

HANSER

Ernest Wallmüller

**SPI - Software Process
Improvement mit CMMI,
PSP/TSP und ISO 15504**

ISBN-10: 3-446-40492-9

ISBN-13: 978-3-446-40492-2

Leseprobe

Weitere Informationen oder Bestellungen unter
<http://www.hanser.de/978-3-446-40492-2>
sowie im Buchhandel

9 Management und Implementierung von SPI-Programmen

Im Folgenden diskutieren wir verschiedene Ansätze, Vorgehensweisen und Erfahrungswerte bei der Planung, beim Management und bei der Implementierung von SPI-Projekten und -Programmen. Danach werden Kosten- und Nutzenüberlegungen sowie Auswirkungen von SPI-Vorhaben anhand von Erfahrungsberichten diverser Firmen dargestellt. In Abschnitt 9.3 werden Risiken, deren Vermeidung und kritische Erfolgsfaktoren analysiert. Den Abschluss des Kapitels bilden Erfahrungsberichte der Firmen AVL, Credit Suisse, Alcatel und Valeo, die SPI-Vorhaben mit CMMI bzw. ISO 15504 implementieren.

9.1 Vorgehen in SPI-Programmen

Viele Organisationen haben einen steigenden Bedarf nach methodischer Unterstützung bei der Etablierung, Implementierung und beim Management von SPI-Projekten und -Programmen, da diese sehr umfangreich bzw. komplex sind und die unterschiedlichsten Risiken enthalten. Grady hat bei HP [Grad97] bereits in den 90er Jahren erkannt, dass solche SPI-Vorhaben nach einem zyklischen, spiralenförmigen Schema ablaufen müssen. Dazu hat er auch den Verbesserungszyklus von Deming (Plan – Do – Check – Study/Act) angepasst und erweitert und unterscheidet im Rahmen eines zyklischen Vorgehens (siehe Abbildung 9.1 auf der nächsten Seite) die folgenden immer wiederkehrenden Phasen:

- Legitimation
- Diagnose
- Planung
- Umsetzung mit Lessons Learned

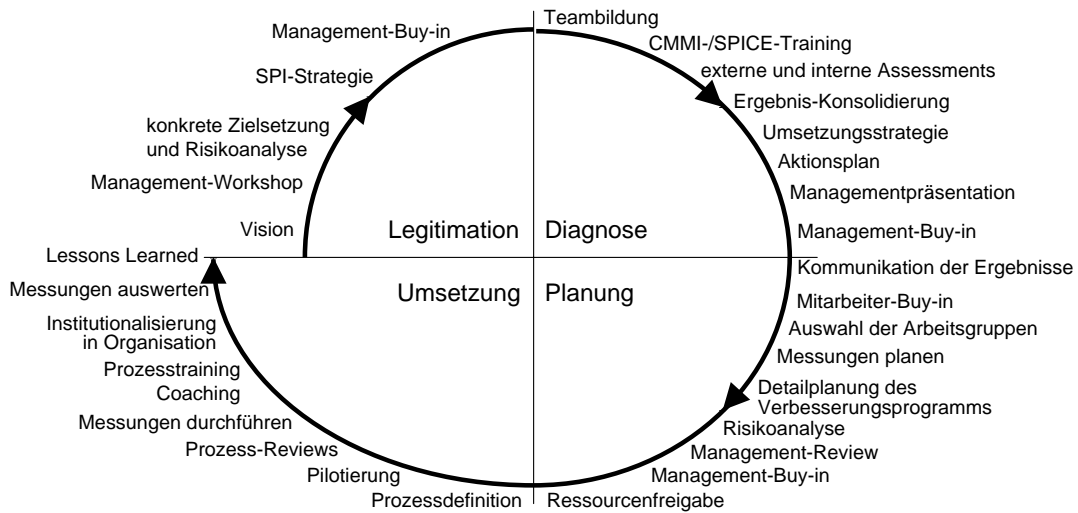


Abbildung 9.1 L-D-P-U-Verbesserungsmodell nach Grady

Legitimation

Im Rahmen der Legitimation legt man die Voraussetzungen und Eckpfeiler für das SPI-Vorhaben bzw. für die nächste Iteration fest. Beim erstmaligen Durchlauf wird eine Engineering-Prozessgruppen-(EPG-)Organisation geschaffen, die als Träger der Verbesserungsinitiative fungiert. Zusätzlich wird eine unabhängige PPQA-Gruppe (Prozess- und Produktqualitätssicherung) etabliert, falls sie noch nicht vorhanden ist. Insbesondere stellt man durch einen Sponsor aus dem höheren Management für das SPI-Vorhaben die Managementverpflichtung und Ressourcenverfügbarkeit sicher, legt die strategischen und taktischen Ziele sowie die mittel- und langfristige SPI-Strategie fest oder überarbeitet sie. Es werden eine detaillierte Stakeholderanalyse und ein Informations- und Kommunikationsplan erstellt.

Diagnose

Danach erfolgt die Diagnose meist in Form eines externen Assessments (SCAMPI Class B oder C). Im Rahmen dieser Diagnose wird Wert auf die Schulung des Managements in Sachen CMMI-Modell, Managementstrategien und -weisungen (Policies) sowie generische Praktiken gelegt. Die Assessmentergebnisse werden konsolidiert und eine Umsetzungsstrategie mit Verbesserungspotenzialen erstellt.

Planung

Im Rahmen der Planungsphase werden die Verbesserungspotenziale verfeinert, detaillierte Prioritäten auf Verbesserungsmaßnahmen gelegt, die Reihenfolge der Verbesserungspakete bestimmt und ein operativer Zeit- bzw. Verbesserungsplan erstellt. In dieser Phase stellt man die Prozessaktionsteams (PAT) zusammen. Die Bereitstellung und Verfügbarkeit von

ausreichenden Ressourcen spielt hier eine große Rolle. Risikoanalysen werden durchgeführt und geeignete Vermeidungs- und Notfallmaßnahmen initiiert bzw. vorgesehen.

Umsetzung mit Lessons Learned

In der Umsetzungsphase erfolgt die Implementierung der Verbesserungsmaßnahmen durch die PAT, Prozesseigner und Prozesscoaches. Ausreichende Prozessreviews stellen sicher, dass die Verbesserungen bzw. die erstellten oder geänderten Prozesse den Anforderungen entsprechen. Die Implementierung und Institutionalisierung der verbesserten Prozesse erfordert eine ausreichende Schulung und ausreichendes Coaching der Betroffenen. Messungen und interne Assessments zeigen auf, inwieweit die Prozesse wirksam sind und auch eingehalten werden. Mit Erfahrungsworkshops und Lessons-Learned-Sitzungen stellt man sicher, dass organisationsweites Lernen und organisationsweite Ausbreitung der Ergebnisse stattfinden.

Die Spirale bestehend aus Legitimation – Diagnose – Planung – Umsetzung wird mit einer Durchlaufzeit von 12 bis 24 Monaten mehrmals durchlaufen.

Das IDEAL-Modell

Das SEI bietet in Zusammenhang mit CMMI das Verbesserungsmodell IDEAL [McFe96] an, das einen systematischen Ansatz bzw. ein Vorgehen für das Management von technologiebedingten Veränderungen und Anpassungen anhand eines Lebenszyklusmodells enthält. Mit der neuen Version 1.1 von IDEAL lassen sich nicht nur SPI-Projekte, sondern auch Prozesse und Verfahren, wie z. B. Risikomanagement oder neue Entwicklungsumgebungen (rechnergestützte Werkzeuge), einführen.

IDEAL (siehe Abbildung 9.2 auf der nächsten Seite) ist ein Engineering-orientiertes Vorgehensmodell und besteht aus Phasen, Aktivitäten und Prinzipien, mit denen man eine langfristige Verbesserungsstrategie umsetzt. Das Modell bietet folgende Phasen an:

■ *I – Initiierung*

In dieser initialen Phase legt man die Grundlagen für ein erfolgreiches Verbesserungsprogramm fest. Die wesentlichen Aufgaben bestehen darin, Treiber für die Veränderung zu identifizieren, den Kontext des Vorhabens festzulegen sowie Sponsorship und Infrastruktur aufzubauen. Erfahrungen des SEI zeigen, dass 2–3 % der Ressourcen der Entwicklungsmannschaft bzw. 3–5 % des Entwicklungsbudgets dafür bereitzustellen sind.

■ *D – Diagnose*

Hier wird evaluiert, wo die Organisation in Bezug auf CMMI steht, und festgelegt, wohin die Reise geht (Geschäfts- bzw. Verbesserungsziele, Reife- oder Fähigkeitsstufe, Entwicklung von Empfehlungen).

■ *E – Etablieren*

Ein Verbesserungsplan wird etabliert, in dem man festlegt, wie, bis wann und mit welchen Ressourcen die Organisation die in Phase D spezifizierten Ziele erreichen will. Die Aufgaben bestehen im Setzen von Prioritäten, in der Entwicklung eines Verbesserungsansatzes und der Erstellung detaillierter Maßnahmenpakete.

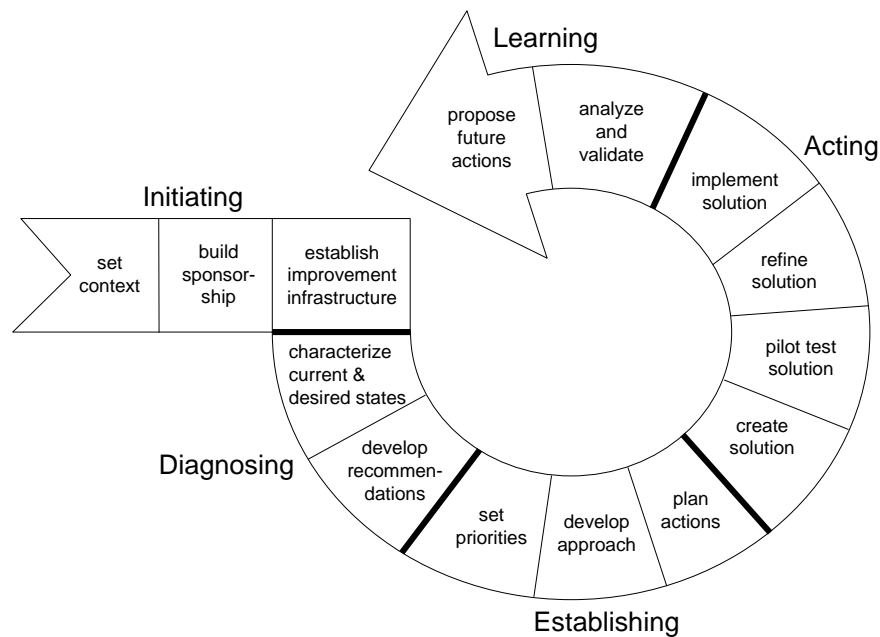


Abbildung 9.2 Das IDEAL-Modell

■ *A – Agieren*

Agieren bedeutet Ausführen der Arbeit (Arbeitspakete), wie sie im Verbesserungsplan festgelegt wurde. Die Aufgaben im Detail: eine Lösung entwickeln, diese pilotieren und testen; die Lösung verfeinern und implementieren.

■ *L – Lernen*

Lernen heißt, aus den gesammelten Erfahrungen die richtigen Schlussfolgerungen (Analyse und Validierung der erzielten Ergebnisse) und Maßnahmen abzuleiten sowie die Fähigkeit der Organisation zu verbessern, um neue Technologien und Prozesse zu adaptieren.

Darüber hinaus wurde am SEI ein Web-basierter Prozessleitfaden, genannt INTRO, entwickelt, der einer Organisation aufbauend auf dem IDEAL-Modell bei der Einführung neuer Tools und Technologien helfen soll.

Sowohl für die strategische als auch für die operative Verbesserungsplanung sind Hilfsmittel erforderlich. In Anhang A4 und A5 finden Sie je ein Inhaltsverzeichnis für einen operativen sowie einen strategischen Verbesserungsplan.

9.2 Kosten, Nutzen und ROI von SPI-Programmen

Vor dem Start eines SPI-Programms und auch danach tauchen immer wieder Fragen bezüglich der Kosten und des Nutzens solcher Programme auf. Dem Autor ist in diesem Zusammenhang in den letzten zwanzig Jahren Folgendes aufgefallen:

- Geschäftsleitungsmitglieder – somit die Sponsoren des SPI-Vorhabens – werden gegenwärtig über ihr Gehalt und insbesondere ihr flexibles Bonusprogramm motiviert. Diese Entlohnungsbestandteile sind stark an die Leistungen des Unternehmens im aktuellen Jahr gebunden, nicht an das, was das Unternehmen in zwei oder drei Jahren verdient oder erwirtschaftet. Jede vorgeschlagene Verbesserung, die eine Investition bedeutet und deren Nutzen sich in „harten“ Geschäftszahlen erst in mehreren Jahre niederschlägt, wird gegenüber Investitionen, die sich kurzfristig rentieren, zurückgestellt.
- Geschäftsleitungsmitglieder haben in der Regel mehr Vorschläge an Verbesserungen, als sie finanzieren können. Jede Softwareprozessverbesserung muss also im Vergleich mit anderen Verbesserungen positiv dargestellt werden.
- Technisch orientierte Mitarbeiter wissen in der Regel nicht, wie sie Investitionsentscheidungen rechtfertigen müssen. Ihnen ist bewusst, dass durch eine gute Organisation, reife Prozesse und praktisches, aktuelles Wissen bzw. gleichmäßig verteilte und aufbereitete Erfahrungen Produkte entwickelt werden, die die Zukunft der Firma und ihrer Mitarbeiter sichert. Höheres Management weiß leider oft zu wenig, wie komplex die Softwareerstellung heute funktioniert und was alles benötigt wird an erprobten Technologien, reifen Prozessen und aufeinander abgestimmten Hilfsmitteln, um am Markt bestehen zu können.

Diese Beobachtungen führen zu folgenden Überlegungen und Schlussfolgerungen:

- Die Softwareindustrie muss Prozessverbesserungen so umsetzen, dass der Nutzen früher auftritt bzw. wahrgenommen wird.
- Der Nutzen muss groß und objektiv feststellbar sein. Subtile Nutzensaussagen wie „die Produktqualität wird besser“ oder „wir werden schneller“ überzeugen kaum mehr.
- Es bedarf einfacher und besserer Hilfsmittel, Werkzeuge und Modelle, um einen Business-Case für beliebige technische Verbesserungen gestalten und rechnen zu können.

Durch einen ständigen Dialog zwischen technischen Mitarbeitern, Management und geschäftsorientierten Mitarbeitern aus Fachabteilungen wird ein gemeinsames Verständnis und die richtige Beurteilung des Werts von Prozessverbesserungen in der Softwareindustrie erarbeitet.

