
Kollaborationspunkte im Software- entwicklungsprozess

Astrid Behm (Forschungszentrum Informatik, Karlsruhe)

Asarnusch Rashid (Forschungszentrum Informatik, Karlsruhe)

ZUSAMMENFASSUNG

Die vorliegende Arbeit beschäftigt sich mit der Strukturierung des Softwareentwicklungsprozesses in einer Art und Weise, die von der üblichen Gliederung abweicht. Es werden nicht die Softwareentwicklungsphasen und ihre Aktivitäten betrachtet, sondern die Punkte im Prozess untersucht, an denen Zusammenarbeit zwischen mehreren Personen stattfindet.

Diese Zusammenarbeit wird auch als Kollaboration bezeichnet und kann über die Faktoren *Kooperation*, *Kommunikation*, *Koordination* und *Kontext* beschrieben werden. *Kooperation* enthält die der Kollaboration zugrunde liegenden Vereinbarungen über die Zusammenarbeit zwischen den Parteien. *Kommunikation* beschreibt die Art und Weise, wie Informationen ausgetauscht werden. Mit *Koordination* wird die Regelung, wer wann was zu tun hat, bezeichnet und *Kontext* umfasst sowohl das organisationsbedingte als auch das persönliche Umfeld des einzelnen Mitarbeiters.

Basierend auf einem Rollenmodell wurden Gruppen von Akteuren (Arbeitsgruppen) identifiziert, zwischen denen Kollaboration stattfindet. Bei diesen Arbeitsgruppen unterscheidet man zwischen Anwender, Systemintegrator und Komponentenhersteller. Aufgabe des Systemintegrators ist die Entwicklung und Komposition von Software. Mit der Herstellung von neuen Komponenten befasst sich der Komponentenhersteller. Systemintegratoren zeichnen sich durch Kontakt zum Anwender aus, während Komponentenhersteller keinen direkten Kontakt zum Anwender haben.

Zwischen diesen Arbeitsgruppen gibt es im gesamten Softwareentwicklungszyklus fünf Kollaborationspunkte. Zwischen diesen Arbeitsgruppen gibt es im gesamten Softwarezyklus fünf Kollaborationspunkte:

- Kollaboration zwischen Anwendern
- Kollaboration zwischen Anwendern und Systemintegratoren
- Kollaboration innerhalb der Arbeitsgruppe(n) des Systemintegrators
- Kollaboration zwischen mehreren Gruppen von Systemintegratoren
- Kollaboration zwischen der Gruppe des Systemintegrators und der Gruppe des Komponentenherstellers

Die Identifizierung der Kollaborationspunkte ist der Ausgangspunkt für ein Referenzmodell, das den kollaborativen Softwareentwicklungsprozess sowohl unternehmensintern als auch unternehmensübergreifend beschreibt.

INHALTSVERZEICHNIS

1	Einleitung	4
2	Zum Begriff „Kollaborative Softwareentwicklung“	5
3	Arbeitsgruppen im Softwareentwicklungsprozess	6
4	Kollaborationspunkte im Softwareentwicklungsprozess	10
4.1	Kollaboration zwischen Anwendern.....	10
4.2	Kollaboration zwischen Anwendern und Systemintegrator.....	12
4.3	Kollaboration beim Systemintegrator.....	13
4.4	Kollaboration zwischen Produktionsteams unterschiedlicher Unternehmen .	14
4.4.1	Softwareerstellung im Auftrag	14
4.4.2	Softwareerstellung ohne direkten Auftrag	15
4.5	Kollaboration zwischen Systemintegrator und Komponentenhersteller	16
4.6	Kritische Untersuchung weiterer möglicher Kollaborationspunkte.....	16
5	Ausblick.....	16
	Literaturverzeichnis	18

1 EINLEITUNG

Im Rahmen des Forschungsverbundes „PRIMIUM“¹ untersucht das Forschungsprojekt „CollaBaWue“² moderne Technologien und Methoden für die komponenten- und semantikbasierte, kollaborative Softwareentwicklung für den Finanzdienstleistungsbereich in Baden-Württemberg. Ziel des Projekts ist eine Produktivitätssteigerung bei der Erstellung von Unternehmenssoftware im Finanzdienstleistungsbereich durch kollaborative und komponentenbasierte Softwareentwicklung.

