

4 Bewertung des Software-Entwicklungsprozesses

Zusammenfassung:

Das wesentliche Ziel der Bewertung eines Softwareentwicklungsprozesses ist die Definition von konkreten nachvollziehbaren Maßnahmen zur Verbesserung des Entwicklungsprozesses in einer Organisationseinheit.

Da ein Entwicklungsprozess in Projekten gelebt wird, muss er dort hinsichtlich seines Qualitätsstandes untersucht werden. Das Verfahren MPTM (Management of Software Processes and Technologies based on Metrics) sieht hierfür drei Analysemethoden vor:

- Qualitative Prozessanalyse
- Eingeschränkte quantitative Prozessanalyse
- Erweiterte quantitative Prozessanalyse

Die Entscheidung für eine bestimmte Analysemethode oder eine bestimmte Kombination dieser Methoden hängt von der Verfügbarkeit geeigneter Daten in der betreffenden Organisationseinheit und vom Erfahrungsstand der Organisationseinheit in der Prozessverbesserung ab.

Die qualitative Prozessanalyse findet anhand eines Bewertungsmodells statt, das sich auf die Bewertung von Kernprozessen der Softwareentwicklung konzentriert und damit auf die Identifikation maßgebender Verbesserungspotentiale ausgerichtet ist. Das Modell berücksichtigt auch den Faktor *Mitarbeiter*. Die quantitative Prozessanalyse geschieht mittels Softwaremetriken, die aufgrund messbarer Merkmale des Entwicklungsprozesses ausgewählt werden. Mit Abschluss der Prozessanalyse steht ein Stärken/Schwächen-Profil des Entwicklungsprozesses, auf dessen Basis geeignete Verbesserungsmaßnahmen definiert werden können.

Dieses Kapitel steht in enger Verbindung mit den Kapiteln 7, 8 und 9: Das vollständige Modell zur qualitativen Prozessanalyse ist in Kapitel 7 enthalten. Kapitel 8 fasst eine Reihe praxisrelevanter Metriken zur Unterstützung einer quantitativen Prozessanalyse zusammen. Kapitel 9 zeigt die Ergebnisse einer Prozessbewertung (Prozessstatus und Verbesserungsmaßnahmen) als Beispiel.

4.1 Einführung und Überblick über den Bewertungsablauf

Nach dem Verfahren MPTM erfolgt die Verbesserung der Softwareentwicklung auf Grundlage einer Bewertung des Entwicklungsprozesses und einer Bewertung der Softwaretechnologien. In diesem Kapitel wird die Prozessbewertung behandelt; die Technologiebewertung ist Gegenstand des Kapitels 5. Zuerst werden einige Begriffe des Entwicklungsprozesses erörtert.

Unter Softwareentwicklungsprozess wird eine Menge von geordneten methodengestützten Tätigkeiten verstanden, die ein Entwicklungsteam ausübt, um Software zu entwickeln oder zu warten; dabei wird die Software als eine Menge von Programmen und Daten zusammen mit zugehörigen Dokumenten zu ihrer Entwicklung und Benutzung aufgefasst. Dieses Prozessverständnis ist umfassend, denn es enthält alle Tätigkeiten der Entwicklung – von der Ermittlung und Analyse der Anforderungen bis zu deren Realisierung und Abnahme – sowie alle Tätigkeiten des Projektmanagements, wie etwa Projektplanung, Qualitätssicherung und Konfigurationsmanagement [Rezagholi 00c].

Für eine Prozessbewertung ist jedoch eine differenzierte Betrachtung des Entwicklungsprozesses erforderlich. Um dies zu ermöglichen wird ein weiterer Begriff eingeführt: *Entwicklungs-Teilprozess*. Ein Teilprozess umfasst eine Reihe von Tätigkeiten, die in einem engen Zusammenhang zueinander stehen und definierte Ergebnisse hervorbringen. Diese Ergebnisse stellen ein Teil der Software dar. Als Beispiel eines Teilprozesses kann die Qualitätssicherung betrachtet werden. Die Ergebnisse der Qualitätssicherung sind etwa Testpläne, Testfallspezifikationen, Review- und Testprotokolle. Sie bilden einen Teil der Entwicklungsdocumentation.

Der Begriff Entwicklungs-Teilprozess legt unmittelbar nahe, dass unter Softwareentwicklungsprozess die Gesamtheit aller Teilprozesse verstanden wird. Das Software-CMM, SPICE, CMMI und andere vergleichbare Modelle sind nach diesem Verständnis strukturiert (siehe z.B. Abb. 2.6, 2.8 und 2.9).

Das wesentliche Ziel der Bewertung eines Softwareentwicklungsprozesses ist die Definition von konkreten nachvollziehbaren Maßnahmen zur Verbesserung der Softwareentwicklung in einer Organisationseinheit. Hierfür ist eine Statusbestimmung des Softwareentwicklungsprozesses notwendig, bei der entsprechend der Verbesserungsziele der Organisationseinheit die wesentlichen Teilprozesse detailliert untersucht werden.

Die Untersuchung erfolgt durch Interviews mit ausgewählten Personen aus der Softwareentwicklung und durch Studium der Planungs- und Entwicklungsdokumente. Die Ergebnisse sollen die Organisationseinheit in die Lage versetzen, die notwendigen Entscheidungen zur Verbesserung des Entwicklungsprozesses auf einer fundierten Basis zu treffen. Abb. 4.1 zeigt die vier Phasen einer Prozessbewertung mit MPTM. Die Phasen werden im Verlauf dieses Kapitels ausführlich behandelt.

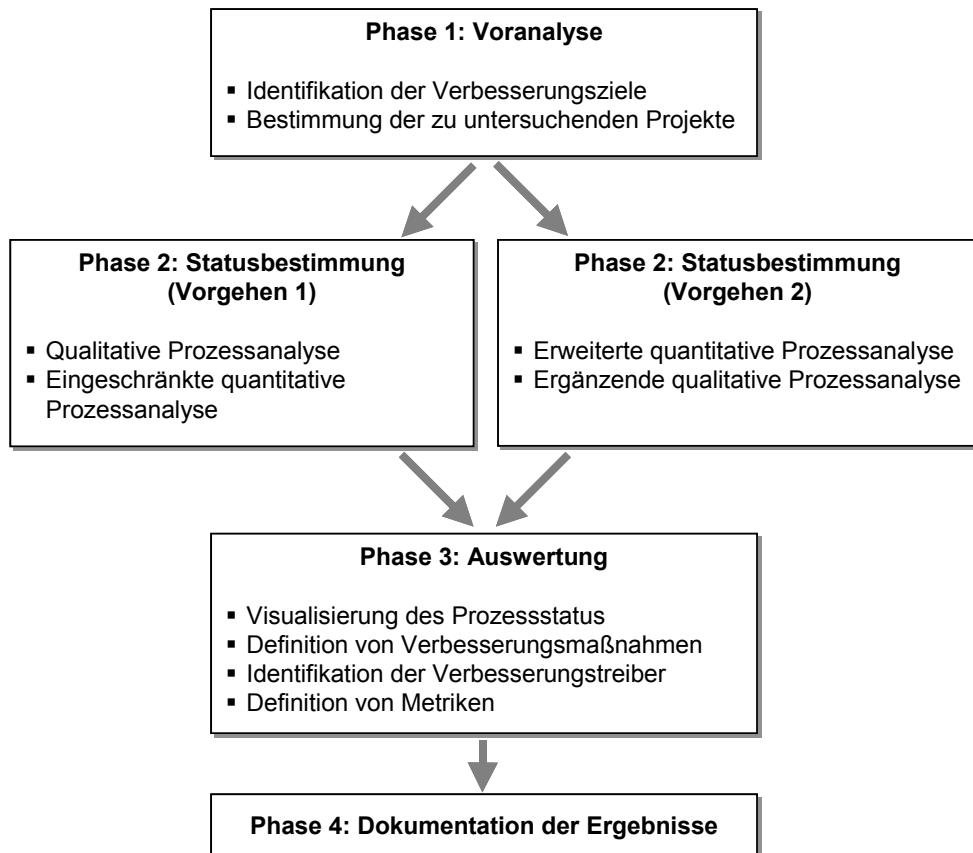


Abb. 4.1 Phasen der Prozessbewertung

4.2 Voranalyse

4.2.1 Identifikation der Verbesserungsziele

Das Vorgehen zur Ermittlung unternehmensindividueller Verbesserungsziele wurde bereits in Abschnitt 3.2 diskutiert. Die ermittelten Ziele müssen nun konsolidiert werden, so dass eine überschaubare Anzahl stabiler entscheidender Ziele entsteht. Es ist von besonderer Bedeutung, dass die definierten Ziele keine kurzfristigen Ziele der Organisationseinheit darstellen, denn es wird – je nach Umfang der Verbesserung – Zeit benötigt, bis die Ergebnisse einer Prozessverbesserung greifen und Erfolge sichtbar werden. Die Ziele dürfen nicht inzwischen ihre Gültigkeit verloren haben.