Messinformationen, die uns erlauben, SPI-Programme zu beurteilen, lassen sich in drei Kategorien einteilen:

- Informationen, die die Bereitschaft (readiness) für den Start eines SPI-Programms bestimmen helfen, beispielsweise Risikoinformationen aus Risikoanalysen oder verdichtete Informationen, wie z. B. das aufsummierte Risikogewicht des ganzen SPI-Programms in Geldeinheiten ausgedrückt.

- Informationen, um das SPI-Programm zu verfolgen und wirksam als ein Projekt oder eine Menge von Projekten (ein Programm) zu managen. Im Wesentlichen geht es dabei um Fortschrittsmetriken oder Abweichungsinformationen.
- Informationen, um das SPI-Programm von Beginn an bis zum Abschluss zu rechtfertigen.

Letzte Kategorie von Informationen werden zur Berechnung von ROI-Zahlen bzw. eines Business-Case verwendet. Die klassische Berechnung des ROI lautet:

$$\text{ROI eines SPI-Programms} = (\text{Programmnutzen} - \text{Programmkosten}) / \text{Programmkosten}$$

Die typischen Kosten eines SPI-Programms mit CMMI als Basis umfassen:

- Aufwandskosten für Arbeitspakete folgenden Typs:
 - Definition bzw. Auswahl und Anpassung der Prozesse inkl. einer geeigneten technischen Infrastruktur (z. B. Werkzeug zur strukturierten Bereitstellung der definierten Prozesse im Intranet)
 - Information und Schulung der Mitarbeiter (inkl. periodische Nachschulung und diverse Informationsveranstaltungen), insbesondere die offiziellen (SEI) CMMI-Einführungs- und Fortgeschrittenenkurse
 - kontinuierliche Überwachung der Einhaltung und Verbesserung der definierten Arbeitsprozesse (Messungen, Reviews, Audits, interne Assessments, externe Assessments z. B. vom Typ SCAMPI A)
- Kosten für die Anpassung der Arbeitsumgebung der Prozessnutzer, z. B. Beschaffung neuer Entwicklungswerkzeuge etc.
- Externe Dienstleistungen, wie z. B. externe Assessmentkosten und Consulting-Unterstützung
- Reisekosten und administrative Kosten

Die Erfassung und Berechnung der Kosten sind immer einfach im Vergleich zur Kalkulation des Nutzens. Die Messung des Nutzens ist schwierig, da die Leistung der Entwicklungsprojekte auf niedriger Reifestufe oft nicht messbar ist. Eine höhere Reifestufe führt zu einer einheitlichen Definition der Leistung als Grundlage für Messungen. Gerade auf niedriger Reifestufe wird die Forderung nach einem quantitativen Nachweis des Nutzens teilweise am vehementesten gestellt, obwohl die Voraussetzungen dort nicht gegeben sind. Das Nutzenpotenzial ändert sich mit der Prozessreife. Folgende Nutzenpotenziale lassen sich nach Kneuper [Kneu05] feststellen:

- von Reifegrad 1 nach 2
 - bessere Schätzungen
 - mehr/bessere Informationen als Grundlage für Entscheidungen und für Projektsteuerung
 - eindeutige Klärung der Anforderungen
 - deutliche Risikominderung

- von Reifegrad 2 nach 3
 - geringere Fehlerquoten
 - beginnende systematische Wiederverwendung
 - schnellere Einarbeitung von neuen Mitarbeitern oder bei Projektwechsel
 - neue Projekte lassen sich schneller und einfacher aufsetzen
 - lernende Organisation etabliert sich
- von Reifegrad 3 nach 4
 - hohe Zuverlässigkeit von Schätzungen
 - massive Wiederverwendung
 - deutliche Reduzierung von Fehlerquoten
 - erhebliche Produktivitätssteigerungen
- von Reifegrad 4 nach 5
 - umfassende Standardisierung und Automatisierung von Abläufen
 - erhebliche weitere Produktivitätssteigerungen

Projekte können nicht zum Vergleich parallel mit und ohne CMMI-Vorgehen durchgeführt werden. Dies würde Vergleichszahlen liefern. Die quantitative Abbildung von Veränderungen bei Prozessverbesserungen ist schwierig, da sich parallel viele andere Parameter ändern können. Beispielsweise sammeln Mitarbeiter Erfahrungen bei der Projektdurchführung und werden normalerweise allein durch diese Tätigkeit besser, auch ohne Prozessverbesserung. Änderungen der Technologie haben natürlich ebenfalls quantitative Auswirkungen. Bei gleichem Produkt gibt es technologiebedingte Verbesserungen von Release zu Release (zumindest für die ersten Releases). Dadurch sind für eine Nutzenberechnung viele Annahmen notwendig. Auch bei gutem Vorsatz ist es schwierig, diese Annahmen immer fair zu treffen.

Typische Nutzenaspekte von SPI-Programmen:

- Planbarkeit (Software wird zum geplanten Zeitpunkt mit geplanten Funktionen zu den geplanten Kosten fertig)
- Alleinstellungsmerkmal gegenüber Mitbewerbern
- Mehr Umsatz durch zusätzliche Aufträge von Kunden durch verbesserte Kundenzufriedenheit
- Mehr Umsatz durch zusätzliches Geschäft wegen verbesserter Produktqualität
- Mehr Umsatz von aktuellen Produkten (Produktlinien) wegen kürzerer Durchlaufzeiten und rascherer Einführung neuer Features/Funktionen
- Reduzierte Kosten durch Reduzierung von Verschwendung (rework)
- Reduzierte Kosten des Betriebs der Systeme und Applikationen durch weniger Fehler und Zwischenfälle im Feld
- Reduzierte Wartungskosten durch verbesserte Information/Dokumentation und bessere Nachvollziehbarkeit der Entwicklungsarbeit

- Reduzierte Änderungskosten durch die bidirektionale Nachverfolgbarkeit der Anforderungen
- Reduzierte Kosten durch verbesserte Produktivität

Im Folgenden geben wir konkrete Erfahrungen der Softwareindustrie hinsichtlich der Wirkung von SPI-Programmen wieder.

A) Performance-Ergebnisse des SEI

Das SEI veröffentlicht quartalsweise Rückmeldungen von Firmen über Erfahrungen mit Verbesserungsprogrammen auf der Basis von CMMI. Die Auswertung dieser Ergebnisse lässt gut erkennen, wie die Ergebnisse statistisch variieren. Den größten Nutzen von SPI-Programmen mit CMMI finden wir beim Faktor Produktivität. Beim Faktor Kundenzufriedenheit spielen hingegen diverse nicht-CMMI-bedingte Einflüsse eine Rolle. Die folgenden Ergebnisse wurden vom SEI im 2. Quartal 2006 festgestellt:

Performance Category	Median	Number of Data Points	Low	High
Cost	20 %	21	3 %	87 %
Schedule	37 %	19	2 %	90 %
Productivity	62 %	17	9 %	255 %
Quality	50 %	20	7 %	132 %
Customer Satisfaction	14 %	5	10 %	55 %
Return on Investment	5:1	16	2:1	28:1

B) Ergebnisse von Boeing

Boeing führt seit den 90er Jahren Verbesserungsprogramme zuerst mit SW-CMM und jetzt mit CMMI durch. Insgesamt wurden mehr als 10 000 Mitarbeiter in den Verbesserungsmodellen geschult. Im Boeing-Konzern sind alle Reifegrade vorzufinden. Interessant ist, dass Boeing ein Messprogramm mit der amerikanischen Universität in Kansas durchgeführt hat, um die Ermittlung der Messwerte unabhängig zu gestalten. Die folgenden Daten stammen vom Leiter des SPI-Programms, Herrn Vu, der sie bei einer SEPG-Konferenz in USA 2005 präsentiert hat. Er bezieht sich auf eine Organisationseinheit, die Software für kommerzielle und militärische Flugzeuge entwickelt.

- Zeitplanvariation bzw. Zeitplanabweichungen bei Softwareprojektschätzungen:
 - Level 3: + 20 % bis – 24 %
 - Level 4: + 12 % bis – 18 %
 - Level 5: + 4 % bis – 7 %
- Etablierung von formalen Design-, Code- und Testreviews über Quality-Gates:
 - Reworkaufwand um 31 % reduziert
 - Aufwand für formale Reviews 4 % des Entwicklungsaufwands

- Durchlaufzeit für die Behandlung eines Änderungsantrags (Change-Request) in Tagen (im Durchschnitt für einen Änderungsantrag):
 - 1998 Level 3: 50 (2) Tage
 - 2000 Level 4: 34 (5) Tage
 - 2004 Level 5: 29 (1) Tage
- Wiederverwendung von Artefakten inkl. Code und somit auch Kostenreduktion:
 - Level 2: 10 %
 - Level 3: 25 %
 - Level 4: 58 %
 - Level 5: 64 %
- Mitarbeiterzufriedenheit auf einer Skala von 1 (extrem unzufrieden) bis 10 (extrem zufrieden):
 - vor Prozessverbesserung: 5
 - nach Prozessverbesserung: 8
- Produktivität
 - Von Level 3 auf Level 5 hat die Arbeitsleistung pro Zeiteinheit durch bessere Prozesse um 150 % zugenommen und die Anzahl der Mitarbeiter konnte um 55 % reduziert werden.
- ROI (Return on Investment)
 - Nach 10 Jahren SPI mit SW-CMM und CMMI hat Boeing einen ROI von 2740 % und einen Nutzen-Kosten-Quotient von 28,5 erzielt.

Boeing ist davon überzeugt, dass der systematische SPI-Ansatz mit CMMI zu besseren Softwareprodukten und -systemen führt und die Geschäftszielerreichung sicherstellt. Dabei sind signifikante Verbesserungen realisierbar.

C) Siemens

Siemens führt seit längerer Zeit Softwareverbesserungsprogramme durch. Erfahrungen von Zopf und Völker [Völk01] bei Siemens PSE in Österreich zeigen eine starke Korrelation zwischen Kundenzufriedenheit und Prozessreife (siehe Abbildung 9.3).

Weitere Erfahrungen von Siemens, die an das SEI berichtet wurden, sind:

- Reduktion der Fehlerkosten von 45 % auf 30 % über 3 Jahre beim Wechsel von SW-CMM Level 5 zu CMMI Level 5
- Reduktion der Zeitplanabweichungen um durchschnittlich 82 % in 3 technischen Bereichen von 1999 bis 2004 von SW-CMM Level 3 bis CMMI Level 5
- 25 % Produktivitätszunahme über 3 Jahre beim Wechsel von SW-CMM Level 5 zu CMMI Level 5
- Reduktion der Fehlerdichte von 1 Defekt/KLOC um 71 % auf 0.25 Defekte/KLOC in 3 technischen Bereichen von 1999 bis 2004
- ROI von 2:1 über 3 Jahre von SW-CMM Level 5 zu CMMI Level 5

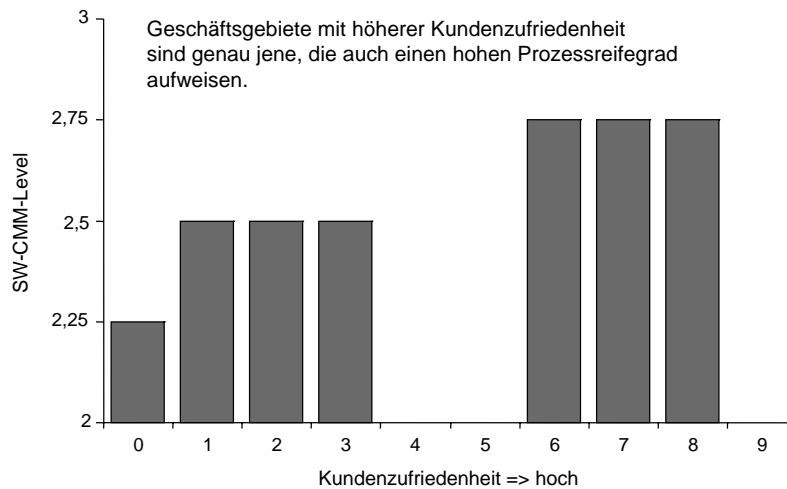


Abbildung 9.3 Korrelation von Prozessreifegrad und Kundenzufriedenheit

Eine gute Übersicht hinsichtlich der Ergebnisse von CMMI-basierten Prozessverbesserungen finden Sie in [Gibs06].

Zusammenfassung

Durch Prozessverbesserung nach CMMI sind erhebliche Verbesserungen möglich, doch nicht in jedem Einzelfall sicher, da es sich dabei um Nutzenpotenziale handelt.

Das Management muss bereit sein, sich selbst an definierte Prozesse zu halten bzw. zu akzeptieren, wenn Mitarbeiter das tun, auch wenn es im Moment unangenehm ist. Typische Beispiele sind Ad-hoc-Änderungen bei Anforderungen und Planung oder der Abschluss einer Phase bei vorhandenen Qualitätsproblemen.

Die Gefahr der Bürokratisierung an Stelle von effizienter Verbesserung besteht natürlich auch, wenn das Management des SPI-Vorhabens sich auf rein administrative Aufgaben beschränkt.

9.3 Risiken und kritische Erfolgsfaktoren von SPI-Vorhaben

Welche Softwareorganisation möchte nicht gerne ihre Produktivität steigern, bessere Produkte schneller ausliefern, die Wartungskosten reduzieren und die Kundenzufriedenheitswerte erhöhen? Das amerikanische Software-Engineering-Institut SEI veröffentlicht regelmäßig die Rückmeldungen verschiedener Unternehmen, die ihr SPI-Vorhaben mit CMMI durchführten (CMMI Performance Results: <http://www.sei.cmu.edu/cmmi/results.html>). Diese Zahlen des SEI und andere Erfolgsgeschichten von Unternehmen sind beeindruckend (siehe auch Abschnitt 9.2). Daneben gibt es aber auch eine steigende Zahl von Mel-

dungen über gescheiterte SPI-Vorhaben. Unsere Erfahrung zeigt, dass die Wahrscheinlichkeit des Versagens eines SPI-Programms ebenfalls recht groß ist – Statz geht dabei von einer 70-prozentigen Wahrscheinlichkeit aus [Stat97].

Ein neues Softwareprozessverbesserungsprogramm einzuführen kann ein riskantes Vorhaben darstellen. Um diese Risiken zu mindern, sind vom SPI-Team regelmäßige Risikobewertungen durchzuführen sowie ausreichende Maßnahmen zur Risikobekämpfung bzw. Risikovermeidung und für Notfälle zu setzen [Wall04].