Das Teilprojekt „Kollaborative Softwareentwicklung“ beinhaltet die Analyse der Unterstützung der Kollaboration durch geeignete Werkzeuge, aber auch die Untersuchung der Voraussetzungen, die im Team und im Unternehmen gegeben sein müssen, damit Kollaboration erfolgreich sein kann. Ziel ist es hierbei, ein geeignetes Referenzmodell zu entwickeln, das kollaborationsintensive Tätigkeiten im Softwareentwicklungsprozess aufzeigt und Hilfestellungen dazu gibt, wie diese Aktivitäten effizient und effektiv durchgeführt werden können.

Nach einer Studie von [Webs03] halten 50% der IT-Unternehmen in den Bereichen Requirements Engineering, Changemanagement und Entwurfsspezifikation die Kollaboration für wichtig. Ziel dieser Arbeit ist es aufzuzeigen, welche Charakteristiken die Kollaborationspunkte im Softwareentwicklungsprozess aufweisen, um diese als Ausgangsbasis für das oben erwähnte Referenzmodell festzulegen. Die vorliegende Arbeit stellt die identifizierten Kollaborationspunkte im (verteilten) Softwareentwicklungsprozess auf Basis von Arbeitsgruppen vor und beschreibt, unter welchen Voraussetzungen die Kollaboration zustande kommen kann. Jede kollaborative Aktivität im Softwareentwicklungsprozess, die in Vorgehensmodellen wie dem Rational Unified Process oder dem V-Modell XT beschrieben wird, kann einem dieser Kollaborationspunkte zugeordnet werden.

Die Arbeit gliedert sich im Folgenden in 5 Kapitel. Es folgt im zweiten Kapitel eine Definition des in dem Projekt verwendeten Begriffs „Kollaborative Softwareentwicklung“. Das Rollenmodell, das dem Referenzprozess zugrunde liegt wird im dritten Kapitel näher beschrieben. In Kapitel 4 werden die identifizierten Kollaborationspunkte im Detail dargestellt und die Arbeit schließt in Kapitel 5 mit einem Ausblick auf zukünftige, auf diesem Arbeitspapier aufbauende Arbeiten.

¹ <http://www.primium.org> (23.12.2005)

² <http://www.collabawue.de> (23.12.2005)

2 ZUM BEGRIFF „KOLLABORATIVE SOFTWAREENTWICKLUNG“

Altmann beschreibt die kooperative Softwareentwicklung [Altm99, S. 20] als

„die Abdeckung der Kommunikations- und Koordinationsbedarfe innerhalb eines Softwareentwicklungsprozesses, die für die Planung, Durchführung und Abstimmung aller aufgabenbezogenen, zeitlich und räumlich verteilten Aktivitäten erforderlich sind. Die kooperative Softwareentwicklung umfasst dementsprechend alle prozess- und produktbezogenen Aktivitäten aller Beteiligten, deren gemeinsames Ziel die Erstellung eines Softwareproduktes ist.“

Zusätzlich betont er den positiven Einfluss sozialer und organisatorischer Unterstützung der Mitarbeiter für eine erfolgreiche Zusammenarbeit. „Dies entspricht nahezu dem Verständnis von kollaborativer Softwareentwicklung im Projekt CollaBaWue, bei der die arbeitsteilige Zusammenarbeit in der Softwareentwicklung im Vordergrund steht. Hierfür muss jedoch die soziale und organisatorische Unterstützung nicht nur als positiver Einfluss, sondern vielmehr als essentieller Bestandteil der Zusammenarbeit verstanden werden.“ [RBHG05, S. 5]. Diese soziale und organisatorische Unterstützung der Mitarbeiter ist in Abbildung 1 als Kontext dargestellt, der sich wie ein Mantel um jeden einzelnen beteiligten Mitarbeiter legt. Jede beteiligte Person befindet sich sowohl in einem organisationsbedingten Kontext, der sich aus der Situation in ihrem Unternehmen und ihrer Abteilung ergibt, und in einem persönlichen Kontext, der aus ihren Qualifikationen, ihren individuellen Fähigkeiten und Vorlieben und auch aus ihrer privaten Situation resultiert.