Darüber hinaus muss noch untersucht werden, ob es unter den Verbesserungszielen der Organisationseinheit Ziele gibt, die eindeutig als Ziele der Prozessverbesserung zu betrachten sind. Es ist allerdings selten von vornherein ersichtlich, ob sich ein Ziel ausschließlich über eine Prozessverbesserung oder nur über eine Technologieverbesserung erreichen lässt. Viele Ziele können nur durch eine konzertierte Verbesserung des Entwicklungsprozesses, der Technologien und des Mitarbeitermanagements erreicht werden. Tendenziell werden Ziele mit Bezug zur Reduktion der Entwicklungszeit, Senkung der Entwicklungskosten und Reduktion der Fehler über eine Prozessverbesserung und Ziele mit Bezug zur Produktqualität (Skalierbarkeit, Zuverlässigkeit, Effizienz usw.) über eine Technologieverbesserung erreicht.

4.2.2 Bestimmung der Projekte

Ein Entwicklungsprozess wird in Projekten gelebt, daher muss er auch dort hinsichtlich seiner Qualität untersucht werden. Es muss also analysiert werden, wie gut der Prozess und die darin enthaltenen Regelungen und Methoden zur Projektdurchführung geeignet sind. Zu diesem Zweck müssen in dieser Phase der Bewertung repräsentative Projekte ausgewählt werden. Der Auswahl ist eine große Bedeutung beizumessen, denn sie bestimmt die Aussagekraft der Ergebnisse maßgebend.

Da die Prozessbewertung in einer einzigen Organisationseinheit stattfindet und sich damit auf eine einzige Anwendungsdomäne bezieht, ist die Projektauswahl erleichtert. Denn zur Eigenschaft einer Anwendungsdomäne zählt, dass die Projekte ähnliche Produkte entwickeln und dazu Ausprägungen eines einzigen Entwicklungsprozesses verwenden. Die Projekte sind also vom gleichen Typ; das einzig bedeutende Unterscheidungsmerkmal bleibt die Projektgröße. Sind die Projekte von ähnlicher Größe, so kann die Prozessbewertung sogar anhand eines einzigen Projektes erfolgen. Weisen die Projekte unterschiedliche Größe auf, so können ein großes und ein kleineres Projekt die Organisationseinheit in der Bewertung repräsentieren.

4.2.3 Studium der Entwicklungsdokumente

Ziel des Dokumentenstudiums ist, erste Erkenntnisse über den Entwicklungsprozess zu gewinnen und diese den ermittelten Verbesserungszielen gegenüber zu stellen. Die Erkenntnisse können sich auf die Unzulänglichkeiten oder Unklarheiten im Prozess beziehen. Sie liefern jedenfalls Anhaltspunkte für die Interviews, die in der zweiten Phase der Prozessbewertung zur Ermittlung des Prozessstatus stattfinden. Zudem können aufgrund dieser Erkenntnisse, geeignete Gesprächspartner für die Interviews bestimmt werden.

Im Folgenden werden eine Reihe von Projektdokumenten aufgelistet, die einen hohen Informationsgehalt haben und deren kritische Prüfung im Vorfeld der Ermittlung des Prozessstatus sinnvoll ist. Der Inhalt und die besondere Bedeutung dieser Dokumente für die Prozessbewertung werden in Abschnitt 4.3 noch klarer.

- **Allgemeine Dokumente**

Diese Dokumente werden vor allem studiert, um die Verflechtungen der Projekte und die in Projekten verwendete Terminologie zu verstehen. Beispiele solcher Dokumente sind die Organisationspläne der Projekte, die Lieferantenverträge, das Prozesshandbuch (Richtlinien der Organisationseinheit für die Softwareentwicklung) und die Produkt- und Technologie-Roadmap.

- **Projektspezifische Dokumente**

Die projektspezifischen Dokumente können eingeteilt werden in Entwicklungsdokumente sowie in Planungs- und Steuerungsdokumente. Zu den Entwicklungsdokumenten zählen Softwarespezifikationen aller Art, wie etwa Anforderungs-, Architektur-, Design- und Testspezifikation.

Beispiele für Dokumente der Planung und Steuerung sind Projektpläne (Termin und Aufwand je Arbeitspaket), Mitarbeitereinsatzpläne (Liste aller aktuellen und ehemaligen Projektmitglieder mit Angaben insbesondere zu deren Funktion im Projekt und zur Dauer ihrer Projektzugehörigkeit), Konfigurationsmanagementpläne, Qualitätssicherungspläne, Projektstatusberichte, Testberichte (Fehlerstatistiken, Zuverlässigkeit der Software usw.).

4.3 Ermittlung des aktuellen Prozessstatus

Ziel der Statusbestimmung ist es, anhand von verfügbaren Daten zu zeigen, wie es mit dem aktuellen Leistungsstand des Softwareentwicklungsprozesses steht und welche Verbesserungspotentiale gegebenenfalls bestehen. Im Einzelnen zielt die Statusbestimmung auf Folgendes ab:

- Darstellung des aktuellen Status des Entwicklungsprozesses in Form eines Stärken/Schwächen-Profils als Grundlage für die Identifikation potentieller Verbesserungsbereiche,
- Prüfung, ob sich die Prozessqualität seit der letzten Bewertung verbessert hat (Wie hoch ist der Grad der bisherigen Prozessverbesserung?),
- Gegebenenfalls Berücksichtigung der Konsequenzen, die sich aus dieser Prüfung für die Identifikation von potentiellen Verbesserungsbereichen ergeben.

Ob in einer Organisationseinheit oder in einem Projekt Probleme im Entwicklungsprozess bestehen, lässt sich leicht feststellen. Auf solche Probleme deuten z.B. folgende Symptome hin:

- Projektmeilensteine sind bereits mehrfach verschoben worden.
- Der Projektfortschritt entzieht sich jeder Beurteilung: Die Projektpläne sind seit längerem nicht mehr aktualisiert worden; eine Schätzung des noch zu erbringenden Aufwandes liegt nicht vor. Folglich gibt es widersprüchliche Aussagen zum Projektstatus.
- Ab etwa der Mitte der geplanten Projektdauer nimmt die Zahl an Überstunden permanent zu.

- Die Entwicklungsergebnisse sind instabil und enthalten viele Fehler.
- Die Entwicklungsdokumente, insbesondere die Spezifikationen, sind mangelhaft. Eine geordnete Implementierung ist erschwert.

Allein die Erkenntnis, dass Probleme im Entwicklungsprozess vorhanden sind, reicht nicht aus, um einen Entwicklungsprozess wirksam zu verbessern. Die Schwachstellen im Prozess müssen im Rahmen einer Analyse ermittelt und insbesondere ihre Ursachen genau lokalisiert werden.

Die Statusbestimmung, im Sinne von MPTM, soll eine möglichst quantitative Darstellung des Entwicklungsprozesses sein. Der Statusbestimmung geht also ein *zielorientiertes Messen* voraus. Messen zielt seinerseits auf die Lieferung von Daten ab, die eine Verbesserung der Software und der Softwareentwicklung ermöglichen. Messen darf also kein Selbstzweck sein; es muss mit Verbesserungstätigkeiten einhergehen.

Das MPTM sieht drei Analysemethoden zur Ermittlung des aktuellen Prozessstatus einer Organisationseinheit vor:

1. Qualitative Prozessanalyse,
2. Eingeschränkte quantitative Prozessanalyse,
3. Erweiterte quantitative Prozessanalyse.

Nicht alle diese Analysemethoden kommen bei der Ermittlung des Prozessstatus gleichzeitig zum Einsatz. Es gibt zwei Vorgehen zur Ermittlung des Prozessstatus. Die Entscheidung, welches Vorgehen geeignet ist, hängt vom Erfahrungsstand der betreffenden Organisationseinheit in der Verbesserung der Softwareentwicklung ab:

Vorgehen 1: Qualitative Prozessanalyse mit einer ergänzenden quantitativen Prozessanalyse

Eine Organisationseinheit, die bisher keine mehrjährige (3- bis 5-jährige) systematische Verbesserung der Softwareentwicklung durchgeführt hat, besitzt demzufolge kaum oder nur wenig motivierte Daten als Ergebnis von Messungen. In einer solchen Organisation kommt die *qualitative Prozessanalyse* zum Einsatz. Der Prozessstatus wird anhand von Aussagen ermittelt, die aus dem Studium der Entwicklungsdokumente und den Befragungen der Fachleute der Organisationseinheit gewonnen werden. Abschnitt 4.3.1 zeigt die Prozesse und Prozessaspekte, die bei einer qualitativen Analyse zu beachten sind.

Über die qualitative Analyse hinaus soll eine *eingeschränkte quantitative Prozessanalyse* als Ergänzung stattfinden. Im Rahmen dieser zusätzlichen Analyse wird der Versuch unternommen, aus den wenigen, mehr oder minder motivierten und zufälligen Daten (z.B. Entwicklungsaufwand, Fehlerzahl), die in einer solchen Organisationseinheit vorhanden sind, quantitative Aussagen zum Prozessstatus zu gewinnen (siehe Abschnitt 4.3.2).