Nachfolgend stellen wir die Gefahren an den Schlüsselpunkten eines SPI-Programmes dar. Nach Untersuchungen von Joyce Statz von TeraQuest/Borland aus dem Jahr 1997 [Stat97] sind folgende Risiken relevant:

Risiko-nummer	Mittleres Risikogewicht (Wahrscheinlichkeit mal Auswirkung)	Kategorie	Risikofaktor
1	3.6	SPI-Projektmanagement	Fähigkeiten und Erfahrung des SPI-Managers mit dieser Art von Projekten
2	3.6	SPI-Mitarbeiter	Erfahrungen der SPI-Mitarbeiter mit organisatorischen Veränderungen
3	3.5	Kultur	Ausmaß an Erfahrung mit der Verwendung von definierten Prozessen in der Organisation
4	3.3	SPI-Projektmanagement	Unterstützende Managementweisungen (Policies) und Standards für das SPI-Vorhaben
5	3.3	SPI-Projektmanagement	Adäquatheit der Bewertung von Auswirkungen durch das SPI-Vorhaben
6	3.3	Management der Organisation	Fähigkeit des Managements, als Team zu funktionieren
7	3.3	Auftrag und Ziele	Auswirkungen auf den Arbeitsablauf, die von SPI-Veränderungen ausgelöst werden
8	3.2	SPI-Mitarbeiter	Mitarbeiterverfügbarkeit für SPI-Aktionsteams
9	3.2	Management der Organisation	Ausmaß an Befähigung in der Organisation
10	3.1	Management der Organisation	Managementunterstützung für SPI-Vorhaben
11	3.1	Inhalt von Arbeitsergebnissen	vollständige und klare Anforderungen an das SPI-Projekt
12	2.9	Management der Organisation	Involvierung des höheren Managements im SPI-Programm

Besondere Risiken, die von außen während der Durchführung eines SPI-Vorhabens einwirken können, sind:

- organisatorische Instabilitäten, wie z. B. hohe Mitarbeiterfluktuationsrate
- häufige Restrukturierungen der Organisation, z. B. alle 6 Monate bis 2 Jahre
- signifikante Änderungen des Geschäftsfokus
- Verkauf der Organisation oder Fusion der Organisation mit einer anderen

Wieggers [Wieg05] unterscheidet Risiken bzw. Fallen vor und während des SPI-Vorhabens. Risiken vor bzw. bei der Planung des SPI-Vorhabens:

- nicht genügend Zeit
- Mangel an Wissen und Erfahrung
- falsche Motivation
- ein dogmatischer, zu starrer Ansatz
- ungenügende Verpflichtung aller Beteiligten

Risiken während des SPI-Vorhabens:

- Managementverpflichtung ungenügend oder nicht vorhanden
- unrealistische Managementenerwartungen
- Projektleiter, die nicht genügend Zeit für ihre Tätigkeiten aufbringen
- ungenügende Umsetzung von Verbesserungsmaßnahmen und -plänen
- CMMI- oder SPICE-Level sind das einzige Ziel; der praktische Nutzen für die tägliche Projektarbeit wird aus den Augen verloren
- Die SPI-Beteiligten werden ungenügend ausgebildet (im CMMI-Modell, im SPI-Vorgehen, in den Prozessen, in den nötigen Fähigkeiten bei der jeweiligen Prozessausführung)
- erwartete Prozesse machen die Mitarbeiter völlig austausch- und ersetzbar (Vernachlässigung von Know-how-Sicherung, Skill-Entwicklung und systematischem Training)
- Versagen bei der Skalierung und dem Tailoring der Prozesse
- SPI wird als Modetrend oder als Spiel gesehen und betrieben
- ineffiziente Durchführung interner Prozessassessments

Ein weiteres häufig anzutreffendes SPI-Risiko wurde von Pickerill [Pick99] festgestellt und veröffentlicht. Dabei geht es darum, wer die Umsetzungsverantwortung für SPI-Maßnahmen trägt. Vielfach wurden QM-Einheiten mit der Verantwortung für SPI-Maßnahmen vom Management beauftragt. Dies führt regelmäßig zu Umsetzungsschwierigkeiten und erheblichen zeitlichen Verzögerungen. Die einzige Lösung besteht in der Übernahme der Verantwortung durch das Entwicklungsmanagement selbst.

Zusammenfassung und Schlussfolgerungen

Basierend auf einer regelmäßigen Risikoanalyse, können SPI-Teams ihre Chancen für den Erfolg ihres Vorhabens beurteilen. SPI-Teams können aus den Risikodaten anderer Firmen und Projekte lernen. Sie können die möglichen Probleme reduzieren, indem sie Risikomanagement zu einem integralen Teil ihres Verbesserungsvorhabens machen. Eine Sammlung von Risikofaktoren in Form einer Risikocheckliste oder Risikozerlegungsstruktur (risk break down structure, RBS), die laufend ergänzt wird, kann dabei ein mächtiges Hilfsmittel sein.

9.4 Empirische Erfahrungen mit SPI-Programmen

Derzeit sind viele SPI-Programme weltweit am Laufen. Im nachfolgenden Abschnitt sollen verantwortliche Manager von SPI-Programmen zu Wort kommen. Sie stellen ihre Planungsgrundlagen, Vorgehensweisen und Erfahrungen vor.

9.4.1 Beherrschung der Komplexität in der Softwareentwicklung und -pflege mit CMMI

von M. Paulweber, AVL, Graz/A

Software gewinnt in der Automobilindustrie zunehmend strategische Bedeutung. Die Anzahl der Steuergeräte nimmt rapide zu, die Komplexität der Software in den Steuergeräten steigt exponentiell. Vergleicht man den Kostenanteil der Elektronik- und Softwareentwicklung mit dem Anteil der klassischen mechanischen Entwicklung, sieht man, dass die Kosten für die Elektronikentwicklung bereits höher geworden sind als die für Mechanik – was bedeutet, dass eine professionelle Softwareentwicklung für den Geschäftserfolg der Automobilfirmen außerordentlich wichtig wird. Meldungen über kostspielige Fahrzeugrückholaktionen aufgrund fehlerhafter Software in Steuergeräten, die man in den letzten Jahren immer wieder in der Presse lesen konnte, unterstreichen dies.

Es verwundert daher nicht, dass gerade Automobilfirmen große Anstrengungen unternehmen, um die Professionalität in der Softwareentwicklung zu erhöhen. Dabei hat die Einführung von CMMI-basierten Prozessen stark zugenommen. Nun stellt sich die Frage, warum gerade dieses Softwareentwicklungsprozessmodell so populär wurde und ob der nicht unerhebliche Einführungsaufwand überhaupt sinnvoll ist.

Produktiver – unproduktiver Aufwand in Softwareprojekten

Analysiert man den Verlauf der Arbeiten in großen Softwareprojekten, wie sie die Erstellung der Software für Fahrzeugentwicklungswerkzeuge oder -steuergeräte darstellt, sieht man oft, dass der Anteil der produktiven Arbeit gegen Ende der Projekte stark abnimmt. Zum einen kann man dies erklären, weil aufgrund der großen Komplexität der Software-

systeme die Anzahl der Fehler zunimmt, und zum anderen, weil für die Behebung dieser Probleme ein immer größerer Zeitanteil aufgewendet werden muss.

Dabei kommt es oft zu so genannten „Down-dates“ der Software, da die von einem Entwicklungsteam durchgeführten Fehlerbehebungen die Ergebnisse eines anderen Teams wieder ersetzen. Gute Projektleiter treffen in der Folge Gegenmaßnahmen, die sich mit einer Optimierung der Abläufe beschäftigen (und Verbesserungen im Softwareentwicklungsprozess einführen). Abbildung 9.4 [McCo97] zeigt diesen Verlauf: Der produktive Anteil nimmt aufgrund der unproduktiven Arbeiten, aber auch aufgrund des Zeitaufwands für die Prozessverbesserungen stark ab. Im schlechtesten Fall bewältigt das Projektteam seine Probleme überhaupt nicht, und es kommt zu einem Projektabbruch.

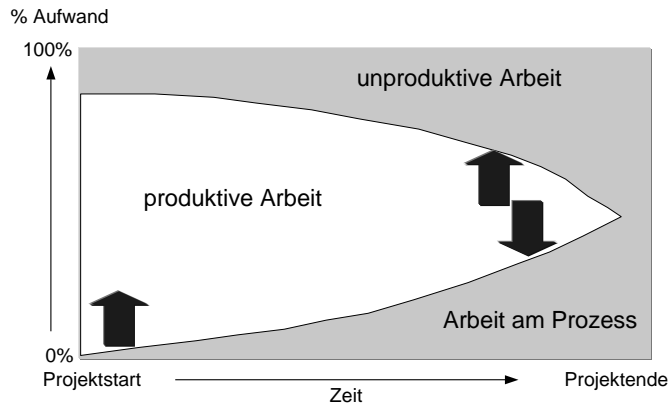


Abbildung 9.4 Schere 1: Produktive versus unproduktive Arbeit in Softwareprojekten (Quelle: „SW Project Survival Guide“ von Steve McConnell, Microsoft Press, 1998)

Die Einführung von CMMI als Prozessmodell hat zum Ziel, diesem unheilvollen Verlauf entgegenzuwirken und mit der gestiegenen Komplexität der Softwaresysteme fertig zu werden. Die Folge sind höhere Aufwände zu Beginn eines Softwareprojekts und ein scheinbar geringerer Projektfortschritt in der ersten Projektphase. Dies wird jedoch in der letzten Phase des Projekts mehr als wettgemacht, was zu einer wesentlichen Effizienzsteigerung in der Softwareentwicklung führt.

Abbildung 9.5 zeigt aber auch ein großes Problem bei der Einführung von CMMI oder ähnlichen Softwareentwicklungsprozessen: Aufgrund des zu Beginn langsameren Projektfortschritts werden viele Softwareentwicklungsingenieure und deren Management oft ungeduldig und glauben, ohne diese Prozessaufwände vorgehen zu müssen. Erst später zeigt sich, dass der ungeplante Fehlerbehebungsaufwand zunimmt und das Softwareprojekt verspätet und mit einer Kostenüberschreitung abgeschlossen wird.

Daher haben Softwareprojektleiter die Wahl, am Anfang ihres Projekts Aufwände in Softwareentwicklungsprozessarbeit zu investieren oder am Ende des Projekts einen meist wesentlich größeren Aufwand für die Softwarestabilisierung aufbringen zu müssen. Der sinnvollere Weg liegt dabei auf der Hand.

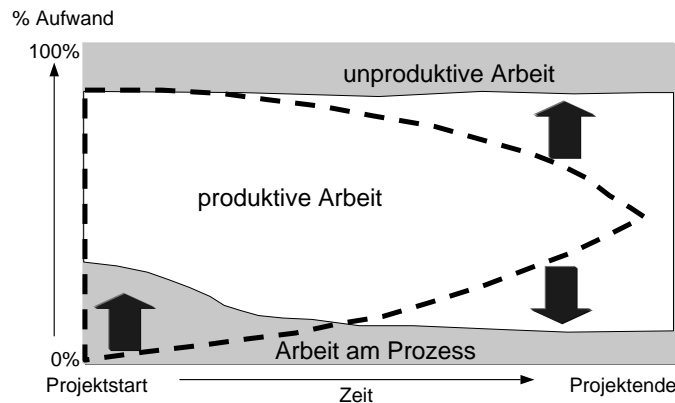


Abbildung 9.5 Schere 2: Produktive versus unproduktive Arbeit in Softwareprojekten (Quelle: „SW Project Survival Guide“ von Steve McConnell, Microsoft Press, 1998)

Einführung einer professionellen Softwareentwicklung mit CMMI

Bei der Einführung von Prozessen in der Softwareentwicklung sind fünf Punkte entscheidend:

- ein gemeinsames Ziel
- eine effiziente und schlanke Prozessorganisation
- ausreichendes Training
- gute Softwareentwicklungswerkzeuge
- die Einführung von Feedback-Schleifen

Ein gemeinsames Ziel

Weil die Einführung von Softwareprozessen am Anfang die Entwicklungsarbeit zu verzögern scheint, ist es unabdingbar, dass sie mit der Geschäftsleitung abgestimmt und von dieser vorbehaltlos unterstützt wird. Nur so kann der anfängliche Widerstand der erfahrenen Softwareentwicklungsingenieure überwunden werden, der zum einen durch den Änderungsprozess, zum anderen durch die anfängliche Verlangsamung im Projekt entsteht.

Es hat sich auch gezeigt, dass ein gemeinsames Ziel, wie die Erreichung einer offiziellen CMMI-Reifestufe, sehr unterstützend wirkt. Ein erreichter Level gewinnt außerdem in der Automobilindustrie zunehmend an Marketingwert, da die großen Automobilfirmen ebenfalls solche Assessments in ihrem Bereich durchführen und um den Wert dieser Anstrengungen wissen.

Der Aufwand der Softwareprozesseinführung steigt mit dem Level, den man einführen will. Für die Einheit ist es wichtig, möglichst bald mess- und sichtbare Erfolge nach den Anstrengungen einer Einführung zu erreichen. Daher stellt sich die Frage, ob es sinnvoll ist, mit der Einführung des CMMI Level 2 zu beginnen. Ein vom SEI veröffentlichtes Diagramm zeigt, dass die mittlere Anzahl von Fehlern mit dem Erreichen des CMMI Level 2 um ungefähr 50 % zurückgegangen ist (siehe Abbildung 9.6).

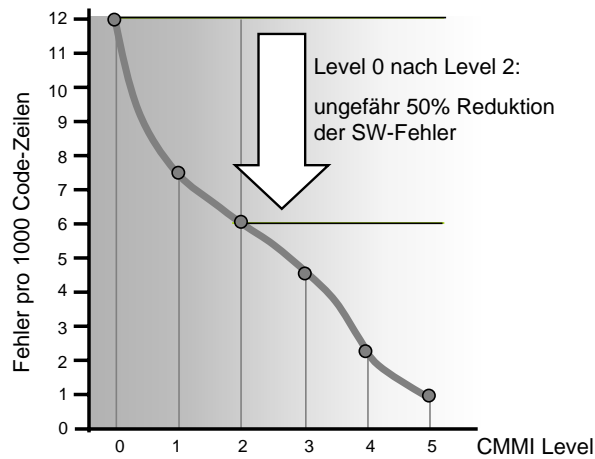


Abbildung 9.6 Fehlerdichte je CMMI-Reifestufe
(Quelle: SEI – SW Engineering Institute, Carnegie Mellon University)

Da, wie bekannt, die Behebung eines Fehlers umso teurer wird, je später sie durchgeführt werden muss, lässt sich das Einsparungspotenzial leicht erahnen.

Als erstes Ziel hat sich daher das Erreichen des CMMI Level 2 durchgesetzt. Die dafür benötigte Zeit hängt von der Ausgangssituation der Einheit ab, liegt aber im Durchschnitt bei ungefähr zwei Jahren. Level 3 kann dann nach weiteren ein bis zwei Jahren erreicht werden.

Eine effiziente und schlanke Prozessorganisation

Eine Prozessorganisation wie in Abbildung 9.7 hat sich als wirksam erwiesen. Sie hat eine Balance zwischen effizienter und schlanker Prozessorganisation zum Ziel.

