel zur Bestimmung des Umfangs eines Projektes mit der FPA

Hinweise: Zuordnung der Entwicklungsarten wird hier nicht beschrieben. Die RfC beinhaltet lediglich Änderungen die die Transaktionstypen FPA-1 bis FPA-6 und den neuen Transaktionstyp FPA-7 betreffen.

Trans. Type	Vorher	Neu	Änderung			Entfern	Test	Total	Nachher
			N	M	H				
FPA-1	4						4		4
FPA-2	7						7		7
FPA-3	4		4						4
FPA-4	6			6					6
FPA-5	5				7				7
FPA-6	7			7					7
FPA-7	0	4							4
FP	33	4	4	13	7	0	11		39
Einfluss		1,00	0,25	0,50	0,75	0,10	0,10		
MFP		4,00	2,00	6,50	5,25	0,00	1,10	18,85	

Um den Aufwand abzuschätzen wird eine Produktivitätsrate von 8 Stunden/FP angenommen. Dies beinhaltet alle Aktivitäten die zu der Neuentwicklung gehören. Da der Einflussfaktor den Aufwand gegen die Neuentwicklung abwägt, kann die Produktivitätsrate ebenfalls für die Änderungen genutzt werden. Für den Basisaufwand ergibt sich ein Wert von 151 ($18,85 \times 8$ St/FP) Stunden.

4.3 Abschätzung von Produktpflege und Anpassungen mit CFFP

Schritt 1: Bestimmung des Releaseumfangs

In der Diagnosephase wird die von der Wartung betroffenen Funktionalitäten identifiziert. Basierend auf den Angaben der BfC werden auch alle Funktionalprozesse identifiziert die in irgendeiner Art und Weise von der Verbesserung betroffen sind. Der Releaseumfang ist die Summe der Umfänge aller identifizierten Funktionalprozesse. Der Umfang wird in COSMIC Funktional Size Units (CFSU) angegeben.

Schritt 2: Zuordnung der Erweiterungsarten

Die gleichen Erweiterungsarten werden bei der CFFP berücksichtigt. Der größte Unterschied ist die Betrachtungsweise der Änderungen. Ein Funktionalprozess

wird gemessen durch Zählung all implementierter Data Movements / Datenbewegungen (Entry / Eingänge, Read / Lesen, Write / Schreiben und Exit / Ausgänge). Änderungen werden gemessen durch Identifizierung der durch die RfC betroffenen Datenbewegungen.

Beispiel Geburtstagsliste:

Berechnung: Datenbewegungen - 5 (lesen Angestellter, lesen Abteilung, verlassen Angestellter, verlassen Abteilung, verlassen Mitteilung), Umfang - 5 CFSU

RfC: Berechnung: Datenbewegungen bewirkt 2 (lesen EMPLOYEE, verlassen EMPLOYEE data) Verbesserungsumfang – 2 cfsu.

Schritt 3: Umfangbestimmung des Verbesserungsprojekte

Nach der Zuordnung der Verbesserungsarten zu den jeweiligen Funktionalprozessen setzt sich der Umfang aus den tatsächlichen Umfängen in cfsu pro Element zusammen. Die Verbesserungsart korrespondiert mit dem Einflussfaktor, ausgenommen bei Änderungen (Siehe Schritt 2).

Anhand des Beispiels der FPA, wird demonstriert wie es funktioniert.

Funkt. Prozess	vorher	neu	Änderung	Entfern	Test	Total	Nachher
FPA-1	5				5		5
FPA-2	8				8		8
FPA-3	5		2				5
FPA-4	8		3				8
FPA-5	6		4				7
FPA-6	9		5				9
FPA-7	0	5					5
cfsu	41	5	14	0	13		47
Einfluss		1,00	1,00	0,10	0,10		
(m)cfsu		5,00	14,00	0,00	1,30	20,30	

Die Produktivitätsrate beträgt 7,2 St./cfsu. Somit ergibt sich ein Gesamtaufwand von 146 Stunden (20,30 cfsu x 7,2 St./cfsu).