Vorgehen 2: Erweiterte quantitative Prozessanalyse mit einer ergänzenden qualitativen Prozessanalyse

Eine Organisationseinheit mit mehrjähriger Erfahrung in Verbesserung der Softwareentwicklung besitzt Metriken, die zielorientiert definiert und systematisch erhoben werden.

Die Organisationseinheit bedient sich der Messergebnisse als Grundlage für Entscheidungen und für Prognosen bezüglich der Softwareentwicklung. In einer solchen Organisationseinheit kommt die *erweiterte quantitative Prozessanalyse* zum Einsatz: eine systematische und umfassende quantitative Ermittlung des Prozessstatus (Abschnitt 4.3.3).

Im Rahmen dieses Vorgehens wird die *qualitative Prozessanalyse* nur als Ergänzung herangezogen. Auf sie kann nicht vollständig verzichtet werden, denn viele Teilprozesse der Entwicklung lassen sich qualitativ wirtschaftlicher bewerten als quantitativ. Dies wird im weiteren Verlauf dieses Kapitels noch aufgegriffen.

Es bedarf allerdings einiger Jahre, bis eine Organisationseinheit eine weitgehend quantitative Bewertung und Verbesserung der Softwareentwicklung durchführen kann, denn die Metriken werden im Zeitverlauf immer solider und können immer zuverlässigere Messergebnisse liefern. Bis dieser Zustand erreicht ist, mit anderen Worten, bis eine Organisationseinheit verlässliche Daten über ihre Projekte besitzt, die für eine ebenso verlässliche Statusbestimmung verwendet werden können, werden die Statusbestimmungen, zumal die ersten, vorwiegend auf einer qualitativen Basis (Vorgehen 1) stattfinden.

4.3.1 Qualitative Prozessanalyse

Der Softwareentwicklungsprozess besteht, wie in der Einführung zu diesem Kapitel dargestellt, aus einer Vielzahl von Entwicklungs-Teilprozessen. Daher kann eine qualitative Analyse des Entwicklungsprozesses recht aufwändig werden, es sei denn, die Analyse beschränkt sich auf die *Kernprozesse der Softwareentwicklung*. Ein Kernprozess der Softwareentwicklung ist ein Entwicklungs-Teilprozess mit folgenden Eigenschaften:

- Er ist erfahrungsgemäß für den Erfolg eines Projektes von besonderer Bedeutung; z.B. ist Requirements Engineering ein Kernprozess.
- Er wird ausschließlich anhand von *Prozessaspekten* (Prozessthemen und -tätigkeiten) beschrieben, die für den erfolgreichen Ablauf des Prozesses unbedingt erforderlich sind. Für das Beispiel Requirements Engineering sind folgende Aspekte ausschlaggebend: Produktstrategie, Versions- und Release-Planung, Ermittlung und Analyse der Anforderungen, Spezifikation der Anforderungen sowie Verfolgung der Anforderungen.

Bei den Kernprozessen ist davon auszugehen, dass sie den wesentlichen Anteil der *Verbesserungstreiber* stellen werden und es verdienen, vorrangig untersucht zu werden. Mit anderen Worten: Konzentriert sich die Prozessanalyse auf die Kernprozesse, so können die ernsthaften Schwachstellen des Entwicklungsprozesses in kürzerer Zeit und mit weniger Aufwand ermittelt werden. Folglich kann die Verbesserung des Entwicklungsprozesses auf Kernprozesse ausgerichtet und damit zielführend gestaltet werden.

Die Kernprozesse lassen sich zwei Kategorien zuordnen: Kernprozesse des Softwaremanagements und Kernprozesse der Softwarerealisierung. Jede Kategorie besteht aus fünf Kern-

prozessen (Abb. 4.2)². Es ist zu beachten, dass eine exakte Zuordnung eines Kernprozesses zu einer Kategorie nicht immer möglich ist, z.B. enthält die Qualitätssicherung neben Komponenten der Softwarerealisierung auch Komponenten des Projektmanagements, der Planung, Steuerung und Kontrolle.

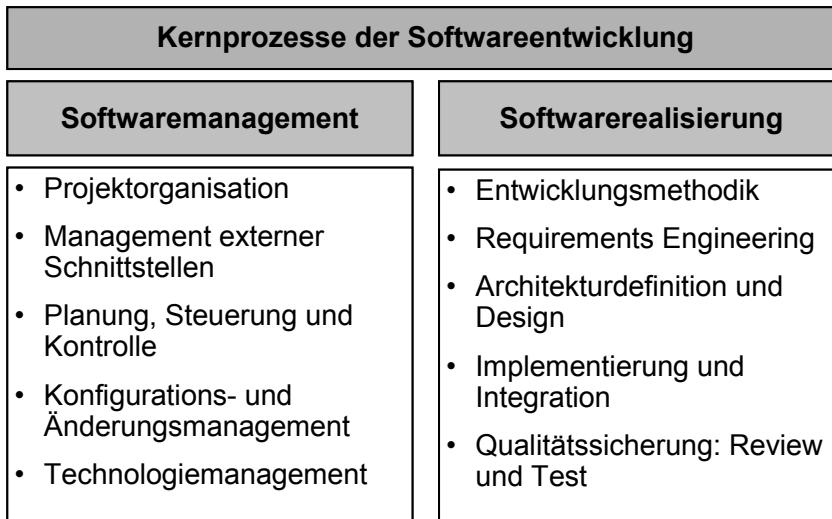


Abb. 4.2 Kernprozesse der Softwareentwicklung

Im Folgenden werden die Kernprozesse der Softwareentwicklung kurz beschrieben. Kapitel 7 enthält eine nähere Beschreibung dieser Prozesse in Form eines Modells. Das Modell dient als Grundlage für eine qualitative Prozessanalyse.

1. Projektorganisation

Es gilt herauszufinden, welche Aufgabenbereiche im Projekt definiert sind und wie sie wahrgenommen werden. Neben der klaren Abgrenzung der Zuständigkeiten müssen Beziehungswege sowohl innerhalb des Projekts als auch zwischen dem Projekt, dem Management, den anderen beteiligten Organisationseinheiten und Projekten festgelegt sein. Entscheidend ist allerdings das Gesamtbild, das auf eine gute Zusammenarbeit im Projekt und in der Organisation hindeutet muss: Wichtige Aspekte sind die Art und Weise des Informationsaustauschs und der Abstimmungen, die Koordination der Aufgaben und die

² Die Auswahl dieser Prozesse als Kernprozesse der Softwareentwicklung gründet auf eigenen Erfahrungen, die im Rahmen von zahlreichen Prozessbewertungen und Projekt-Risiko-Audits gemacht wurden, und auf Erfahrungen anderer, die mir in Form von Bewertungsberichten zur Verfügung standen (insbesondere Erfahrungen mit CMM, und Bootstrap-Methode) oder in unterschiedlichen Literaturstellen [siehe insbesondere SPMN 03; Jones 00; Rezagholi 00a; Rezagholi 00b; SPMN 00; Brown 98; ISO 15504; Peters 98; SPMN 98a; SPMN 98b; Humphrey 97; McConnell 97; Paulk 97; Yourdon 97; Flowers 96; McConnell 96; Brooks 95; Kornwachs 95; Lowe 95; Thomsett 95; Trillium 94; SEL 92; Humphrey 89; DeMarco 86] zu finden sind.

Konfliktlösung zwischen den betroffenen Gruppen. Eine lebendige Kommunikationsstruktur im Projekt ist von besonderer Bedeutung, da nur so Risiken aufgedeckt und Informationen ausgetauscht werden können.

2. Management externer Schnittstellen

Alle für den Erfolg des Projektes wesentlichen externen Schnittstellen, insbesondere die Schnittstellen zu den Kunden und zu den Lieferanten, müssen untersucht werden. Durch geeignetes Management dieser Schnittstellen, muss gewährleistet sein, dass die Aufgabenteilung auf die jeweils verantwortlichen Stellen und der Informationsaustausch zwischen diesen Stellen im Hinblick auf die optimale Abwicklung des Projekts ausgerichtet sind.

3. Planung, Steuerung und Kontrolle

Es ist zu prüfen, ob der Projektplan eine verbindliche Grundlage für die Entwicklung darstellt: Die auszuführenden Arbeiten sind eindeutig bestimmt und der Projektablauf ist so gestaltet, dass die Zeit-, Kosten- und Qualitätsziele unter bestehenden Restriktionen eingehalten werden können. In diesem Zusammenhang kommt dem Aspekt Aufwandsschätzung eine große Bedeutung zu.

Die wesentlichen Projektrisiken sind identifiziert und allen Beteiligten mitgeteilt. Die Auswirkungen der Risiken sind ermittelt; um folgenschweren Risiken zu begegnen, werden Maßnahmen definiert. Ferner wird der Ist-Zustand des Projektes, etwa hinsichtlich des Aufwands und der Termine, regelmäßig geprüft, um gegebenenfalls in das Projektgeschehen steuernd eingreifen zu können.