- SPO Der „Strategic Process Owner“ ist der Sponsor in der Geschäftsführung.
- OPO Der „Operational Process Owner“ ist für die Koordination aller Prozessaktivitäten verantwortlich.
- SPSC Das „Software Process Steering Committee“ gibt Prozessdefinitionen frei und überwacht deren Implementation. Es besteht aus den Entwicklungsleitern aller großen Softwareentwicklungseinheiten, dem SPO und dem OPO.
- SEPG Die „Software Engineering Process Group“ besteht aus einem erfahrenen Vertreter jedes großen Entwicklungsstandorts. Sie coacht in Softwareprozessfragen und ist für die Verbreitung des Prozesswissens an den Standorten verantwortlich.
- SEAG Die „Software Engineering Architecture Group“ definiert Schnittstellen zwischen unterschiedlichen Produkten und gemeinsam verwendeten Komponenten. Sie legt die Grobarchitektur fest und muss Softwareentwicklungswerkzeuge freigeben. Die Mitglieder der SEAG kommen ebenfalls aus den unterschiedlichen Entwicklungsstandorten.

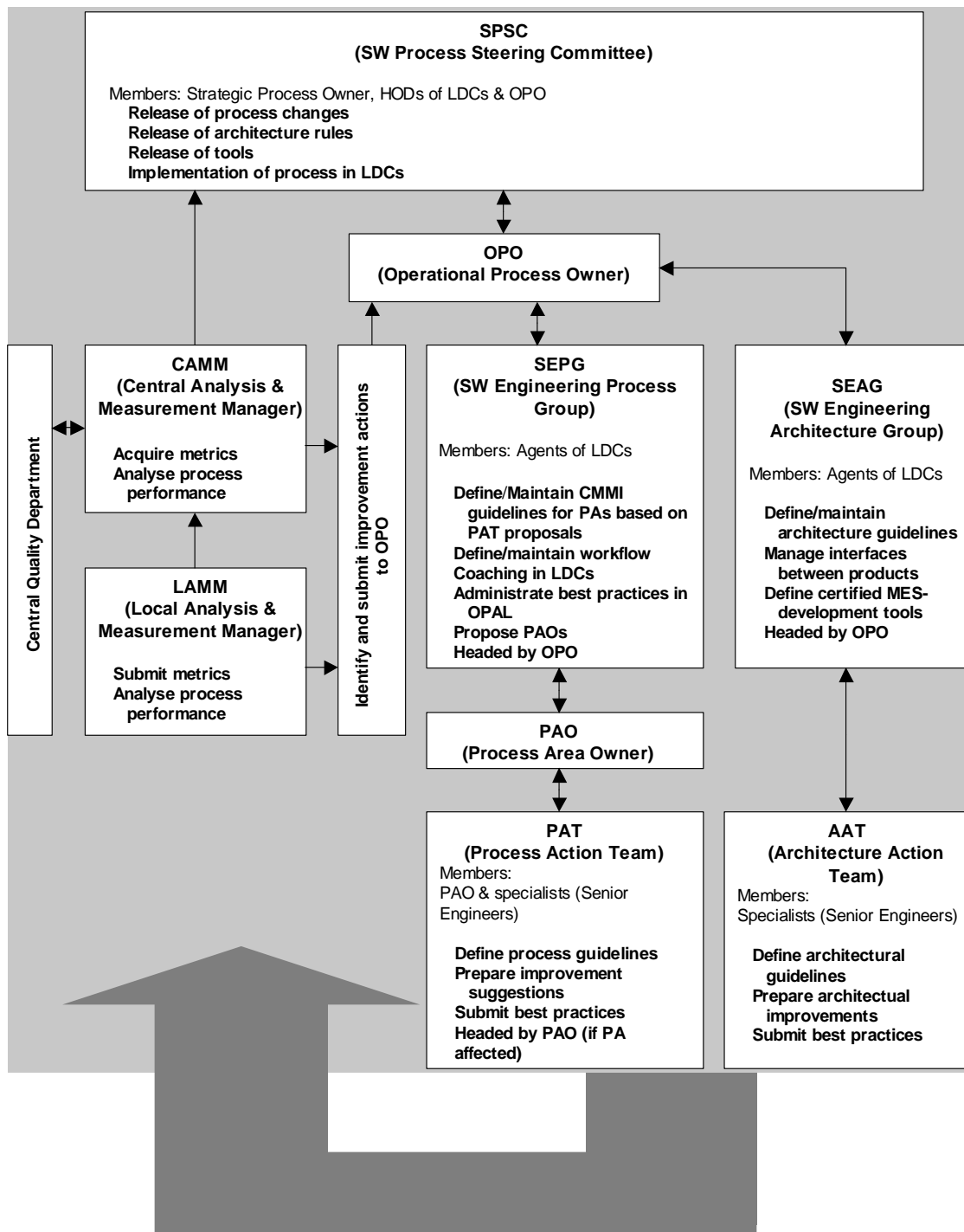


Abbildung 9.7 AVL Prozessorganisation

- CAMM** Der „Central Analysis & Measurement Manager“ sammelt das Prozess-Feedback von den unterschiedlichen Entwicklungsstandorten und berichtet an die Geschäftsführung.
- LAMM** Die „Local Analysis & Measurement Manager“ holen das Prozess-Feedback in ihrem Entwicklungsstandort ein, geben die Information an ihr Entwicklungsteam zurück und kommunizieren die Daten an den CAMM weiter.
- PAT** Die „Process Area Action Teams“ erstellen und verbessern die Regeln für einen bestimmten Tätigkeitsbereich in der Softwareentwicklung.
- AAT** Die „Architecture Action Teams“ erstellen die Interface-Beschreibungen und Spezifikationen und sind für die Auswahl bestimmter Softwareentwicklungswerkzeuge zuständig

Bei der Zusammensetzung der Teams ist unbedingt darauf zu achten, dass die Teams nicht mehr als 5 bis 7 Personen umfassen. Die Mitglieder sind die erfahrensten und initiativsten Mitarbeiter auf dem Gebiet der Softwareprozesse.

Die SEPG trifft sich zumindest alle zwei Wochen, um die Prozessimplementierung zu organisieren und Verbesserungen einzubringen. Sie wird durch kleine Teams unterstützt, die die Best Practices für die unterschiedlichen Tätigkeiten in einem erfolgreichen Softwareentwicklungsteam zusammentragen und entsprechende Richtlinien herausgeben. Es hat sich herausgestellt, dass sich die CMMI-Prozessbereiche (wie z. B. Requirements Management, Konfigurationsmanagement etc.) hervorragend für die Strukturierung dieser Teams eignen. Für jeden CMMI-Prozessbereich werden die besten Experten auf diesem Gebiet in der Organisation gesucht, die dann die PATs – „Process Area Action Teams“ darstellen. Einer davon, der Process Area Owner (PAO), leitet dieses Team. Sie erstellen für ihr Gebiet:

- Prozessregeln,
- Templates,
- eine Liste empfohlener (und vorgeschriebener) Entwicklungswerkzeuge, die mit der SEAG abgestimmt sind, sowie
- Beispiele von Entwicklungsergebnissen auf dem entsprechenden Gebiet, die in der OPAL (Operational Process Asset Library) gesammelt und veröffentlicht werden.

Bei der Umsetzung der Prozessregeln haben sich die Templates und die Beispiele von Entwicklungsergebnissen als Erfolgsfaktoren herausgestellt.

Ausreichendes Training

Nach der Erstellung der Prozessbeschreibungen, Templates und Beispielsammlungen sind professionelle Trainings für alle SoftwaremitarbeiterInnen von großer Bedeutung. Von Zeit zu Zeit wiederholt man diese Trainings für neu hinzugekommene MitarbeiterInnen.

Daneben werden Änderungen und Verbesserungen im Prozess bei regelmäßig stattfindenden „Mitarbeiternachmittagen“ vorgestellt. Als weiteres erfolgreiches Medium hat sich ein SEPG-Newsletter herausgestellt, der an alle SoftwareentwicklerInnen per E-Mail verteilt

wird. Die Newsletter bringen den SoftwareentwicklungsingenieurInnen die Softwareentwicklungsprozesse ständig in Erinnerung.

Gute Softwareentwicklungswerkzeuge

Sie helfen, Prozesse leichter einzuführen. Es hat sich herausgestellt, dass ein Schlüsselfaktor für die erfolgreiche Einführung von professionellen Softwareentwicklungsprozessen die Verwendung von Softwareentwicklungswerkzeugen darstellt. Dabei ist es wichtig, dass diese Werkzeuge gestatten, einen Arbeitsablauf abzubilden, der dem eingeführten Softwareentwicklungsprozess in der Einheit entspricht. Die Wichtigkeit der Tools wird klar, wenn man bedenkt, dass Softwareentwicklungsingenieure hochkreative Menschen sein müssen und auch sind. Sie lassen sich von vorgegebenen Richtlinien kaum beeindrucken. Wenn die Richtlinien von Werkzeugen „erzwungen“ werden, die Werkzeuge aber auch Routinetätigkeiten vereinfachen oder abnehmen, lassen sich Softwareprozesse wesentlich leichter einführen.

Folgende Werkzeuge haben sich als Grundausstattung herauskristallisiert:

- Anforderungsdatenbank
- Versions- und Konfigurationsmanagementsystem
- Compiler und Debugger
- „Daily (nightly) build“-System, das alle Source-Files nur dem Versionsmanagementsystem entnimmt und jede Nacht ein komplettes „Build“ der Projekte erstellt. Eine Auslieferung an Kunden darf nur über dieses Build-System erfolgen. Damit stellt man sicher, dass ausgelieferte Software jederzeit reproduziert werden kann – eine Grundvoraussetzung für die Wartbarkeit der Software.
- Fehlerverfolgungsdatenbank

Daneben sind weitere Werkzeuge sinnvoll, wie Testmanagementsystem, Test Sequenzer, Profiler, automatische Code-Checker etc.

Mit der Verwendung der Werkzeuge ergibt sich ein Basis-Arbeitsablauf, der von diesen Werkzeugen unterstützt und erzwungen wird und in Abbildung 9.8 (auf der nächsten Seite) beschrieben ist.

Die Einführung von Feedbackschleifen

Nach der erfolgreichen Definition und Einführung von Softwareentwicklungsprozessen müssen diese zum Leben erweckt und stabilisiert werden. Techniker verwenden für Stabilisierungsaufgaben seit Jahrhunderten so genannte Regler. Diese sind auch in der Softwareprozessverbesserung erforderlich (siehe Abbildung 9.9 auf der nächsten Seite).

Die Regelstrecke stellen die Softwareentwicklungsprojekte dar. Sie bekommen als Sollgrößen die vereinbarten Zielwerte für die Prozessdefinitionen (Regeln, Templates, Beispiele, Werkzeuge, Workflows). Die Regler sind die Verbesserungsmaßnahmen auf zwei Ebenen – der Softwareentwicklungsprojektebene und der Softwareprozessebene.

Dies bedeutet, dass auch auf diesen beiden Ebenen Istwerte vorhanden sein müssen, die für das Funktionieren von Reglern unabdingbar sind. Was sind nun diese Messgrößen und die

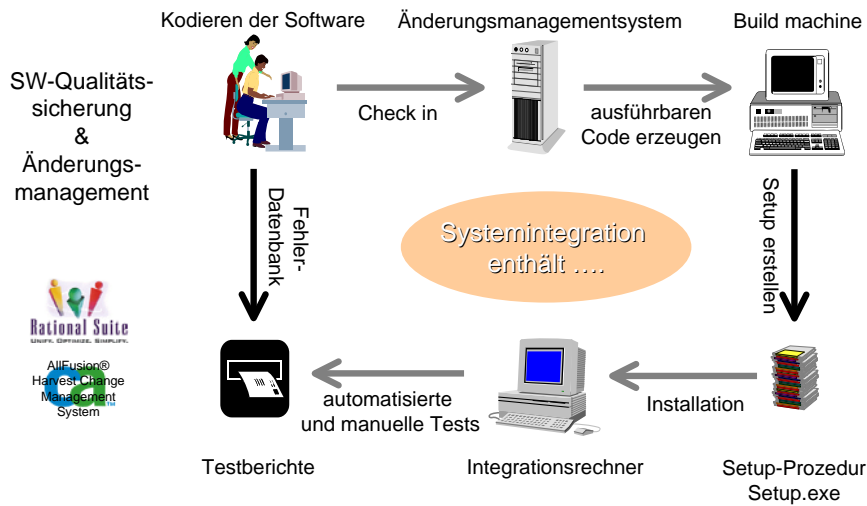


Abbildung 9.8 Umfang der Systemintegration

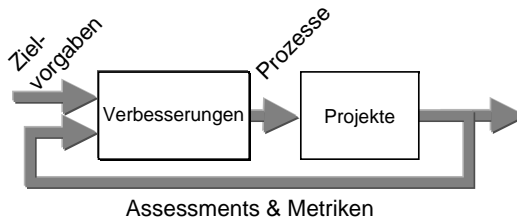


Abbildung 9.9 Verbessern mit Feedbackschleifen

dafür notwendigen Messinstrumente? Ein einfaches und sehr wirkungsvolles Messinstrument sind Projektreports mit den entsprechenden Analysen (wie z. B. „earned value trend“-Analyse, „cost trend“-Analyse oder „milestone trend“-Analyse).

Dabei ist unbedingt zu beachten, dass ein Reporting nur sinnvoll ist, wenn zwei Randbedingungen erfüllt sind:

- Die Reports müssen regelmäßig vom Management analysiert und daraus Aktionen abgeleitet werden.
- Die erfolgreiche Umsetzung der Aktionen muss überprüft werden.

Als Messgrößen für die Softwareentwicklungsprojektebene eignen sich die von CMMI geforderten Metriken hervorragend. Für die Softwareprozessebene können interne und externe Assessments durchgeführt werden, die dann zu Verbesserungsmaßnahmen führen.

Softwareprozesse allein sind nicht alles

Die Einführung und erfolgreiche Umsetzung von Softwareentwicklungsprozessen hilft, keine Tätigkeiten zu vergessen, die für eine effiziente Softwareentwicklung wichtig sind. Oft werden die Aufwände für die begleitenden Aktivitäten, wie Anforderungsmanagement,

Konfigurationsmanagement, Integration und so weiter unterschätzt, was zu der am Beginn des Beitrags beschriebenen Ineffizienz von Softwareprojekten gegen Ende der Laufzeit führt.

Abbildung 9.10 zeigt, dass die „klassischen“ Tätigkeiten, wie Software-Design, Implementierung und Modultest, nur etwa 50 % der Gesamtaufwendungen in einem Projekt darstellen. Die anderen 50 % werden oft vergessen.