5 Anwendung der Funktionenmessung in Wartungsprojekten

5.1 Outsourcing

Dieses Fallbeispiel beschreibt eine Werkzeugfirma, welche ein Betriebssystem dass seit mehr als 15 Jahren von einer Software Servicefirma betreut wird. Dies

beinhaltet Softwarereparatur (bug fixing), Softwareerweiterung, helpdesk und Schulung.

Das Management der Werkzeugfirma möchte einen Einblick in die Leistung der Servicefirma und so auch die Kosten in den Griff bekommen. Sogeti wurde beauftragt den aktuellen Vertrag einzuschätzen und eine Blaupause für einen neuen Vertrag der auf einer leistungsabhängigen Bezahlung basiert.

Zunächst wurde der Umfang der Applikation bestimmt. Das System lief seit 15 Jahren, ausreichende Dokumentation war nicht vorhanden, so dass die Umfangsmessung anhand der Bedienungsanleitung und eines Live-Tests am laufenden System gemacht wurde. Es wurde ein Größe von 6900 Funktionspunkten festgestellt. Dafür wurden 104 Stunden gebraucht inklusive Vorbereitung und Berichterstattung.

Die Performance Analyse basierte auf einem Vergleich abgelaufener Projekte und den zugehörigen Rechnungen, was 40 Stunden dauerte. Für die Erstellung einer neuen Vertragsvorlage wurden 16 Stunden benötigt. Insgesamt dauerte es 168 Stunden um einen neuen Vertrag aufzusetzen, hiermit konnten beide Parteien gut leben.

Die Basis der Lieferperformance war eine Produktivitätsrate von 8 St. pro FP. Jedoch wurde aufgrund der speziellen Systemarchitektur (Modellierung und wieder verwendbare Routinen) eine Produktivitätsrate von 6,5 St. pro FP im Vertrag festgelegt. Die Umfangbestimmung des Releases basierte auf der Sogeti Methode zur Bestimmung von Erweiterungsprojekten. Die Rechnung der Servicefirma soll den bearbeiteten Umfang in MFP auflisten. Falls notwendig, würde Sogeti den Umfang der Erweiterung bestimmen.

Für die anderen Performance Indikatoren einigte man sich auf eine jährliche Bestimmung. Für Schulung, korrigierende Wartung und Helpdesk wurden die folgenden Performance Indikatoren gewählt: 0,15 St./FP; 0,10 St./ FP; 0,10 St./ FP. Würde die Werkzeugfirma diese Aktivitäten an den gleichen Lieferanten auslagern, ergäben sich durch Synergien Vorteile und somit ein All-In-Indikator von 0,3 St./FP.

Nach einem Jahr werden die Indikatoren überprüft und falls nötig angepasst. Ebenso wurde der All-In-Indikator auf 0,35 St./FP erhöht.

Die mittleren Wartungskosten sanken um fast 10% und die Kundenzufriedenheit stieg. Letzteres wurde auf die exaktere Bestimmung der Lieferungstermine der Erweiterungsprojekte zurückgeführt.

5.2 Release Management

Die IT Abteilung einer öffentlichen Organisation muss 3 Releases pro Jahr liefern. Das begrenzte Budget erlaubte keine Erweiterungen der Ressourcen. Sowohl Geschäftsabteilung als auch das IT Management waren nicht zufrieden mit dem

Release Fortschritt, jedes Mal war der Liefertermin gefährdet. Zumeist enthielt das Release nicht die geforderten Funktionalitäten. Dieses wirkte sich entsprechend ungünstig auf das nächste Release aus. Die Einführung der Funktionsumfangsmessung soll diesen Releaseprozess besser Steuerbar machen.