Die Planung, Steuerung und Kontrolle sind die entscheidenden Faktoren für das Gelingen eines Projekts und damit auch oft das Gebiet mit dem größten Verbesserungspotential. Mängel in diesem Bereich haben große Auswirkungen auf den Projekterfolg. Es muss daher sehr genau untersucht werden, ob die Projektsteuerung wirksam erfolgt und damit alle Projekt-Beteiligten in die Lage versetzt, zielführend zusammenzuarbeiten.

4. Konfigurations- und Änderungsmanagement

Es ist zu untersuchen, ob die Konsistenz der Projektergebnisse gegeben ist. Das bedeutet zum einen, dass die wesentlichen Entwicklungsergebnisse dem Konfigurationsmanagement unterstellt werden, dass hierfür ein angemessenes Werkzeug im Einsatz ist, und dass die Ergebnisse systematisch freigegeben werden. Zum anderen muss das definierte Verfahren zum Änderungsmanagement gewährleisten, dass die Änderungen an freigegebenen Arbeitsergebnissen, etwa aufgrund von Fehlerkorrekturen oder Anforderungsänderungen, kontrolliert vorgenommen werden.

5. Technologiemanagement

Erfolgreiche Projekte erfordern eine konsequente Ausrichtung von Produkt- und Prozess-technologien auf Marktbedürfnisse und Wirtschaftlichkeit. Es ist daher zu prüfen, ob diese Fokussierung von einem effizienten Technologiemanagement unterstützt wird: Es werden geeignete Methoden und Techniken bereitgestellt, um den technologischen Handlungsbedarf der Organisationseinheit auf der Grundlage einer systematischen Analyse

und Bewertung zu ermitteln, die entsprechenden Technologien zu beschaffen und kontrolliert in die Entwicklung zu integrieren [Rezagholi 00a; Rezagholi 00b].

Ferner ist zu untersuchen, ob der Wissenstransfer innerhalb der Projekte sowie zwischen den Projekten, den Kunden und den Technologie-Lieferanten auf geeignete Weise stattfindet.

6. Entwicklungsmethodik

Das projektspezifische Prozessmodell muss definiert und von allen Beteiligten verstanden und akzeptiert sein. Es muss sichergestellt sein, dass das gewählte Prozessmodell und die darin festgelegten Methoden und Werkzeuge für das Entwicklungsvorhaben geeignet sind. Dabei ist nicht nur die Ausstattung der Projekte mit Softwarewerkzeugen zur Unterstützung der Entwicklungs- und Projektmanagement-Tätigkeiten von Bedeutung. Ebenso wichtig ist eine Infrastruktur, die den Austausch von Entwicklungsergebnissen ermöglicht, insbesondere wenn diese an unterschiedlichen Orten (verteilte Entwicklung) entstehen.

7. Requirements Engineering

Requirements Engineering ist ein Prozess von herausragender Bedeutung, denn er bestimmt die Stabilität und Qualität der Anforderungen. Daher muss genau geprüft werden, ob die Anforderungen anhand von geeigneten Methoden ermittelt, analysiert und dokumentiert werden. Ferner muss die Vorgehensweise zum Requirements Engineering gewährleisten, dass alle Projektbeteiligten das gleiche Verständnis der Anforderungen gewinnen.

8. Architekturdefinition und Design

Es muss ein systematisches Verfahren zur Definition tragfähiger Architekturen für die Produkte der Organisationseinheit geben. Das Verfahren regelt insbesondere die Hauptaufgaben der Architekturdefinition und der Design-Spezifikation. Das sind etwa die funktionale Aufteilung des Systems, Definition der Systemkomponenten und deren Schnittstellen, Modellierung des dynamischen Verhaltens der Komponenten, Sicherstellung der nicht funktionalen Eigenschaften, Wiederverwendung verfügbarer Komponenten sowie Koordination aller Architekturausprägungen und Schnittstellen.

9. Implementierung und Integration

Die Implementierung muss auf eine systematische Umsetzung der Anforderungen in ein konkretes Produkt mit den entsprechenden Dokumentationen hindeuten. Im Mittelpunkt stehen die Erstellung und der Test von Modulen bzw. Komponenten des Softwareprodukts. Diese Arbeiten erfolgen unter Einhaltung definierter Vorgaben und Konventionen.

Bei der Integration werden Module bzw. Komponenten – unter Wiederholung entsprechender Tests – schrittweise zusammengefügt, bis das vollständige Produkt erstellt ist. Der Integrationstest konzentriert sich auf die Prüfung der Schnittstellen zwischen den verschiedenen Modulen bzw. Komponenten.

10. Qualitätssicherung: Review und Test

Es ist zu untersuchen, ob das Projekt die Einhaltung von Anforderungen und Standards laufend überwacht sowie die Review- und Test-Ergebnisse erfasst und analysiert, mit

dem Ziel, Aussagen bezüglich der Qualität der Ergebnisse treffen zu können und Fehler möglichst früh zu lokalisieren.

Es muss festgelegt sein, welche Entwicklungsergebnisse zu welchem Zeitpunkt einem Review und/oder Test unterzogen werden. Alle Reviews und Tests werden geplant. Der Plan regelt nicht nur den Ablauf, sondern auch die jeweils einzusetzenden Methoden. Er legt ferner die Erfassung und Verwendung von Review- und Testergebnissen fest.

Die Prüfung, ob ein Entwicklungsprozess diese Anforderungen erfüllt, geschieht, wie bereits erwähnt, durch Interviews mit Projektmitgliedern. Als Gesprächspartner kommen lediglich die entscheidenden Projektmitglieder in Frage, wie etwa Projektleiter, Produktmanager, Systemarchitekt, Verantwortliche für Konfigurationsmanagement, Verantwortliche für Qualitätssicherung, ausgewählte Entwickler und Tester. Diese können in Gruppen oder einzeln befragt werden. Die Entscheidung hierfür muss immer im Sinne der Effizienz der Informationsgewinnung getroffen werden. Gegebenenfalls können durch Gespräche mit ausgewählten Kunden und Lieferanten weitere Informationen über die Zusammenarbeit an den Schnittstellen des Projekts gewonnen werden.

Zusammengefasst ermittelt die qualitative Analyse den Prozessstatus einer Organisationseinheit in Bezug auf die Kernprozesse der Softwareentwicklung. Dieses Ergebnis, insbesondere die festgehaltenen Schwächen des Entwicklungsprozesses, ist die Grundlage für die Definition von Verbesserungsmaßnahmen. Der Prozessstatus soll knapp aber präzise dargestellt werden (zur Darstellung des Status eines Entwicklungsprozesses als Ergebnis einer qualitativen Prozessanalyse siehe Abb. 9.2 in Abschnitt 9.3).

4.3.2 Eingeschränkte quantitative Prozessanalyse

Die quantitative Prozessanalyse ermöglicht es, die oben genannten Ziele der Statusbestimmung, nämlich die Darstellung des Leistungsstands eines Prozesses und die Identifikation von Verbesserungspotentialen, auf eine sachlich nachvollziehbare Basis zu stellen. Dadurch werden die Ergebnisse der Bewertung leichter verstanden und können ebenso leichter vermittelt werden. Aus diesem Grund muss schon sehr früh versucht werden, als Ergänzung zu einer qualitativen Analyse eine quantitative Statusbestimmung vorzunehmen und damit die auf *Messen* basierte Prozessverbesserung einzuleiten.

Es ist also anzustreben, Prozesse, die einer quantitativen Bewertung unterzogen werden können, wie etwa der Test-, Planungs- und Steuerungsprozess, auch quantitativ zu bewerten. Gerade das scheint aber im Widerspruch zu der Tatsache zu stehen, dass in einer Organisationseinheit, in der das systematische Bewerten und Messen bisher keine Beachtung gefunden hat, keine zweckdienlich aufbereiteten Daten vorliegen.

Trotzdem ist es möglich, unverzüglich mit einer quantitativen Prozessanalyse anzufangen und ihr zunehmend eine größere Bedeutung in der Prozessbewertung einzuräumen. Diese Forderung ist dadurch begründet, dass auch Organisationseinheiten ohne Erfahrung in systematischer Prozessverbesserung über Daten, zum Teil sogar ihnen unbekannte Rohdaten, verfügen, die zur quantitativen Darstellung der Softwareentwicklung verwendet werden

können: Aufgrund mangelhafter Qualität der Daten kann es sich hier nur um eine *eingeschränkte* quantitative Prozessanalyse handeln. An diese Statusbestimmung dürfen keine hohen Anforderungen gestellt werden. Es ist jedoch möglich, durch Fokussierung auf ein oder zwei wesentliche Ziele und durch Analyse einiger für diese Ziele charakteristischer Daten einige Hinweise auf Verbesserungspotentiale zu gewinnen.