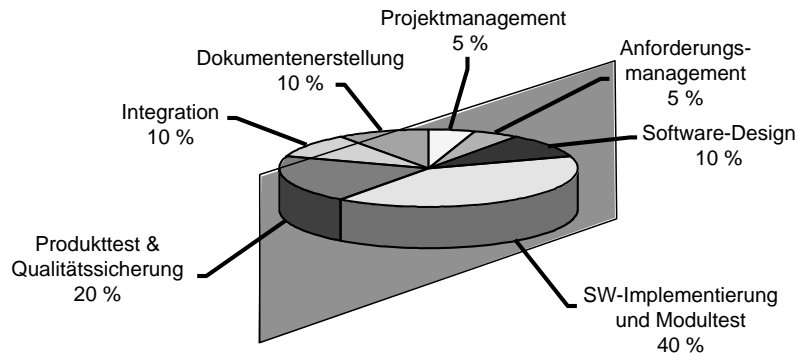


Abbildung 9.10 Prozentuale Verteilung der Tätigkeiten in einem Softwareprojekt

CMMI hilft, diese 50 % der Tätigkeiten nicht zu vergessen. Auf diese Weise ist es möglich, die drei Hauptfaktoren in einem Projekt – Budget, Termine und Funktionen – in Balance zu bringen.

Gleichzeitig darf aber nicht vergessen werden, dass es einen weiteren Erfolgsfaktor für eine Softwareentwicklungseinheit gibt, nämlich die Nachhaltigkeit (sustainability) der entwickelten Softwareergebnisse. Einer der wichtigsten Faktoren dabei ist die Erstellung einer Architektur, die die Pflege und Erweiterbarkeit der entwickelten Softwareprodukte noch Jahre später ermöglicht.

Bildlich gesprochen (Abbildung 9.11) steht ein Tisch auch auf drei Füßen: Budget, Termine, Funktionen. Ein stabiler Tisch benötigt als vierten Fuß aber noch eine tragfähige Architektur des Softwareprodukts. Die dazu notwendigen Ideen können den Konzepten der agilen Softwareentwicklung entnommen und mit den Richtlinien von CMMI zu einer professionellen Softwareentwicklungsbasis kombiniert werden.

So gibt CMMI einen effizienten Rahmen vor, um mit den immer komplexeren Softwaresystemen der Automobilindustrie – und auch anderer Bereiche – fertig zu werden.

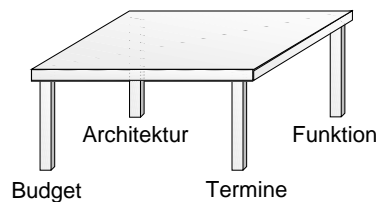


Abbildung 9.11 4-Säulenmodell

9.4.2 Erfahrungen mit einem großen CMMI-SPI-Programm

von Ph. Landucci, Credit Suisse, Zürich/Schweiz

Initiativen zur Verbesserung von Softwareentwicklungsprozessen sind anspruchsvolle, tiefgreifende und über lange Zeiträume wirkende Changemanagement-Vorhaben, die zu erheblichen Produktivitäts- und Qualitätssteigerungen und bemerkenswert hohem Return on Investment führen können. Um deren Erfolgchancen zu erhöhen, sind solche Initiativen als strategische Projekte zu führen. Als Projekt erfordern sie unter anderem ein striktes Management der damit verbundenen Risiken. Die hier aus der Praxis einer realen Initiative vorgestellten „Key Success Factors“ und die entsprechenden Implementierungsansätze zeigen auf, wie diese Risiken zu realen Chancen transformiert werden können.

Neben den „klassischen“ Herausforderungen wie Kostendruck, Time-to-Market und Technologiewandel muss sich die IT der Credit Suisse weiteren, für eine „firmeninterne IT“ neuen Herausforderungen stellen:

- Ausrichtung ihrer Prozesse nach weltweit etablierten Standards, um die Auftragsabwicklung mit einer Anzahl global operierender externer Partner zu optimieren.
- Proaktive Positionierung gegenüber externen IT-Anbietern inklusive Benchmarking nach anerkannten Evaluationsverfahren (CMMI-Appraisals).

Bei der Credit Suisse, einem der global führenden Finanzinstitute, hat Prozessverbesserung eine langjährige Tradition, die in den letzten Jahren durch eine Bündelung der verschiedenen laufenden Aktivitäten und deren Überführung in zwei strategische Programme noch mehr an Elan und Bedeutung gewonnen hat:

- Einführung einer „Lean-Sigma“-Methodik innerhalb der ganzen Bank, um die Kundenbedürfnisse systematisch ins Zentrum der Prozessinitiativen zu stellen.
- Anwendung von CMMI als Referenz-Rahmenwerk bei der IT-Entwicklung, primär in der Schweiz, doch auch zunehmend im Ausland, mit dem Ziel, den Reifegrad der entsprechenden Bereiche der IT-Organisation auf Reifestufe 3 zu bringen.

Die vorliegenden Empfehlungen in Form von „Key Success Factors“ basieren auf den Erfahrungen, welche im Rahmen der CMMI-Initiative gewonnen wurden und weiter gesammelt werden. Es sind dies:

- „Burning Platform“
- Einbezug des obersten Managements und dessen klare Unterstützung
- Klare Ausrichtung
- Messbare Ziele
- Zielorientierte, schrittweise Implementierung mit Fokus auf der Fähigkeit, zu lernen und sich zu verändern
- Kommunikation mit den „relevanten Stakeholdern“
- Prozessarchitektur
- Prozessverbesserung als Projekt mit Rechten und Pflichten
- breite Unterstützung

Die CMMI-Initiative wurde Mitte 2004 lanciert und betrifft die Neu- und Weiterentwicklungsprojekte in den Bereichen der Anwendungsentwicklung sowie der technischen Infrastruktur (Anwendungsplattformen, Middleware etc.). Involviert sind ca. 570 Projekte, 350 Projektleiter innerhalb der IT und 300 Projektleiter innerhalb der Fachabteilungen sowie ca. 1 000 IT-Entwickler und 100 Linienmanager.

„Burning Platform“

Der Bedarf an Prozessveränderung muss durch die Firma, deren IT-Abteilung und Mitarbeiter als zwingend und dringend empfunden werden. Leider liegt es in der Natur der Sache, dass ein empfindlicher „Schmerzpegel“ erreicht oder gar überschritten wird, bevor ihn eine Organisation wirklich wahrnimmt. Heutzutage kann sich eine solche „Burning Platform“ wie folgt manifestieren:

■ *für die Firma*

Die Einhaltung der gesetzlichen Vorschriften, wie Basel II oder Sarbanes-Oxley, stellt eine nicht zu unterschätzende Herausforderung dar. Die Erfüllung dieser neuen Vorschriften erfordert die Fähigkeit detaillierter Rückverfolgbarkeit und Auditierbarkeit, aufbauend u. a. auf institutionalisiertem Projektmanagement und Softwareentwicklungsprozessen. Eine angemessene Implementierung der CMMI-Anforderungen trägt zur Erreichung vieler gesetzlicher Bestimmungen in den Bereichen der Softwareentwicklung, des Konfigurationsmanagements und des Risikomanagements explizit bei.

■ *für die Nutzung der IT der Firma als Wettbewerbsvorteil*

Die Entwicklung komplexer Softwaresysteme wandelt sich in steigendem Maße zu einem entscheidenden Erfolgsfaktor für die Erreichung der Geschäftsziele. Mehr denn je müssen Unternehmen ihre Produkte immer besser, schneller und kosteneffektiver auf den Markt bringen. Die Produkte und ihre Entwicklung werden gleichzeitig immer komplexer. Man entwickelt immer mehr Komponenten an verschiedenen Standorten und fügt sie anschließend zu einem Endprodukt zusammen. Organisationen müssen somit in der Lage sein, diese komplexen Entwicklungsprozesse zu planen, zu steuern und zu überwachen. Dies erfordert einen hohen Grad an organisatorischer Reife und eine institutionalisierte, fortlaufende Verbesserung von Prozessen.

■ *für die IT-Experten*

Weil sich durch die Globalisierung auch unsere Arbeitsmodelle verändern, hängt der Marktwert von IT-Experten zunehmend von deren Qualifikation in den Bereichen des Projektmanagements, des Anforderungsmanagements, des Testens sowie der prozess- und methodenbezogenen Fähigkeiten ab. In vielen Fällen geht diese Entwicklung mit dem Sinken der Nachfrage und des Marktwerts rein technik- oder implementierungsbezogener Qualifikationen einher. Einführen von neuen Qualifikationen und Verhaltensweisen, Arbeiten mit reiferen Prozessen sowie Verinnerlichung von industriellen „Best-Practices“ in der täglichen Arbeit stellen daher ein hochwertiges fachliches Aktivum dar.

Einbezug des obersten Managements und dessen klare Unterstützung

Eine enge und intensive Zusammenarbeit mit dem obersten Management und dessen unbedingte, gegenüber der Organisation stark sichtbare Unterstützung sind zwingende Elemente des Erfolgs.

Die meisten Firmen befinden sich, explizit oder implizit, im Softwaregeschäft. Um ihr Geschäft betreiben zu können, sind sie von Software und auch von deren Qualität abhängig. Softwarequalität muss daher die höchste Priorität durch das oberste Management erhalten.

Dazu müssen Grundsätze („Policies“) als Mandat des obersten Managements in der Organisation implementiert werden. Das Management erklärt mit den Grundsätzen seine Erwartungen bezüglich der Anwendung von Best Practices in den Projekten und in der Organisation. Die Implementierung von Prozessverbesserungen in der Organisation darf in keinem Fall durch ein Projekt oder durch die Qualitätsorganisation in Auftrag gegeben, sondern muss von der höchsten Managementebene initiiert werden. Die Organisation zur Prozessverbesserung sowie die Qualitätsorganisation helfen, die Grundsätze („Policies“) zu erfüllen.

Die Einführung und regelmäßige Pflege dieser Grundsätze für die Software- und/oder Systementwicklung ist mit einem – sowohl für die Prozessverbesserungsinitiative als auch für alle Führungskräfte des Zielbereiches – erheblichen, adäquat zu planenden Aufwand verbunden. Notwendigkeit, Zweck und geplante Evolution der Grundsätze müssen projektmäßig und systematisch jedem Mitwirkenden der Organisation kommuniziert und erklärt werden. Nicht nur das oberste Management, sondern jede weitere folgende Managementstufe ist hier in der Pflicht, Verständnis und Einverständnis („Commitment“) des Einzelnen im jeweiligen Verantwortungsbereich strukturiert einzufordern.

Klare Ausrichtung

Nicht selten sind in einem größeren Unternehmen mehrere Prozessverbesserungsinitiativen zeitgleich im Gange. Steuerung und Geltungsbereiche müssen besonders klar definiert werden, um wechselseitige Einflüsse zu verstärken und Überdeckungen mit potenziell gegenläufigen Zielen zu vermeiden.

Die Grundsätze zeigen den eindeutigen Weg auf, den zielgerichtete Projektarbeiten erfordern. Die ersten Versionen der Grundsätze sollen vor allem dazu dienen, die im Unternehmen meist nur punktuell vorhandenen Best Practices zu erkennen, zu würdigen und für eine breitere Verwendung zu positionieren. Anschließend sollen die Grundsätze sukzessive genauer und umfassender auf die einzuhaltenden Vorgaben und auf die von den Standards empfohlenen Best Practices ausgerichtet werden. Der Schwerpunkt soll in erster Linie auf dem „Was“ liegen, um die Kreativität bei der Lösungsfindung des „Wie“ zu fördern, sowie um der Organisation die notwendige Zeit zu lassen, das Bedürfnis nach bzw. die Akzeptanz von organisationsweiten Standardprozessen zu entwickeln. Mit steigender Maturität der Organisation können die Grundsätze sowohl im Umfang reduziert als auch vermehrt in der „Sprache der Organisation“ abgefasst werden, da die Institutionalisierung der Standardprozesse, die alle notwendigen Aspekte des „Was“ und „Wie“ abdecken, weit genug fortgeschritten ist.

Unmittelbar nach Inkraftsetzen der ersten Version von Grundsätzen müssen interne Audits laufend kontrollieren, ob diese eingehalten werden und als Regelkreis für die Prozessverbesserung sowie für gezielte Coaching- und Weiterbildungsmaßnahmen dienen.

Messbare Ziele

Ziele zur Prozessverbesserung müssen auf allen Ebenen der Organisation definiert und verfolgt werden. Jede Managementstufe ist für das Erreichen der Ziele verantwortlich, damit sie den notwendigen Fokus auf die Prozessverbesserung behält. Die Prozessverbesserungsziele müssen deshalb in die meist vorhandenen Systeme und Instrumente, wie z. B. „Balanced Score Card“ und „Management by Objectives“, eingebaut werden.

Je nach Reifestufe einer Organisation kann die Sicherstellung der Qualität der Basisdaten mehr oder weniger anspruchsvoll ausfallen. Der Messaufwand muss daher auf die Bereiche, in denen die Verantwortung des Managements für Projekte verstärkt wird, fokussiert werden. Die Organisation mit ihren Projekten ist regelmäßig mit internen Audits daraufhin zu überprüfen, ob und wie gut sie den Grundsätzen folgt. Andere Indikatoren, wie Kosten- und/oder Zeitabweichung, sowie Anzahl der während der Projektphasen entdeckten Fehler – einschließlich betrieblicher Umgebungen – stellen zu Beginn einen guten ersten Wurf von Messgrößen dar.

Zielorientierte, schrittweise Implementierung mit Fokus auf der Fähigkeit, zu lernen und sich zu verändern

Ein Implementierungsansatz mit kleinen, schrittweisen Prozesseinführungen hat vielfältige Vorteile: Die Organisation hat mehr Zeit, um sich an die neuen Prozesse zu gewöhnen; die Korrekturrate von identifizierten Verletzungen wird mit kleinstmöglicher Verzögerung ermittelt; das Projektteam hat die Möglichkeit, auf die Probleme während der Entwicklung schnellstmöglich zu reagieren; das Potenzial kontinuierlicher Prozessverbesserungen wird in kürzester Zeit angezeigt; die Schritte sind klein, und ein einfacher, zu meisternder Veränderungsrythmus etabliert sich und wird von der Organisation erwartet.

Nach dem Aufsetzen des Verbesserungsprojekts sollten die ersten Lieferungen eine Anzahl von Elementen beinhalten, welche eine Basis für die erfolgreiche Prozessimplementierung legen, wie z. B. die organisationsweiten Grundsätze, Managementschulungen mit dem Schwerpunkt auf den Mechanismen zur Institutionalisierung der Prozesse (z. B. die generischen Praktiken des CMMI), eine gesamthafte Prozessarchitektur und das Aufsetzen einer internen Auditororganisation.