Die drei vorangegangenen Releases wurden anhand des Sogeti Wartungsansatzes untersucht. Die Produktivitätsrate wurde basierend auf Lieferumfang und Arbeitszeit zu 12,5 St./FP bestimmt. Darauf basierend kann die Anzahl (m)fp, die in einem Release geliefert werden können, ermittelt werden. Die zur Verfügung stehende Kapazität betrug 128 Mann Monate pro Release, wobei ein Mannmonat 120 St. (21,75 Tage x 8 St./Tag x 0,7) entspricht. Unter Berücksichtigung der Sommerferien wurde für das Sommerrelease ein Mannmonat mit 110 St. Und für die anderen Releases ein Mannmonat mit 125 St. angenommen (was auch gut zur Produktivitätsrate passt). Die Erfahrungen der letzten drei Releases zeigen, dass ca. 10% der Zeit für die Wartung des aktuellen Releases gebraucht wurde und ca. 15% für hotfixes. Die Unterstützung der Akzeptanztests schlägt mit 5% der zeit zu Buche. So dass nur noch 70% der Zeit effektiv für die Arbeit am aktuellen Release übrig blieb. Für ein reguläres Release entspricht ein Mannmonat 7 (M)FP, mit 128 Mannmonaten ergeben sich ca. 900 FP's. Der Pilot wurde mit 800 FP's angesetzt.

Die Geschäftsleitung zeigte zunächst wenig Vertrauen und war auch unzufrieden. Die Endnutzer mussten sich auf ein kleineres Release als gewünscht einigen und hatten die Erfahrungen im Hinterkopf. Als das Pilotprojekt ohne den üblichen Stress abgeliefert wurde und die vorher beschriebenen Funktionalitäten enthielt, waren alle Abteilungen sehr zufrieden. Die Releases waren nun kleiner im Umfang, jedoch konnte aufgrund gesteigerter Produktivität (11,2 St./FP) ausreichend Funktionalität in das Release eingebaut werden, die auch den Erwartungen entsprachen. Insgesamt waren die Endbenutzer zufriedener und der Release Managementprozess lief kontrolliert ab.

6 Schlußfolgerung

Mit der Erweiterung des Umfang der Meßmethoden, Funktionspunktanalysen und COSMIC Full Function Point, wie in diesem Dokument beschrieben, sind Messungen von Verbesserungen eines Projekts möglich. Die Produktivitätsraten von Neuentwicklungen können bei Verbesserungsprojekten verwendet werden, weil die Bewertung von Änderungen relative zu Neuentwicklungen ist.

Literaturhinweise

Englische Fassung dieser Tagung ist veröffentlicht in:

“Investigations in Software Measurement, Proceedings of the 13th International Workshop on Software Measurement”, September 23-25, 2003, Montréal, Canada, Shaker Verlag, ISBN 3-8322-1880-7

1. NESMA (1996), “Definitions and counting guidelines for the application of function point analysis A practical manual, version 2.0”, <http://www.nesma.org>
IFPUG (1999), “Function Point Counting Practices Manual 4.1”, <http://www.ifpug.org>
2. Abran, J.M. Deshairnes, S. Oigny, S. St-Pierre, C. Symons (2003), “Measurement Manual COSMIC Full Function Points 2.2, The COSMIC Implementation Guide for ISO/IEC 19761”, <http://www.cosmicon.com>
3. ISBSG (2003), “The ISBSG Estimation, Benchmarking & Research Suite (release 8)”, International Software Benchmark Standards Group, <http://www.isbsg.org.au>
4. B.P. Lientz, E.B. Swanson (1980), "Software Maintenance Management: A Study of the Maintenance of Computer Application Software in 487 Data Processing Organizations", Addison-Wesley, ASIN: 0201042053
5. Ton Dekkers (1994), "Van Functiepuntanalyse naar Integrale Functiepuntanalyse", IP/Informatica Projectgroep, ton.dekkers@sogeti.nl
6. NESMA (1996), “Rapport Functiepuntanalyse in onderhoud en beheer, functionele en technische richtlijnen, versie 1.0”, <http://www.nesma.org>
7. NESMA (2001), “Function Point Analysis for Software Enhancement”, <http://www.nesma.org>