Zu diesem Zweck müssen alle verfügbaren Daten betrachtet werden, die einen Beitrag zur quantitativen Bestimmung des Zustands eines Entwicklungsprozesses leisten können. Als Folge der Interpretation dieser Daten werden Stärken und Schwächen des Prozesses und damit die Verbesserungspotentiale identifiziert. Diese sollten mit den zuvor vom Management formulierten Zielen korrespondieren. Korrespondiert ein Teil von identifizierten Verbesserungspotentialen nicht mit den Zielen, dann kann das ein Hinweis auf die Unvollständigkeit der Ziele sein. Es ist zu beachten, dass auch das Management einen Lernprozess durchlaufen muss, bevor es die wesentlichen Verbesserungsziele sicher identifizieren und festlegen kann.

Auf der Suche nach Daten dürfen nicht nur Quellen betrachtet werden, wo entsprechende Daten in der Regel zu finden sind, sondern auch alle möglichen Projekt- und Entwicklungs-dokumente, die in einem solchen Unternehmen erstellt werden. Zudem muss oft, mangels dokumentierter Daten, auf das Gedächtnis der Projektmitarbeiter zurückgegriffen werden. Es ist also herauszufinden, was für Daten verfügbar sind und welche Schlüsse aus ihnen gezogen werden können.

Eine systematische Vorgehensweise zur Erlangung von benötigten Daten gibt es in diesem Fall nicht. Es muss situativ und aus der Erfahrung heraus gehandelt werden. Erfahrungsge-mäß können folgende Primärdaten leicht ermittelt werden:

- **Anzahl der von Kunden gemeldeten Fehler**

Jede softwareentwickelnde Organisationseinheit, unabhängig von ihrer Reife, erfasst zwangsläufig die von Kunden gemeldeten Fehler. Viele von ihnen erfassen auch noch den Aufwand oder die Kosten der Fehlerbehebung, einige sogar noch die Fehlerklassen (Klassifizierung der Fehler nach ihrem Ausmaß und ihrer Bedeutung). Die Anzahl von gemeldeten Fehlern kann über eine Reihe von Projekten oder Releases analysiert werden. Wesentlich ist jedoch, dass lediglich die Fehler beachtet werden, die in einem festgelegten Zeitraum gemeldet werden (z.B. im ersten Jahr des Einsatzes jedes Releases), um die Vergleichbarkeit sicherzustellen.

- **Anzahl der Anforderungsänderungen**

Viele softwareentwickelnde Organisationseinheiten erfassen die Anzahl der Anforde-rungsänderungen, die von Kunden an das Projekt gestellt werden. Eventuell kann noch herausgefunden werden, wie viele Anforderungsänderungen bisher angenommen, abge-wiesen oder zurückgestellt wurden.

- **Entwicklungszeit**

Fast immer können die geplante und tatsächliche Laufzeit eines Projekts und die Dauer der wesentlichen Projektphasen einigermaßen genau ermittelt werden. Im ungünstigsten Fall, d.h. wenn keine Projektpläne verfügbar sind, besteht die Möglichkeit, den Projekt-

leiter und die Projektmitglieder nach der geplanten und tatsächlichen Projektdauer gezielt zu befragen: Wann begann das Projekt? Wann begann der Test? Wann wurde die Software fertiggestellt?

- **Entwicklungsaufwand**

Auch der Aufwand kann, sollten die Projektpläne hierzu keine Daten liefern können, auf einem Umweg ermittelt werden: Zunächst ist die Anzahl von Projektmitgliedern zu bestimmen. Ist ein Kontierungssystem im Einsatz, so kann es dazu verwendet werden, die Anzahl der Personen und nach Möglichkeit auch die Anzahl der erbrachten Stunden festzustellen. Ansonsten kann wieder durch die Befragung der Projektbeteiligten ermittelt werden, wann und für wie lange jede Person bisher im Projekt tätig war. Nun kann der Versuch unternommen werden, den Entwicklungsaufwand in Mitarbeitermonaten zu berechnen. Eventuell kann noch versucht werden, den Aufwand phasenbezogen zu ermitteln.

- **Softwaregröße**

Die Softwaregröße wird in der Regel in Anzahl von Function Points oder Lines of Code ausgedrückt. Es ist unwahrscheinlich, dass eine in Prozessverbesserung unerfahrene Organisationseinheit die Softwaregröße mittels Function Points-Methode misst, denn diese Methode erfordert eine gewisse Prozessqualität. Daher bleibt Lines of Code meist die einzige Möglichkeit die Softwaregröße zu ermitteln. Die Voraussetzung dafür ist, dass der im Projekt entwickelte Code klar ausgemacht werden kann, dass gewisse Programmierkonventionen von den Entwicklern eingehalten werden und dass geeignetes Messwerkzeug vorhanden ist.

Alternativ kann die Softwaregröße durch die Anzahl der Anforderungen oder die Anzahl der Module repräsentiert werden. In diesem Fall muss berücksichtigt werden, dass die Anforderungen unterschiedliche Realisierungsumfänge implizieren und die Module unterschiedlich groß sein können. Die Größe eines objektorientiert entwickelten Softwareprogramms kann auch mittels Class Points ausgedrückt werden.

- **Produktivität**

Sind der Entwicklungsaufwand und die Softwaregröße ermittelt worden, so kann das Verhältnis der Softwaregröße zum Aufwand als ein einfacher Indikator für die Produktivität betrachtet werden.

- **Testgrößen**

Letztlich kann noch versucht werden, Primärdaten aus dem Test zu gewinnen. Je nach Reife der Organisationseinheit können Anzahl der Testfälle, Anzahl fehlgeschlagener Tests, Anzahl gefundener Fehler, Aufwand und Dauer von Tests usw. ermittelt werden. Auch hier besteht die Möglichkeit aus dem Gedächtnis der Projektmitglieder zu schöpfen.

Der Metrikkatalog (Kapitel 8) enthält weitere Metriken, die in einer eingeschränkten quantitativen Analyse eingesetzt werden können. Im Metrikkatalog werden die Metriken klassifiziert und einzeln beschrieben. Aufgrund der Metrikbeschreibung kann entschieden werden,

welchen Nutzen die betroffene Metrik für die Prozessanalyse hat, ob sie erhoben werden kann (stehen die notwendigen Datenquellen zur Verfügung?) und wann sie auszuwerten ist.

Die folgenden Beispiele zeigen wie diese Daten interpretiert und zu einer Statusbestimmung zusammengesetzt werden können.

Beispiel 1: Ein wesentliches Ziel eines Unternehmens ist die Reduktion von Fehlern, die von Kunden gefunden werden. Nun besteht die Möglichkeit, über den Kundendienst oder die Hotline zu ermitteln, wie viele Fehler in den vorausgehenden Releases von Kunden gemeldet wurden. Die Anzahl der von Kunden gemeldeten Fehler liefert lediglich einen Ansatzpunkt und noch keine sichere Grundlage zur Beurteilung der Produktqualität. Kann noch die Softwaregröße ermittelt werden, dann kann diese zur Relativierung der von Kunden gemeldeten Fehler benutzt werden. Damit ist eine Grundlage zur Beurteilung der Produktqualität geschaffen.

Beispiel 2: Das Verbesserungsziel ist die Reduktion der Entwicklungszeit. In diesem Fall können die Ist- und Plandaten miteinander verglichen werden. Liegen die geplante und die tatsächliche Projektlaufzeit weit auseinander, so ist das ein erstes Indiz für schlechtes Projektmanagement. Wenn es noch gelingt, den Projektaufwand und die Größe des Releases zu ermitteln, dann kann eine einfache Produktivitätsmessung stattfinden. So wird nicht nur das Verständnis für das gesetzte Ziel erhöht, sondern auch Hinweise über die Schwachstellen des Prozesses erzielt.

Insgesamt ist festzuhalten, dass die eingeschränkte quantitative Prozessanalyse eine qualitative Prozessanalyse nur ergänzen kann. Die Aussagen einer solchen Analyse müssen entsprechend der Qualität der verfügbaren Daten beachtet werden.

4.3.3 Erweiterte quantitative Prozessanalyse

Die erweiterte quantitative Prozessanalyse gründet stark auf dem Messen. Sie kommt in einer Organisationseinheit zum Einsatz, die seit einigen Jahren eine systematische Verbesserung durchführt. Eine solche Organisationseinheit kommt im Laufe der Zeit immer mehr in den Besitz von systematisch ermittelten Primärdaten und Metriken, so dass sie den Prozessstatus überwiegend auf der Basis von Metriken feststellen kann.

Allerdings ist zu beachten, dass sich nicht jeder Prozess und vor allem nicht bezüglich aller seiner Aspekte wirtschaftlich, d.h. mit angemessenem Nutzen/Kosten-Verhältnis, quantitativ bewerten lässt. Es gibt Prozessfragen, die anhand einer qualitativen Analyse effizienter beantwortet werden können, z.B. ob in einer Organisationseinheit die Verantwortlichkeiten festgelegt sind, die Berichtswege vorhanden sind oder ob die ausgewählte Entwicklungsmethodik die Anforderungen der Projekte erfüllt. Folglich benötigt die erweiterte quantitative Prozessanalyse eine qualitative Prozessanalyse als Ergänzung.