Die folgenden Prozesserweiterungen werden in „Releases“ gruppiert, deren Inhalt durch Nutzen und logische Abhängigkeiten zwischen den Prozessgebieten getrieben wird. Die Zielorganisation soll eine regelmäßige Lieferkadenz erleben, wobei z. B. alle zwei Monate ein Satz von Prozessartefakten freigegeben und institutionalisiert wird. Prozessartefakte werden, wo immer sinnvoll, durch definierte Pilotprojekte überprüft, und angemessene, zeitweise intensive Trainings müssen zur Verfügung gestellt werden.

Bestehende Lieferergebnisse sollen so wenig wie möglich verändert werden. Jedes Projekt wird damit beginnen, die neuen Prozesse ab ihrer Bekanntgabe einzusetzen. Alle Projekte werden bezüglich der Einhaltung der Grundsätze alle paar Monate überprüft. Abweichungen werden auf Listen geführt und durch Projektleiter und/oder Linienvorgesetzte vor dem nächsten Auditzyklus gelöst.

Kommunikation mit den „relevanten Stakeholdern“

Diese ist intensiv, vielschichtig und erfordert viele Kanäle. Zumindest müssen die bei einem solchen Vorhaben klassischen Fragen der Zielorganisation adressiert werden: „Warum tun wir das? Welche Schlüsselkonzepte werden angewandt? Welche Prozessveränderungen werden wir erleben? Welche Veränderungen an Rollen und Arbeit werden wir sehen? Was werden die Konsequenzen sein? Wie ist die Organisation involviert? Wann werden wir was erhalten? Wie wird der Übergang vonstatten gehen? Wie sollen wir uns vorbereiten? Was wird wann kommuniziert? Welche Schnittstellen sind zu koordinieren? Was kommt nach dem, was wir jetzt gerade anstreben?“

Prozessarchitektur

Angewendet auf Prozesse kann die Architektur als ein formales Modell mit einem Bauplan für alle Prozesselemente verstanden werden. Der definierte und/oder durch das Prozessverbesserungsprojekt revidierte Prozessbestand muss von ein und derselben Prozessarchitektur abgeleitet sein – offensichtlich wegen der Konsistenz und zum Zwecke der Anpassbarkeit („Tailoring“).

Die praktische Implementierung der Prozessarchitektur stellt typischerweise eine Prozessbestandsbibliothek dar (mit Prozessbeschreibungen, Templates, Guidelines, Checklists, etc.). Zur Veröffentlichung des Prozessbestands (wenn möglich im Kontext angemessen begrenzter Rollen und/oder Projektarten) ist die Verwendung eines webbasierten Prozessportals sehr hilfreich.

Prozessverbesserung als Projekt mit Rechten und Pflichten

Das Prozessverbesserungsprojekt ist innerhalb der Organisation unübersehbar und muss als Rollenmodell dienen. Deshalb muss es als Vorreiter die angezielten Verbesserungen bei sich selbst als Erster implementieren.

Das oberste Management muss ausdrücklich dem Projekt Rechte übergeben, wie z. B. das Recht, häufig durch die Unternehmenskanäle zu kommunizieren und die Verbindlichkeit des Managements offen zu zeigen; das Recht, organisatorische Veränderungen zu veranlassen, und das Recht, die in seinem Rahmen befindlichen Prozesse einzuführen und durchzusetzen.

Breite Unterstützung

Eine breite Unterstützung durch einen klaren Nutzen für alle Anwender und bedingt durch die personelle Besetzung des Prozessverbesserungsprojekts bildet den abschließenden Er-

folgsfaktor. Obwohl Qualitätsmodelle wie CMMI keine Wissenschaft darstellen, sondern eine bewährte Anwendung, können Anwender Zweifel und Befürchtungen bezüglich administrativer Überlastung hegen. Selbst der für alle Beteiligten offensichtliche Nutzen von mündigen Projekten (weniger „Fire-Fighting“ und Überzeit) muss intensiv und wiederholt erklärt werden. Projekte mit größter Einhaltung der Grundsätze gilt es zu belohnen und in der Organisation zu präsentieren. Schließlich kann Prozessverbesserung nicht ohne eine Kultur der Suche nach Verbesserung beginnen und Eigendynamik erhalten.

Der Einbezug großer und repräsentativer Teile der Organisation ist beim Aufbau der erforderlichen Basis für die Ausbreitung und Institutionalisierung der Prozesse von größter Wichtigkeit. Die abgebildete Organisationsstruktur (siehe Abbildung 9.12 auf der nächsten Seite) veranschaulicht, wie die folgenden Leitgedanken umgesetzt werden können:

- Die Prozessverbesserungsinitiative muss unmissverständlich vom obersten Management gewollt und getragen werden. Idealerweise soll deren Leitung direkt dem CIO (Chief Information Officer) berichten.
- Die Trennung zwischen dem „Software Process Improvement“ (SPI) und dessen Kontrolle, der Qualitätssicherung (PPQA bzw. QS), ist zwingend erforderlich. Auf diese Weise lässt sich ein Maximum an Objektivität bei den Beobachtungen bezüglich der Anwendung und auch der Anwendbarkeit der neuen Prozesse gewinnen. Dies wiederum ermöglicht das Schließen zweier eminent wichtiger Regelkreise:
 - einerseits zur kontinuierlichen Information des oberen Managements über die Maturität der Organisation und die Fähigkeit, die Prozesse auszuführen, damit Maßnahmen im Bereich des strategischen Skill-Managements, der Ausbildung und Weiterbildung getroffen werden können,
 - andererseits zur kontinuierlichen Verbesserung der Qualität und Anwendbarkeit der Artefakte („Process Assets“) aus dem SPI-Teil sowie um gezielte, kurzfristige Trainings- und Coaching-Maßnahmen zu ergreifen.
- Die Prozessverbesserungsinitiative ist aus vielen Gründen, nicht zuletzt „per Design“, exponiert. Daher muss sie mit fragten Spezialisten ihres Wissensgebiets besetzt werden, welche gleichzeitig den „Bodenkontakt“ nicht verlieren dürfen. Eine Mischung von Projektkräften (Kernteam) und Spezialisten aus der Zielorganisation (erweiterte Teams) ist ein wirksames Rezept gegen das „Elfenbeinturm-Syndrom“ und vereinfacht die Definition, Akzeptanz, Schulung und Durchsetzung der Prozesse in erheblichem Maße. Diese Ergänzung vom Kernteam durch erweiterte Teams gilt sowohl für den SPI- als auch für den PPQA-Teil der Organisation.
- Im SPI-Bereich ist es sehr hilfreich, auf eine bestehende Prozessorganisation zurückgreifen zu können, sonst bietet sich eben die Gelegenheit, diese aufzubauen. Für jedes Prozessgebiet soll ein Mitglied des oberen Managements, das aufgrund seines Verantwortungsbereiches imperatives Interesse an der Weiterentwicklung und Leistung der betroffenen Prozesse hat, die Rolle des Prozessverantwortlichen übernehmen und ein Prozessteam aus Expertenteams, Projekten und anderen Initiativen rekrutieren. Die personelle Zusammensetzung und der Grad der Beschäftigung hinsichtlich der Prozessarbeit

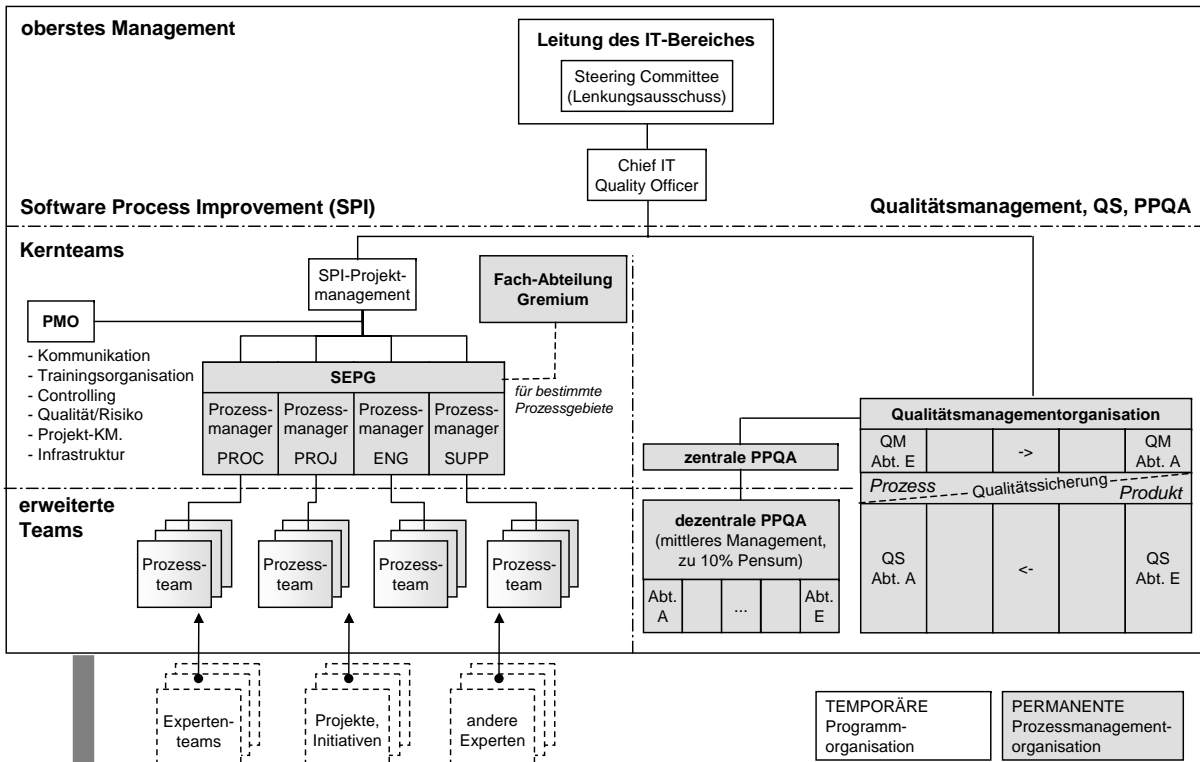


Abbildung 9.12 Organisationsstruktur für ein CMMI-Programm

können dabei variabel sein, das Mandat hingegen muss permanent sein. Die Prozessmanager des Kernteams unterstützen die Prozessteams mit der nötigen Expertise und zu leistenden Konzentration auf die Prozessarbeit. Darüber hinaus stellen sie sicher, dass die Schnittstellen zwischen Prozessgebieten adäquat implementiert und die Vorgaben der Prozessarchitektur eingehalten werden.

- Bei der PPQA lohnt es sich aus vielen Gründen, die Mitglieder des mittleren Managements in eine sehr aktive Rolle zu setzen, und zwar mit regelmäßigen Einsätzen als Auditoren bzw. Assessoren (z. B. zwei Tage pro Monat) in anderen Organisationsbereichen als denjenigen, die sie verantworten. Auf diese Weise stellt man sicher, dass die Grundsätze wirklich ernst genommen werden und vor allem, dass das mittlere Management im Rahmen der Auditorausbildung und -praxis „Best Practices“ kennen lernt, deren Anwendung bei zahlreichen Projekten und Bereichen der Organisation kontrolliert und vor allem im eigenen Verantwortungsbereich sicherstellt. Die zentrale PPQA hat bei diesem Modell primär die Aufgabe, die Audits zu planen, die Berichte zu verfassen und die zeitgerechte Behebung der Abweichungen zu verfolgen.

Zuletzt darf die projektmäßige Durchführung einer Initiative zur Prozessverbesserung nicht den Eindruck erwecken, dass die Investitionen und Bemühungen einmalig bzw. temporärer Natur sind. In der Tat hat das „Projekt“ vor allem den Zweck des Erlangens eines

höheren Fähigkeitsgrads in einem oder mehreren Prozessgebieten und muss sich dabei auf eine starke, etablierte Qualitäts- (und Prozessmanagement-)Organisation abstützen können. Nach Abschluss des Projekts ist es genau diese Qualitäts- (und Prozessmanagement-)Organisation, die den Hauptbeitrag zur kontinuierlichen Einhaltung und Pflege des Erreichten sichert und somit den langfristigen Nutzen einer solchen Initiative auch tatsächlich realisiert.

9.4.3 Erfahrungen und Nutzen bei der Umsetzung von CMMI bei Alcatel

von Ch. Ebert, Alcatel, Paris/F

Das integrierte Capability Maturity Model (CMMI) ist heute ein grundlegender und weltweit geforderter De-facto-Standard der Produktentwicklung. Es ist seit den achtziger Jahren des vergangenen Jahrhunderts an der Carnegie Mellon University in Pittsburgh entstanden, wo das Software Engineering Institute (SEI) gegründet wurde, um die Qualität und Verlässlichkeit der Softwareentwicklung zu verbessern. Die Autoren des CMMI erkannten, dass nur die Zusammenstellung einzelner „bester Praktiken“ keine Lösung verspricht. Man muss den gesamten Lebenszyklus von der Konzeption bis zur Lieferung betrachten, um anhaltende Verbesserungen zu erreichen. Die Elemente (sogenannte Prozessbereiche) des CMMI sind einzeln einführbar, passen wie ein Puzzle zusammen und sind unabhängig von der eingesetzten Methodik, von der Größe und Art des Unternehmens und von den spezifischen Anwendungen, IT-Systemen, Produkten oder Lösungen, die entwickelt werden. Die Anwendung des CMMI geht daher inzwischen weit über die Softwareentwicklung hinaus.

CMMI passt sehr gut zu den relevanten, die Software- und Systementwicklung bestimmenden ISO-Standards. Es ist kompatibel zu ISO 15504 (Standard for Information Technology – Software Process Assessment) für Assessments und Modelle der Prozessverbesserung, ISO 9001:2000 (Standard for Quality Management Systems – Requirements) sowie ISO 12207 (Standard for Software Life Cycle Processes) und unterstützt marktspezifische Standards, wie beispielsweise TL 9000 in der Telekommunikationsindustrie. CMMI läuft konform zu berufs- und aufgabenspezifischen Anforderungen wie das PMBOK des Project Management Institutes und ist nicht auf die Softwaretechnik beschränkt, sondern wurde für das komplette Systems Engineering auf eine wesentlich breitere Anwendungsbasis gestellt. So ist es heute nicht nur in der IT, sondern auch im Flugzeug- oder Automobilbau weit verbreitet.

Alcatel ist einer der weltweit führenden Lieferanten und Systemintegratoren von Lösungen für Netzbetreiber im Telekommunikationsbereich, Zugangs- und Servicelieferanten des Internets und Unternehmen, die Sprach-, Daten- oder Videodienste selbst einsetzen oder anderen anbieten. Mit einem Umsatz von 12,3 Milliarden Euro im Jahre 2004 und 56 000 Mitarbeitern, davon ein Viertel Ingenieure, ist Alcatel in 130 Ländern aktiv. Die Herausforderungen in diesem Geschäft sind langlebige, hochgradig verteilte Systeme, die aus

Hardware- und Softwarekomponenten – oft von verschiedenen Herstellern – bestehen. Diese Systeme müssen eine sehr hohe Zuverlässigkeit besitzen, was nicht nur hohe Qualitäts- und Verfügbarkeitsanforderungen stellt, sondern auch eine einfache und anhaltende Wartbarkeit voraussetzt. Schließlich müssen die Systeme ständig erweitert und erneuert werden, beispielsweise in den achtziger und neunziger Jahren für die digitale Telefonie und den Mobilfunk und heute für überall verfügbare Breitbandlösungen. Die eingesetzten Systeme verbinden Echtzeitanforderungen mit denen eingebetteter Systeme und verlangen einen betont systemischen Ansatz, um die Qualität – von den Komponenten bis zur installierten Lösung – nachweisbar zu gewährleisten.

Die Qualität der Produkte und Lösungen wird durch die Qualität der zugrunde liegenden Prozesse bestimmt. Diesen Gedanken verfolgt Alcatel schon seit Mitte der neunziger Jahre in dem Bestreben, die Prozesse mit dem Capability Maturity Model (CMM) zu verbessern. CMM wurde schon damals unternehmensweit eingesetzt, um die Qualität der Produkte zu verbessern. Dazu wurde es frühzeitig ein wenig „erweitert“, um auch für Hardware- und Systementwicklung tauglich zu sein. Dies war im Falle des CMM nicht besonders schwer, da nichts weiter zu tun war, als bei Hardwaretätigkeiten auf den Zusatz des Worts „Software“ zu verzichten. Die Methodik ist gleichermaßen anwendbar, egal, ob es um Projektmanagement oder Prozessverbesserung geht. War es anfangs primär die Verringerung der Fehler im Feld und somit eine erhöhte Zuverlässigkeit der gelieferten Systeme, haben sich diese Geschäftsziele je nach Marktsituation und Produktlinie in den vergangenen Jahren an geänderte Bedürfnisse angepasst. Beispielsweise wurde Mitte der neunziger Jahre die Lieferqualität von Vermittlungssystemen von Alcatel jährlich um mehr als 20 Prozent verbessert – eine Folge des Einsatzes qualitätssichernder Maßnahmen und früher Fehlerentdeckung, wie sie durch das CMM propagiert werden. Ende der neunziger Jahre wurde im gleichen Produktbereich die Zykluszeit innerhalb von nur drei Jahren auf ein Drittel reduziert, was ohne eine solide Basis von Prozessen auf Reifegrad 3 nicht möglich gewesen wäre. Schon seit vielen Jahren entwickelt Alcatel seine Lösungen weltweit mit heute über zwanzig Entwicklungszentren in allen Regionen der Erde, was ohne solide und gut kommunizierte Entwicklungsprozesse nicht möglich wäre.

Eine Zusammenfassung dieser frühen Erfahrungen mit dem CMM bei Alcatel ist in [Eber97] und [Eber02] beschrieben. Mit dem CMM konnten wir die Verlässlichkeit unserer Projekte verbessern, deren Qualität erhöhen und gezielte Prozessverbesserungen bedarfsorientiert implementieren. Es half, unsere Prozesse zu verstehen und zu verbessern und ermöglichte ein kontinuierliches Benchmarking, sowohl im Konzern als auch mit anderen Unternehmen. So konnten wir mit verschiedenen Wettbewerbern bereits in den neunziger Jahren Erfahrungen zu Produktlinien austauschen und dann erfolgreich implementieren [Eber03]. Dedizierte Verbesserungsprojekte mit vielen detaillierten Erfahrungen sind in [Eber04] beschrieben.

CMM wurde dabei immer als Unterstützung und Werkzeug zur Erreichung der Geschäftsziele von Alcatel und somit der Verbesserung der Kundenzufriedenheit und nicht als Selbstzweck verstanden. Es ist ein großer Fehler, den wir teilweise auch im eigenen Unternehmen beobachten konnten, wenn das „Zertifikat“ (das es formal als solches gar nicht

gibt) zum Ziel wird. Sicherlich verführt die einfache Messbarkeit des Reifegrads, verbunden mit der determinierten Vorgehensweise, zur Prozessverbesserung – die man sich als Consulting-Leistung leicht einkaufen kann – dazu, sich ein solches Ziel zu setzen. Dann versuchen Manager allerdings nur noch, dieses Ziel zu erreichen, ohne sich darüber Gedanken zu machen, ob die Änderungen anhaltend sind und ob sie auch einen Nutzen über den Reifegrad hinaus erbringen. In unseren Augen ist ein solches Verhalten hochgradig unprofessionell, weil auf diese Weise begrenzte Ressourcen und auch die Mitarbeitermotivation ohne Geschäftswert verbrannt werden.

Ist Prozessverbesserung nach einer Dekade von guten Ergebnissen mit CMM überhaupt noch ein Thema? Die Antwort bei Alcatel ist ein klares „Ja“. Märkte und Kundenbedürfnisse ändern sich heute schneller als je zuvor. Die Unternehmensergebnisse müssen sich ständig verbessern. Begrenzte Forschungs- und Entwicklungsressourcen müssen optimal genutzt, Termine, Budgets und die vereinbarte Qualität verlässlich eingehalten werden. Das Produktportfolio erneuert sich schneller denn je und verlangt nach darauf abgestimmten, innovativen Prozessen für das Produktmanagement, den Lebenszyklus und das Requirements Management. Ein zunehmender, weltweiter Wettbewerbsdruck und die rasche Kommodisierung von Produkten, wenn sie einmal am Markt sind, erfordern eine gut synchronisierte, weltweite Entwicklung von Produkten, Lösungen und Dienstleistungen. Die entsprechenden Rezepte sind bekannt und weit verbreitet. Das haben wir vom CMM gelernt. Die Zutaten sind ebenfalls vorhanden: hervorragende Ingenieure, Entwicklungswerkzeuge, Methodik und Prozesse. Kommt es dennoch zu schlechten Ergebnissen, ist der Grund fast immer ein – vermeidbarer – Managementfehler. KPMG fasste diese Beobachtungen kürzlich wie folgt zusammen: „Nur durch eine Entwicklung, die nach klaren und einheitlichen Prozessen sowie Zielen und Abbruchkriterien handelt und durch ein ganzheitliches Projekt-Controlling gesteuert wird, können die Hersteller den Herausforderungen begegnen.“

Weshalb CMMI? Wie bereits dargelegt, hatten wir bei Alcatel schon früh einige Schwächen des SW-CMM erkannt und darauf reagiert. Das klassische CMM betrachtete einseitig die Softwareentwicklung und gab keine Hilfestellung bei Schnittstellen jenseits der Entwicklungsarbeiten. CMMI hat hingegen den Vorteil, alle Bereiche der Produktentwicklung gleichermaßen zu verbessern. Es kann für reine Softwareanwendungen eingesetzt werden, aber auch für komplexe hierarchische Systeme, wie sie im Telekommunikationsbereich vorkommen. Der Begriff des Lebenszyklusmanagements, zurzeit in aller Munde, wurde in der Konzeption des CMMI bereits vorweggenommen. Die Stärken des CMMI liegen aus unserer Sicht vor allem in der umfassenden Behandlung der „frühen Phasen“ des Lebenszyklus, beispielsweise der Konzeption, der Anforderungsentwicklung und der Verfolgung von Anforderungen bis zur gelieferten Lösung [Eber05]. Schnittstellen werden nicht abstrakt wie beim SW-CMM als Prozessbereich behandelt, sondern explizit in Bezug auf Erwartungen und Verhaltensweisen beschrieben. Das Lieferantenmanagement nimmt im CMMI eine dominierende Rolle ein und trägt damit der arbeitsteiligen und weltweiten Entwicklung von Lösungen mit zunehmendem Integrationscharakter Rechnung [Eber05b]. Gleichermaßen fordert CMMI solide Prozesse für jegliche externe Dienstleistungen und

Komponenten, egal ob es sich um eine kundenspezifische Softwarelösung handelt oder um eine Open-Source-Komponente, die im Internet gefunden und im Produkt eingesetzt wird. Prozesse wie das Risikomanagement werden vom CMMI im Gegensatz zum SW-CMM explizit gefordert. Schließlich hat CMMI eine gegenüber dem SW-CMM verbesserte Struktur, die für jeden Prozessbereich klare Vorgaben an die Institutionalisierung stellt, die dann in die Sprache des Prozessbereichs „übersetzt“ werden. Die gleichen Inhalte des CMMI werden kontinuierlich und gestuft dargestellt (also Reifegradebenen), was das Implementieren beträchtlich erleichtert.

Der Umstieg zum CMMI erfolgte bei Alcatel als weltweites Projekt mit allen Geschäftsbereichen. Zunächst wurden Erfahrungen aus anderen Unternehmen gesammelt und verglichen. Schließlich ergab es keinen Sinn, das Versuchskaninchen zu spielen. Das Geschäft geht vor, und ein solches neues Modell wird dann eingeführt, wenn man sicher ist, dass es funktioniert. Nach dieser Evaluierung liefen ein Jahr lang diverse Pilotprojekte, bei denen die Implementierung des CMMI im Detail untersucht wurde. Dabei ging es primär um Planungsdaten und Kostenschätzungen. Gleichzeitig wurde ein unternehmensweites internes Training aufgebaut, in dem zunächst eine kurze CMMI-Schulung mit Benchmarkdaten und Einflussanalyse auf relativ breiter Basis vermittelt wurde (Ingenieure und Management). Später kam es zu dedizierten Schulungen von CMMI und SCAMPI-Einführungen nach den Standards des SEI. Aus den Piloten konnten wir eine interne Handlungsanweisung ableiten, die zusätzlich zu den CMMI- und SCAMPI-Standards als Umsetzungsrichtlinie für eine weltweite Konsistenz sorgt. Beispielsweise wurden Werkzeuge für das PIID und die Bewertung von Appraisals ausgewählt und eingeführt. Auch konnten einzelne Elemente des PIID bereits vorpopuliert werden, denn viele Prozesse und Richtlinien bei Alcatel sind unternehmensweit verbindlich und werden durch ein konzernweites Prozess- und Portfoliomanagement unterstützt. Damit lassen sich die Kosten bei der Einführung des PIID in den ersten Appraisals einer Produktlinie beträchtlich reduzieren. Auf dieser Basis wurde schließlich ein Umstiegsplan vorgestellt, bewertet und ausgeführt. Die Einführung des CMMI in einem großen Unternehmen wie Alcatel dauert einige Jahre, ist allerdings auf einer solchen Basis vorhersehbar, effektiv und bewirkt keine unnötigen Störungen.

Zum Schluss noch einige praktische Tipps für das Änderungsmanagement mit CMMI:

- Setzen Sie klare Ziele auf Quartalsbasis. Vermeiden Sie zu viele parallele Initiativen. Erzielen Sie zuerst ein Ergebnis, dann das nächste, usw.
- Setzen Sie um, was Sie sagen. Ihre Mitarbeiter und Kollegen beobachten IHR Verhalten genau. SIE müssen die Veränderung sein, die Sie im Unternehmen sehen wollen!
- Personalisieren Sie Änderungen. Führen Sie Änderungen immer als Projekt durch (d. h. Projektleiter, Zielvorgaben, Projekt-Controlling etc.). Wenn die Verbesserungsziele nicht erreicht werden, muss dies Konsequenzen nach sich ziehen.
- Kombinieren Sie beim CMMI die kontinuierliche mit der gestuften Repräsentation. Nutzen Sie das kontinuierliche Vorgehen für Ihr Verbesserungsprojekt und die Reifegradebenen beim Appraisal und für das Benchmarking.

- Messen Sie frühzeitig und kontinuierlich. Steuern Sie gegen, bevor die Scorecard rot wird. Fokussieren Sie auf den ROI. Sind Ihre Projekte besser geworden? Setzen Sie Leistungskennzahlen und Kunden-Feedback ein.
- Verbesserungen sind nicht umsonst. Je nach Geschwindigkeit und Umfang kosten Prozessverbesserungen ein bis drei Prozent des Forschungs- und Entwicklungsbudgets. Dieses Investment sollte sich vom ersten Jahr an auszahlen.
- Seien Sie nie zufrieden. Niemals! Wettbewerber wachsen ständig nach. Bauen Sie die Fallen für die anderen, oder Sie tappen blindlings hinein.

Die Ergebnisse sind vielfältig. Daher wollen wir zum Abschluss in Abbildung 9.13 die Auswirkungen eines höheren Reifegrads auf die Verlässlichkeit von Projektzielen darstellen.

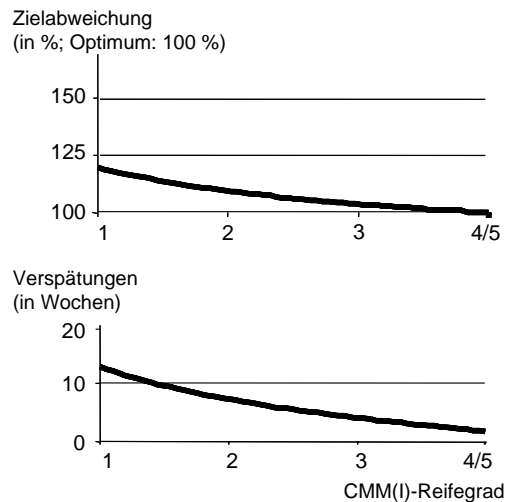


Abbildung 9.13 Ergebnisse in Abhängigkeit vom CMM(I)-Reifegrad

Wir konnten deutlich belegen, dass sich Prozessverbesserungen in der Produktentwicklung positiv auf das Geschäftsergebnis auswirken, und einen statistisch signifikanten Einfluss auf die Qualität und Liefertreue messen. Unsere Mitarbeiter bemerkten, dass aus der anfänglich wahrgenommenen Belastung eine verlässlichere und produktivere Umgebung entstand.

9.4.4 ISO 15504 als Landkarte zum eigenen Prozessmodell

von D. Assmann, Valeo Schalter & Sensoren GmbH, Bietigheim-Bissingen/Deutschland

9.4.4.1 Kontext

Die Valeo Schalter und Sensoren GmbH (VSDC) ist eine eigenständige, in die Valeo Unternehmensgruppe integrierte Einheit. Mit einem Entwicklungszentrum und zwei modernen Werken gehören wir in den Bereichen Umfelderkennungssysteme (z. B. Ultraschall-

parkhilfe) und Lenksäulenmodule zu den weltweit führenden Zulieferern in der Automobilindustrie.

Durch die Einbettung in den Valeo Konzern gibt es für die Produktentwicklung übergreifende Standards und Vorgehensmodelle. Da viele unserer Produkte (z. B. Einparkhilfe, Lenkwinkelsensor, Spurerkennungssysteme) softwareintensiv sind, benötigen wir eine produktspezifische Verfeinerung der konzernweiten Vorgehensmodelle (das Prozessmodell).