Die Fähigkeit einer Organisationseinheit, ihre Prozesse weitgehend quantitativ zu bewerten, impliziert nicht, dass die Organisationseinheit eine Vielzahl von Softwaremetriken einsetzt. Es deutet vielmehr darauf hin, dass das Messen in dieser Organisationseinheit systematisch erfolgt:

- Die benötigten Metriken werden zielorientiert identifiziert und präzise definiert.
- Die Erhebung der Primärdaten und die Berechnung oder Ermittlung von Metriken sind genau festgelegt.
- Das Vorgehen zur Prozessanalyse mittels Metriken ist definiert.
- Es gibt ferner ein datenbankgestütztes Werkzeug, das eine geeignete Aufbewahrung und Analyse von Primärdaten und Messergebnissen ermöglicht.

Sind diese Voraussetzungen erfüllt, so ist die Organisationseinheit imstande, durch die Nutzung vorhandener Metriken den Zustand des Softwareentwicklungsprozesses jederzeit – ohne eine eigens geplante Prozessbewertung – darzustellen, den Handlungsbedarf entsprechend zu erkennen und Verbesserungen zu initiieren. Sie kann auf der Basis der Kunden- und der eigenen Bedürfnisse quantitative Verbesserungsziele, wie etwa Reduktion der Entwicklungszeit um 10%, begründen und die Zielerreichung kontrollieren.

Die erweiterte quantitative Prozessanalyse baut auf sechs Metrikkategorien auf:

1. Aufwands- und Kostenmetriken,
2. Zeitmetriken,
3. Umfangsmetriken,
4. Prozessqualitäts-Metriken,
5. Produktqualitäts-Metriken,
6. Geschäftsmetriken.

Im Folgenden werden diese Metrikkategorien kurz behandelt. Kapitel 8 (Metrikkatalog) enthält zu jeder dieser Kategorien eine Reihe von praxisrelevanten Metriken mit detaillierter Beschreibung. Der Metrikkatalog stellt eine Grundlage für die Auswahl, Definition und Anwendung von Metriken zur Analyse der Softwareentwicklung sowie zur Prüfung des Fortschritts in der Verbesserung dar.

Aufwands- und Kostenmetriken

Im Vordergrund stehen der Vergleich des tatsächlich erbrachten mit dem geplanten Aufwand (Aufwandsabweichung) sowie der Vergleich der tatsächlichen mit den geplanten Kosten (Kostenabweichung). Die Abweichungen können für den gesamten Lebenszyklus des Prozesses, für die einzelnen Entwicklungsphasen oder für bestimmte Arbeitspakete ermittelt und untersucht werden. Eine weitere, eng mit Aufwands- und Kostenmetriken verbundene Metrik ist die Anzahl von Projektmitarbeitern im Verlauf der Entwicklung oder je Entwicklungsphase (Personalaufbau und -abbau im Zeitverlauf).

Aus den historischen Messergebnissen sollen Aufwand und Kosten neuer Projekte vorausgesagt werden können – Vergleichbarkeit der Projekte vorausgesetzt. Darüber hinaus soll, unter Einbeziehung von Qualitäts- und Umfangsmetriken, der Einfluss der Faktoren Aufwand, Kosten, Entwicklungszeit und Qualität aufeinander analysiert werden können.

Anhand der Aufwands- und Kostenmetriken können eine Reihe von Projektfragen beantwortet werden, wobei die Kenntnis von Terminen und/oder das Wissen über den Umfang der zu erbringenden oder erbrachten Leistungen erforderlich ist. Beispiele solcher Fragen sind:

- Entsprechen die Projektergebnisse dem Projektplan?
- Ist der geplante Aufwand bzw. sind die geplanten Kosten konsistent mit dem vorgegebenen (geplanten) Budget?
- Reicht der geplante Aufwand aus, um die Termine einhalten zu können?
- Der Ressourcenverbrauch steigt. Ist das Projekt daher neu zu planen?

Zeitmetriken

Die Zeitmetriken dienen, ähnlich wie die Aufwands- und Kostenmetriken vor allem der Ermittlung von Terminabweichungen über den Vergleich der aktuellen Termine mit den geplanten Terminen. Auch hier können die Abweichungen in Bezug auf die gesamte Entwicklung, die einzelnen Entwicklungsphasen oder die Entwicklungszeit bestimmter Arbeitspakete ermittelt werden. Werden die Zeitmetriken über einen längeren Zeitraum erhoben, dann können sie zur Prognose von Entwicklungszeiten und Terminen in neuen Projekten verwendet werden. In Verbindung mit Aufwands-, Qualitäts- und Umfangsmetriken können weitere Analysen durchgeführt werden.

Ferner können mit Zeitmetriken weitere Projektfragen beantwortet werden. Auch hier ist es oft erforderlich, den Aufwand und den ihm gegenüberstehenden Leistungsumfang zu kennen. Beispiele solcher Fragen sind:

- Liegen die Projektergebnisse zu den geplanten Terminen vor?
- Lässt sich aus dem bisher erbrachten Aufwand den Schluss ziehen, dass die Termine eingehalten werden können?
- Der Ressourcenverbrauch steigt. Sind die Projekttermine daher neu zu planen?

Umfangsmetriken

Der Umfang der Ergebnisse einer Softwareentwicklung (Spezifikationen, Programme) lässt sich über unterschiedliche Maßeinheiten, wie etwa Anzahl der Function Points, der Lines of Code, der Class Points, der Anforderungen oder der Seiten ausdrücken. Mit Umfangsmetriken soll der Projektfortschritt im gesamten Lebenszyklus der Entwicklung dargestellt werden: Der Fertigstellungsgrad von Spezifikationen oder des Codes im Zeitverlauf.

Aus den historischen Umfangsdaten kann in zukünftigen Projekten die Gesamtgröße der Software, die Größe einzelner Softwarekomponenten, die Größe des neu geschriebenen Codes oder die Größe des wiederverwendeten Codes vorausgesagt werden. Dann besteht auch die Möglichkeit, die Abweichung der tatsächlichen von der geschätzten Softwaregröße zu analysieren.

Umfangsmetriken sind Anzeichen für die Vollständigkeit der Anforderungen sowie für die Fähigkeit des Projektteams, das Softwareprodukt innerhalb des Budgets und Termins zu entwickeln. Sie sind damit die unabdingbare Ergänzung zu den Aufwands- und Zeitmetriken, um zuverlässige Aussagen zum Projektstatus machen zu können. Mittels Umfangsmetriken können Fragen folgender Art beantwortet werden:

- Weicht der tatsächliche Umfang der Entwicklungsergebnisse von der geschätzten ab? Wie groß ist die Abweichung?

- Lässt die Zunahme der Softwaregröße im Projektverlauf darauf schließen, dass der geplante Termin und Aufwand einzuhalten ist?
- Wie steht es mit der aktuellen Produktivität des Projektteams insgesamt (über alle Entwicklungsergebnisse hinweg)?
- Ist es aufgrund des aktuellen Fertigstellungsgrads von Entwicklungsergebnissen notwendig, das Projekt neu zu planen?

Prozessqualitäts-Metriken

Die Prozessqualitäts-Metriken dienen der Beurteilung der Qualität aus Prozesssicht, z.B. in Bezug auf die Erfüllung von Anforderungen durch die Software oder die Wirksamkeit der Fehlerfindung und -behebung. Sie liefern also Hinweise, sowohl auf die Produktqualität als auch auf die Prozessqualität. Sie zeigen ferner, worauf eine mangelnde Prozessqualität zurückzuführen ist.

Die Prozessqualitäts-Metriken weisen eine große Vielfalt auf. Sie beziehen sich auf die Anforderungen (Anzahl erfüllter oder nichterfüllter Anforderungen), Anforderungsänderungen (eingegangen, offen, angenommen, implementiert), Fehler (Anzahl, Priorität, wann gemeldet, wann analysiert, wann behoben, Fehlerbehebungskosten) usw.

Beispiele für Fragen, die mittels Qualitätsmetriken beantwortet werden können, sind:

- Werden die spezifizierten Anforderungen auch tatsächlich implementiert?
- Wie verhält sich die Anzahl von Anforderungsänderungen im Projektverlauf? Nimmt sie ab oder steigt sie?
- Wie entwickeln sich die Fehlerzahl, die Fehlerzahl pro Fehlerart und die Fehlerdichte (Anzahl Fehler pro Modulgröße) im Zeitverlauf?
- Wie entwickeln sich die Fehlersuche und -behebung im Projekt oder in einzelnen Testphasen?
- Kann die Software an den Kunden geliefert werden (zeigen die aktuellen Metriken, dass die Software die entsprechende Lieferqualität erreicht hat)?
- Führt die aktuelle Softwarequalität (Anzahl noch zu behebender Fehler) zu Problemen in der Wartungsphase oder bei der Weiterentwicklung der Software?