Wie in der Automobilindustrie üblich, sind unsere Softwareanteile langsam gewachsen. Ursprünglich kamen die Produkte aus der Mechanik- und Elektronikentwicklung. In diesen Bereichen wurden Funktionen durch programmierbare Bausteine vereinfacht. Diese „Vereinfachung“ hat sich verselbstständigt: Heute werden die wesentlichen Funktionen in Software realisiert, Mechanik und Hardware sind primär Ausführungsumgebung und Schnittstelle. Somit ist auch die Softwareentwicklung von Maschinenbauern oder Elektrotechnikern vorangetrieben worden und nicht von Personen mit Software-Engineering-Hintergrund.

Als Unternehmen mit softwareintensiven Produkten brauchen wir für unsere Entwicklung adäquate Methoden, Techniken und Werkzeuge (M/T/W). Die konkrete Auswahl und Anwendung geeigneter M/T/W ist im Prozessmodell beschrieben.

9.4.4.2 Prozessverbesserung

Vorgeschichte

In der kurzen Geschichte der Softwareentwicklung bei VSDC (1991 Entwicklung der ersten Ultraschallparkhilfe für den Markt, 2005 Weltmarktführung) gab es verschiedene Initiativen zur Verbesserung der Softwareentwicklungsprozesse. Allerdings war keine auf Dauer erfolgreich. Die Gründe dafür sind dieselben wie die aus der Literatur von anderen Branchen bekannten. Im Folgenden einige beobachtete Defizite:

- Prozessaufbau/Prozessverbesserung fand ausschließlich durch externe Berater statt und hatte keinen organisatorischen Rückhalt.
- Prozessbeschreibungen, Templates und Infrastruktur wurden bereitgestellt, aber die Anwendung der zugrunde liegenden M/T/W wurde nicht geschult.
- Die Wahrnehmung im Management war sehr niedrig und das Verständnis für die Besonderheiten von Software fehlte auf allen Ebenen.
- Die definierten Prozesse waren nicht vollständig/durchgängig; besonders Schnittstellen zu Firmen- und Konzernstandards wurden vernachlässigt.
- Keine/wenig Software-Engineering-Expertise war in der Organisation vorhanden.

Start SPI

Was war der Auslöser für SPI?

Die externe Motivation kam durch nicht zufriedenstellende ISO-15504-Assessments. Zudem betonten unsere Kunden immer stärker die wachsende Komplexität der Software und die damit verbundenen zunehmenden Risiken.

Die wachsende Komplexität war auch die interne Motivation; sie zeigte sich in einer steigenden Zahl von Fehlern, die erst in späten Phasen gefunden und damit teurer wurden.

Was waren die Planungsvariablen?

Da schon Geld und Zeit in andere SPI-Maßnahmen investiert wurden, war der Zeitplan ambitioniert. Die Zielvorgabe durch Kunden und Management war ISO 15504 Level 2 nach einem Jahr und Level 3 nach einem weiteren. Im Gegensatz zu den vorherigen Ansätzen sollte diesmal die Prozessverbesserung intern getrieben werden. Zudem sollten die wesentlichen Randbedingungen¹ berücksichtigt werden.



Abbildung 9.14 Modell der kontinuierlichen Verbesserung mit einmaligem Start ([Demi82], [Basi94])

SPI-Projekt

Unser Management rief daraufhin das „SPI-Projekt“ ins Leben. Aufgabe war die Definition angemessener Prozesse und deren Pilotierung sowie Information und Beratung des Managements zu Softwarethemen (entspricht Define/Pilot/Check in Abbildung 9.14).

Was wollten die Stakeholder bzw. das Management?

Da kein Stakeholder mit der Prozessbewertungslogik der ISO 15504 (SPICE) vertraut war, bestand der erste Schritt darin, ein klares, gemeinsames Verständnis zu schaffen. Dazu wurde das Ziel (Erreichen von Level 2 und Level 3) in die Bereiche Schaffung des Rahmens (Prozesse), Etablierung der Infrastruktur und breite Anwendung in den Projekten gegliedert und der Umfang auf 21 ausgewählte Prozesse festgelegt.

¹ Als wesentliche Randbedingungen sind zu nennen: Konzernvorgaben, Wünsche diverser Kunden, Berücksichtigung existierender Standards, Nutzung von bewährtem/etabliertem Material, Benutzerfreundlichkeit bei Erstellung und Nutzung, Berücksichtigung der verschiedenen Produktlinien, Wartbarkeit für künftige Erweiterungen.

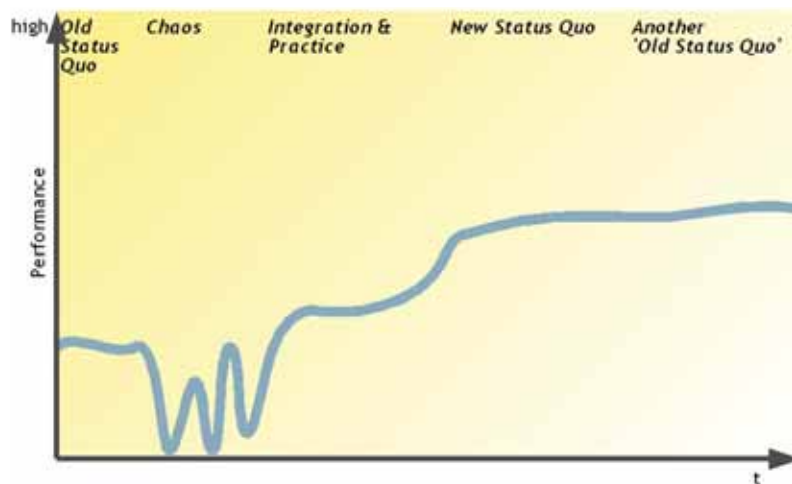


Abbildung 9.15 Modell des Änderungszyklus nach Virginia Satir [Sati91]

Damit stellten wir uns drei Herausforderungen, die jeweils dem Änderungszyklus (gemäß dem Modell in Abbildung 9.15) unterlagen.

- Know-how: durch Definition und Bereitstellung von Prozessen
- Technologie: durch Schaffung einer adäquaten Infrastruktur
- Kompetenz: durch Lernen und Umdenken während der Roll-Out-Phase in die Projekte und Abteilungen; der Roll-Out berührt die Unternehmenskultur am stärksten, wobei besonders das Management als Vorbild gefordert ist.

Was waren die Kriterien für die Prozessauswahl?

Die Prozessauswahl war zu Beginn kundengetrieben bzw. über den HIS-Umfang (Herstellereinitiative Software) gegeben. Einzelne Ausprägungen wurden intern definiert.

Im Rahmen der Erarbeitung der Modelle fiel auf, dass der HIS-Umfang im mathematischen Sinn nicht abgeschlossen ist, d. h. es fehlten Prozesse, um intern eine funktionierende prozessbasierte Organisation zu schaffen. Dies findet nun nach und nach statt.

Wie sah das Projektteam aus?

Das Team setzte sich aus einem Projektleiter, dem Prozesseigner (Process Owner) und einem Coach zusammen. Der Begriff Process Owner beschreibt in etwa die Rolle eines Paten für einen oder mehrere Prozesse. Bei uns gab es eine Doppelbesetzung in der Form, dass der Projektleiter gleichzeitig Process Owner war.

Die Aufgabe des Projektleiters war, den notwendigen Software-Engineering-Hintergrund zu vermitteln, die Planung und Durchführung des Projektes zu treiben, den ISO-15504-Standard zu schulen, Grundlagen der Prozessmodellierung zu vermitteln, beim Modellieren und Dokumentieren der Prozesse zu unterstützen und die Schnittstelle zu den Stakeholdern zu bilden.

Die Process Owner waren Experten aus der Organisation, die aber keine Prozess Erfahrung besaßen. Ihre Aufgabe war es, den ihnen zugewiesenen Prozess zu beschreiben und unterstützendes Material bereitzustellen.

Der Coach steuerte Erfahrung in den einzelnen Prozessen und zu Abläufen der Verbesserung bei.

Wie war die Arbeitsweise?

Da die Process Owner intensiv in das operative Geschäft eingebunden waren, fanden die SPI-Arbeiten in Form externer Workshops statt. Diese Workshops setzten sich aus Diskussion, Training, Modellierung und gegenseitigen Reviews zusammen.

Nach 17 Workshops in 18 Monaten stand ein Intranetportal bereit, das graphische Prozessbeschreibungen und unterstützendes Material sowie Schulungen enthielt.

Eine Vorversion des Prozessportals wurde für zwei Pilotprojekte veröffentlicht, um frühes Feedback von den späteren Nutzern zu bekommen. Zudem standen die Process Owner den Piloten als Coaches zur Verfügung.

In den Abteilungen fand parallel ein erster Aufbau der zusätzlichen Infrastruktur (primär Werkzeuge) statt, der mit dem SPI-Team abgestimmt war, aber durch die Abteilungen getrieben wurde.

Über die Laufzeit des SPI-Projekts gab es monatlich eine Statuspräsentation für das Management zur Darstellung von Status, Risiken und Ausblicken und regelmäßige Kommunikation zur SEPG.

Zur Zeit des offiziellen Release-Termins wurde die Kommunikation intensiviert. In den Abteilungen fanden entsprechende Trainings statt, Fachvorträge wurden angeboten und bewährte Medien (wie interne Newsletter) genutzt.

Und wo bleibt ISO 15504?

Wir haben während der Modellierungsarbeit verschiedene Standards betrachtet, wie z. B. CMMI, IEEE, V-Modell XT und kundenspezifische Modelle. Unsere Richtschnur war allerdings immer der Standard ISO 15504, zuerst noch als technischer Report, später als internationaler Standard und in Form von AutomotiveSPICE.

Bei jeder Verbesserungsinitiative muss man sich für ein Lebenszyklusmodell entscheiden. Dafür kann man bestehende Modelle verwenden oder ein eigenes definieren. Auf den Standard ISO 15504 (basierend auf ISO 12207) fiel unsere Wahl, weil uns Kunden nach diesem Modell bewerten und dieser Standard aufgrund seines hohen Abstraktionsgrads leichter zu erweitern ist als ausformulierte Modelle wie das CMMI. Für unsere Prozesse bildete ISO 12207/ISO 15504 den Rahmen, in den wir nach Bedarf weitere Elemente und Verfeinerungen integrierten.

Die größte Herausforderung bestand darin, keine 1:1-Kopie zu erstellen und die Prozesse, speziell für höhere Reifegrade, so zu definieren, dass die Level für die Nutzer transparent sind. Wichtig ist, dass die Anwender die Prozesse ohne Kenntnis und Verständnis des Standards nutzen können.

SPI₂-Projekt

Software Process Improvement Implementation (SPI₂) ist das Folgeprojekt, das den Roll-Out koordiniert (entspricht Roll-Out/Analyze in Abbildung 9.14). Hier wird ein eigenständiges Projekt aufgesetzt, das den Fokus der Wandlung der Arbeitskultur durch Schulung hat und somit auch andere Kompetenz und Befugnisse braucht. Basierend auf dem Änderungsmodell (Abbildung 9.15), soll das Projekt durch die Chaos- und Aufbauphase führen.

In diesem Rahmen werden wir verstärkt externe Unterstützung nutzen, da temporärer Zusatzbedarf abgedeckt werden soll. Gezielten Einsatz von Experten sehen wir in fünf Bereichen:

- Projekte von Mehrarbeit in der Umstellungsphase entlasten
- Abteilungen beim Aufbau der notwendigen Infrastruktur unterstützen
- Infrastruktur (Werkzeuge) auswählen, bereitstellen und schulen
- Schulung von methodischen Grundlagen (die Schulungen werden in das Prozessmodell integriert)
- Coaching der Prozesse, um Verständnis und Akzeptanz für die Prozesse zu schaffen

9.4.4.3 SPI-Lektionen

Einige Lektionen greifen wir auf (mit „wir“ spreche ich jetzt nur noch die Prozessverbesserer in VSDC an) – ob wir sie wirklich gelernt haben, wird die Zeit zeigen:

- Wiederholung:
Wiederholung ist der Schlüssel zum Erfolg. Wie eine tibetanische Gebetsmühle sollte man nicht müde werden, einfache und komplizierte Zusammenhänge immer wieder zu wiederholen. Die Annahme, nach zwei- oder dreimaliger Erklärung sei ein Sachverhalt verstanden worden, kann verhängnisvoll sein. Die Ursache ist nicht Desinteresse, sondern die Tatsache, dass unsere Gegenüber sich die meiste Zeit mit ganz anderen Themen beschäftigen. Zudem haben wir so die Möglichkeit, das Denken unserer Stakeholder und Nutzer zu formen.
- Widerstand:
Widerstand ist unangenehm, aber ein gutes Zeichen. Die Menschen beschäftigen sich mit dem Thema und bringen ihre Gefühle und Erfahrungen zum Ausdruck. Deswegen müssen wir mit Widerstand vorsichtig umgehen. Weisheit ist nötig, um die Frage hinter der Frage zu sehen und zu beantworten.
- Erwartungen:
Wir selbst müssen realistische Erwartungen an SPI haben und schaffen. Der Prozess gibt nur einen Rahmen vor, der Inhalt ist der Mensch. Erfahrung und Methodenkompetenz können nicht in einen Prozess gepackt werden.
- Menschen:
Wir müssen unser Team sorgsam behandeln. Wie in der Fabel von Äsop ist es unsere Gans, die die goldenen Eier liefert. Deswegen müssen wir die Gans hegen und pflegen

und nicht nach mehr Eiern gieren. Hegen und Pflegen bedeuten gute und positive Kommunikation; jeder kann Lob und konstruktive Korrektur gebrauchen.

■ **Management:**

Das Management ist im Allgemeinen Stakeholder und Geldgeber, doch für ein erfolgreiches SPI brauchen wir vom Management zudem klare Ziele, klare Befugnisse und eine echte Beteiligung. Wir müssen unser Management davon überzeugen, uns zu treiben und nicht umgekehrt. Dazu hilft konstruktive und regelmäßige Kommunikation. In den meisten Fällen sind wir auch Berater des Managements und müssen uns in dieser Rolle sehen. Dazu gehört, vorausschauend zu planen, zu informieren und Konsequenzen von Entscheidungen (oft in Form von Kosten) aufzuzeigen.

SPI ist ein Projekt, das wie jedes andere Projekt Rahmenbedingungen, Widrigkeiten und Risiken unterliegt. Wird es nicht routinemäßig realisiert, befindet es sich meist auf Level 0 bis Level 1, mit allen Konsequenzen.

SPI lohnt sich immer, wenn es gelingt, das Denken Einzelner zum Guten – hier also zur Prozessorientierung – zu beeinflussen. Prozesse und Prozessdisziplin sind hilfreiche Werkzeuge und keine Bürde.