Produktqualitäts-Metriken

Die aus der Sicht des Softwareprodukts definierten Qualitätsmetriken haben eine besondere Stellung unter den Metriken. Mit ihnen können die Qualitätsmerkmale oder -teilmerkmale eines Softwareprodukts bewertet werden. Beispielsweise können Fragen bezüglich der Zuverlässigkeit, der Benutzbarkeit, der Verfügbarkeit oder des Zeit-Verhaltens einer Software beantwortet werden. Die Produktqualitäts-Metriken liefern zusätzlich Hinweise, worauf eine mangelnde Produktqualität zurückzuführen ist.

Geschäftsmetriken

Die Kategorie Geschäftsmetriken enthält Metriken, die sich auf die gesamte softwareentwickelnde Organisationseinheit beziehen, also neben dem Softwareentwicklungsprozess auch auf andere Prozesse, etwa im Marketing, Service und Management. Beispiele für solche

Metriken sind Kundenzufriedenheit, Mitarbeiterzufriedenheit, Innovationsfähigkeit, Anzahl der Schulungstage pro Mitarbeiter und Umsatzentwicklung. Die Geschäftsmetriken liefern keine direkten Aussagen zum Zustand der Softwareentwicklung und werden daher hier nicht ausführlich behandelt. Lediglich die Metriken Kundenzufriedenheit und Schulungsgrad werden aufgrund ihrer besonderen Bedeutung für die Softwareentwicklung in den Metrikkatalog aufgenommen (siehe Kapitel 8).

4.4 Auswertung der Ergebnisse

4.4.1 Visualisierung des Prozessstatus

Ziel der Visualisierung ist, die Ergebnisse der Prozessanalyse so darzustellen, dass der Beherrschungsgrad einer Organisationseinheit im Hinblick auf bestimmte Kernprozesse eindeutig ersichtlich ist. Es ist entscheidend, dass die qualitativen und die quantitativen Aussagen zum Prozessstatus insgesamt ein einheitliches und widerspruchsfreies Bild des Softwareentwicklungsprozesses wiedergeben.

Wird der Status des Entwicklungsprozesses unter Einsatz von Metriken quantitativ analysiert, so kann der Prozessstatus entsprechend der Definition der angewandten Metriken in Form von Diagrammen, Verteilungen oder Trenddarstellungen (chronologisch geordnete Folge von Messergebnissen) aufbereitet werden. Kapitel 8 (der Metrikkatalog) schlägt für jede Metrik, wenn angemessen, eine geeignete Darstellungsform vor.

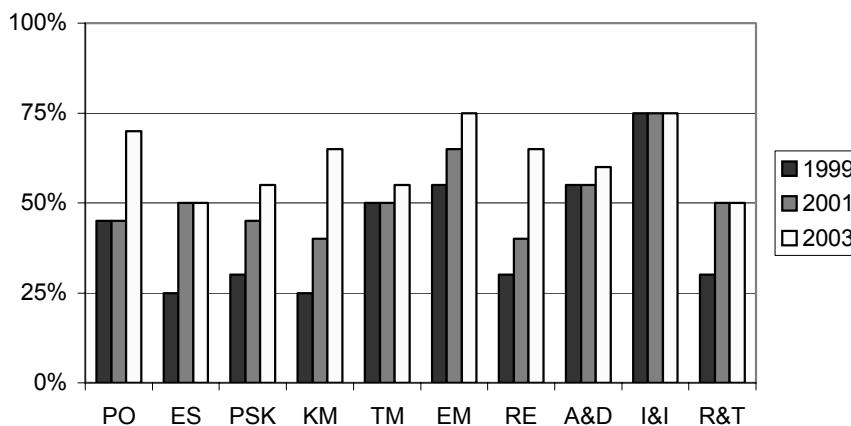
Erfolgt die Statusbestimmung anhand einer qualitativen Analyse, so wird ein Bewertungsschema benötigt, das eine Quantifizierung der Analyseergebnisse ermöglicht. MPTM stellt den Erfüllungsgrad eines Kernprozesses der Softwareentwicklung auf einer Skala von 0% bis 100% dar. Der höchste Erfüllungsgrad (100%) liegt vor, wenn der untersuchte Kernprozess gänzlich systematisch und methodengestützt vollzogen wird; mit anderen Worten, wenn die in Kapitel 7 gestellten Anforderungen an diesem Prozess voll erfüllt werden. Erfüllt der Prozess diese Anforderungen nicht, so hat dies eine entsprechende Abstufung des Beherrschungsgrads zur Folge (Abb. 4.3). Es ist zulässig, nach Bedarf zwischen den vorgegebenen Werten zu interpolieren.

Das arithmetische Mittel der Erfüllungsgrade aller Aspekte eines Kernprozesses ergibt den Erfüllungsgrad des Kernprozesses (siehe Abb. 9.3 in Abschnitt 9.3). Eine Gewichtung einzelner Prozessaspekte, entsprechend ihrer Bedeutung in einer Organisationseinheit, ist möglich. Der Erfühlungsgrad eines Kernprozesses wird dann über das gewichtete arithmetische Mittel errechnet.

Zusätzlich ist eine graphische Visualisierung des Prozessstatus sinnvoll. Sie hat den Vorteil, die seit der letzten Bewertung erzielte Prozessverbesserung komprimiert darstellen zu können (Abb. 4.4, siehe auch Abb. 9.3 und 9.4 in Abschnitt 9.3).

Erfüllungsgrad	Interpretation
0%	Kein fundiertes Vorgehen: Die Anforderungen an den Prozess werden nicht erfüllt.
25%	Teilweise Integration in die Entwicklungsaktivität: Der Prozess erfüllt teilweise die gestellten Anforderungen.
50%	Ansätze eines systematischen Vorgehens: Der Prozess erfüllt die Hälfte der gestellten Anforderungen.
75%	Gute Integration in die Entwicklungsaktivität: Der Prozess erfüllt einen wesentlichen Teil der gestellten Anforderungen.
100%	Vollständige Integration in die Entwicklungsaktivität: Der Prozess erfüllt die gestellten Anforderungen.

Abb. 4.3 Bewertungsschema zur Visualisierung des Prozessstatus



Legende:

- PO: Projektorganisation
 ES: Management externer Schnittstellen
 PSK: Planung, Steuerung und Kontrolle
 KM: Konfigurations- und Änderungsmanagement
 TM: Technologiemanagement
 EM: Entwicklungsmethodik
 RE: Requirements Engineering
 A&D: Architekturdefinition und Design
 I&I: Implementierung und Integration
 R&T: Qualitätssicherung: Review und Test

Abb. 4.4 Erfüllungsgrad der Kernprozesse im Zeitverlauf

4.4.2 Definition von Verbesserungsmaßnahmen

Die Definition von Verbesserungsmaßnahmen ist das Hauptziel einer Prozessbewertung. Nach einer erweiterten quantitativen Analyse muss jedes Messergebnis einzeln, im Zeitverlauf (bisherige Entwicklungsrichtung) und in Kombination mit anderen Messergebnissen ausgewertet werden. Aus dieser Auswertung können Schwachstellen bestimmter Kernprozesse erkannt und geeignete Maßnahmen zur Verbesserung der betroffenen Prozesse abgeleitet werden. Voraussetzung dafür ist, dass die Metriken zielorientiert definiert worden sind: Für jede Metrik ist also festgehalten, welche Fragen sie beantworten kann, mit anderen Worten, wie die aus ihrer Anwendung erzielten Ergebnisse zu interpretieren sind (siehe Metrikbeschreibungen in Kapitel 8).

Wird der Entwicklungsprozess qualitativ analysiert, so werden die Verbesserungsmaßnahmen auf der Basis des zugrunde liegenden Bewertungsmodells definiert. Alle Anforderungen, die das Modell an einen Prozess stellt, die aber von dem untersuchten Prozess nicht erfüllt werden, können als Verbesserungsmaßnahmen formuliert werden (zur Definition von Verbesserungsmaßnahmen als Ergebnis einer qualitativen Prozessanalyse siehe Abb. 9.2 in Abschnitt 9.3).

An dieser Stelle ist noch zu betonen, dass das Vorliegen eines Bewertungsmodells – im Falle qualitativer Bewertung – oder Metrikdefinitionen – im Falle quantitativer Bewertung – für die Definition von Verbesserungsmaßnahmen notwendig aber nicht hinreichend ist. Für die Definition qualitativ gute Verbesserungsmaßnahmen sind zudem Erfahrungen in Prozessbewertung und -verbesserung sowie Kenntnis der Rahmenbedingungen einer Organisationseinheit unerlässlich.

4.4.3 Identifikation der Verbesserungstreiber

Die Identifikation geeigneter Verbesserungstreiber ist entscheidend für eine zielorientierte und kostenoptimale Prozessverbesserung. Als *Verbesserungstreiber* werden dabei diejenigen Kernprozesse oder Prozessaspekte bezeichnet, deren Verbesserung eine positive Auswirkung auf die Zielerreichung hat. Sie bilden folgerichtig den Schwerpunkt in der sich anschließenden Prozessverbesserung.

Es wird bereits in der Voranalysephase, d.h. bei der Definition von Verbesserungszielen, damit begonnen, die Verbesserungstreiber zu identifizieren: Welcher Prozess lässt sich mit welchem Ziel sinnvoll in Verbindung bringen und wie groß ist sein Beitrag zur Zielerreichung? Die Antwort auf diese Frage nimmt während der Prozessanalyse immer mehr Gestalt an. Sie kann auch die Prozessanalyse in die eine oder andere Richtung intensivieren. Nach Abschluss der Prozessanalyse muss diese Frage beantwortet werden können.

Die Auswertung der Analyseergebnisse, insbesondere die festgestellten Schwachstellen im Prozess, bilden die Grundlage zur Identifikation von Verbesserungstreibern. Auch hier muss unterschieden werden, ob es sich um die Auswertung von Messergebnissen oder um die Auswertung qualitativer Analyseergebnisse handelt. Im ersten Fall liefert die Beobachtung der Trends (Entwicklung von Messergebnissen im Zeitverlauf) Hinweise, welche Prozesse

oder Prozessaspekte im Hinblick auf die Zielerreichung bedeutend sind und damit als Verbesserungstreiber in Frage kommen. Die Identifikation von Verbesserungstreibern in diesem Fall ist deshalb erleichtert, da jede Metrik mit mindestens einem Verbesserungsziel im Zusammenhang steht.

Hingegen ist die Identifikation von Verbesserungstreibern auf Grundlage qualitativer Analyseergebnisse schwieriger und fordert mehr Erfahrung in der Prozessbewertung und im Umgang mit solchen Ergebnissen. Hier muss unter den Prozessen mit den schwerwiegenden Schwachstellen (unter den Prozessen, die den Großteil der Anforderungen des Bewertungsmodells nicht erfüllen), der oder die Verbesserungstreiber identifiziert werden. Einige Prozesse lassen sich einem Verbesserungsziel direkt zuordnen: Der Beitrag des Prozesses zur Zielerreichung in solchen Fällen ist eindeutig und erfahrungsgemäß naheliegend. Bei anderen Prozessen müssen die Auswertungen eingehender untersucht und aufgrund der Rahmenbedingungen entschieden werden.

Nachdem die Prozesse den Zielen zugeordnet wurden und die Stärke ihres Beitrags zur Zielerreichung eingeschätzt wurde, lassen sich die Verbesserungstreiber leicht ausmachen: Das sind alle Prozesse, deren Verbesserung einen markanten Beitrag zur Erreichung der Ziele leistet. In der Regel ist es sinnvoller, die Verbesserungstreiber nicht unter den Prozessen, sondern unter den Prozessaspekten zu suchen. Durch diese differenzierte Betrachtung können die Schwerpunkte der Prozessverbesserung noch genauer bestimmt werden (zur Ermittlung der Verbesserungstreiber als Ergebnis einer qualitativen Prozessanalyse siehe Abb. 9.5 und 9.6 in Abschnitt 9.3).

Mit dem Abschluss der Identifikation von Verbesserungstreibern sind zugleich die Prioritäten der Verbesserungsmaßnahmen festgelegt: Alle Maßnahmen, die einen Bezug zu einem der Verbesserungstreiber haben, müssen mit höchster Priorität im Rahmen der anschließenden Prozessverbesserung umgesetzt werden.

Können durch die Analyse der Kernprozesse keine Verbesserungstreiber identifiziert werden, so besteht die Möglichkeit, anhand der verfügbaren Modelle, wie etwa CMMI for Software Engineering oder SPICE, andere Prozesse und Prozessaspekte in die Analyse einzubeziehen. Zuvor ist aber die Frage zu beantworten, ob die Behebung der auf diese Weise eventuell ermittelten Prozessschwächen noch ein gutes Nutzen/Kosten-Verhältnis aufweist, und ob nicht der Sättigungsgrad der Prozessverbesserung bereits erreicht wurde. Dann sollte auf eine weitere Prozessverbesserung verzichtet werden; gegebenenfalls ist eine Verbesserung der Softwareentwicklung durch Verbesserung der Technologien oder des Mitarbeitermanagements sinnvoller.

4.4.4 Definition von Metriken zur Kontrolle der Zielerreichung

Nachdem die Verbesserungstreiber festgelegt und den Verbesserungszielen zugeordnet worden sind, muss unmittelbar diskutiert werden, wie die Zielerreichung aus Management-Sicht beurteilt werden kann und welche Informationen dazu notwendig sind.

Zu diesem Zweck wird in jedem als Verbesserungstreiber identifizierten Prozess nach Merkmalen gesucht, durch die der Beitrag des betreffenden Prozesses zur Zielerreichung *gemessen* werden kann. Können solche Merkmale gefunden werden, so können geeignete Metriken definiert oder ausgewählt werden, womit zugleich der erste Schritt zu einer metrik-gesteuerten Bewertung und Verbesserung des betreffenden Prozesses getan ist.

Als Beispiel kann der Testprozess betrachtet werden. Hier sind Merkmale wie Fehlerfindung, Fehlerbehebung, Testdurchführung usw. messbar. Zu ihrer Messung dienen Metriken wie etwa Fehlerfindungsrate, Aufwand der Fehlerfindung und -behebung sowie Testfortschritt (für weitere Beispiele siehe Metrikkatalog, Kapitel 8). Es ist darauf zu achten, dass die Metriken aussagekräftig und einfach zu ermitteln sind.

Die nicht oder nur mühsam messbaren Merkmale eines Prozesses müssen weiterhin qualitativ analysiert und anhand ihres aktuellen Erfüllungsgrads beurteilt werden; von einer zwanghaften Definition von Metriken wird abgeraten.

4.5 Dokumentation der Bewertungsergebnisse

Nachdem der Entwicklungsprozess bewertet wurde, müssen die Ergebnisse in angemessener Breite und Tiefe in Form einer sachlichen Darstellung von Tatsachen dokumentiert werden. Die Dokumentation gilt als eine Art Anforderungsspezifikation für die anschließende Prozessverbesserung. Zum inhaltlichen Bestandteil dieser Dokumentation gehören:

- ein Überblick über die charakteristischen Eigenschaften der Organisationseinheit und der Projekte, die die Organisationseinheit in der Bewertung repräsentieren,
- die Auflistung der Verbesserungsziele,
- die Darstellung des aktuellen Prozessstatus, insbesondere im Hinblick auf die Schwachstellen des Entwicklungsprozesses,
- die Definition von konkreten Maßnahmen zur Behebung der Schwachstellen,
- die identifizierten Verbesserungstreiber sowie deren Zuordnung zu den Verbesserungszielen,
- die Metriken, die benötigt werden, um die Zielerreichung zu überwachen.

Ferner sollte die Dokumentation gründlich und verständlich gestaltet werden (Die Abschnitte 9.2 und 9.3 zeigen im Ganzen, wie eine solche Ergebnisdokumentation aussehen kann).

4.6 Voraussetzungen einer erfolgreichen Prozessbewertung

Im bisherigen Verlauf dieses Kapitels wurde der Ablauf einer Prozessbewertung behandelt. Die Einhaltung dieses Ablaufs alleine reicht nicht aus, um eine Prozessbewertung erfolgreich durchzuführen. Folgende Faktoren sind Voraussetzungen einer erfolgreichen Bewertung:

- **Planung**

Der zeitliche Ablauf aller Bewertungstätigkeiten und der daraus resultierende Aufwand müssen geplant werden. Der Plan stellt sicher, dass alle Dokumente und Personen zu vereinbarten Terminen zwecks Analyse oder Befragung zur Verfügung stehen. Die Analyse muss auf diejenige Dokumente oder Interview-Partner eingeschränkt werden, die qualifizierte Aussagen zum Entwicklungsprozess liefern können.

- **Teambildung**

Die Prozessbewertung sollte von einem Team von zwei Personen durchgeführt werden. Das ist erfahrungsgemäß die geeignete Teamgröße. Beide Personen müssen mit Softwareentwicklung und Prozessbewertung vertraut sein. Ein Team-Mitglied sollte ein Mitarbeiter der Organisationseinheit sein: Seine Kenntnis der internen Organisationsstrukturen hilft die Prozesse schneller zu verstehen. Das zweite Team-Mitglied sollte hingegen von außerhalb der Organisationseinheit stammen, damit ein unbefangener Blick auf das Entwicklungsgeschehen gewährleistet ist. Bei Bedarf kann das Team mit weiteren Experten verstärkt werden. Diese unterstützen das Team bei Klärung bestimmter Spezialfragen.

- **Offene Kommunikation**

Eine Prozessbewertung zeigt gegebenenfalls die Notwendigkeit einer Prozessverbesserung. Eine Organisationseinheit kann ihren Mitarbeitern diese Notwendigkeit dann vermitteln, wenn sie ihnen den Bewertungsablauf darlegt und insbesondere wenn sie die Bewertungsziele und -Ergebnisse ausführlich erörtert.