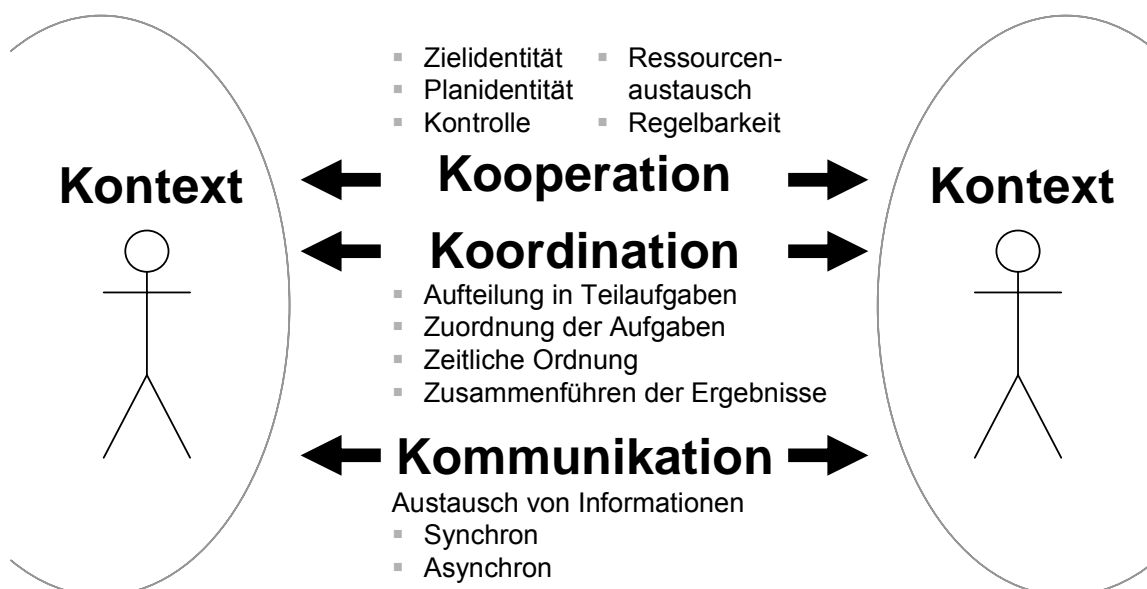


Abbildung 1: Elemente der Kollaboration in der Softwareentwicklung

3 ARBEITSGRUPPEN IM SOFTWAREENTWICKLUNGSPROZESS

Softwareentwicklung ist in Disziplinen eingeteilt, die parallel je nach Projektfortschritt unterschiedlich intensiv abgearbeitet werden. Als Beispiel lassen sich für die Disziplinen die Anforderungserhebung, die Implementierung oder das Projektmanagement anführen. Jede Disziplin bündelt Aktivitäten, die wiederum in Teilaktivitäten unterteilt sein können. Die zu entwickelnde Software sowie alle Dokumente, die dabei erstellt werden, werden als Produkt oder Artefakt bezeichnet und sind das Ergebnis einer oder mehrerer Aktivitäten. Die Ausführung dieser Aktivitäten erfolgt durch eine oder mehrere Rollen (siehe Abbildung 2).

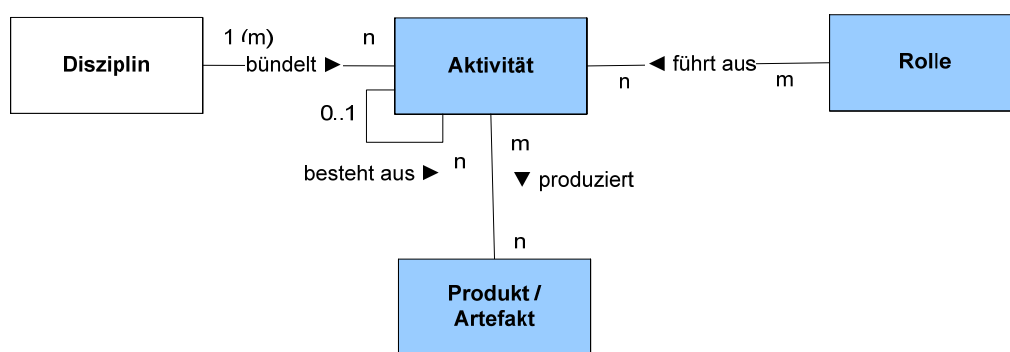


Abbildung 2: Prozessmetamodell der Softwareentwicklung

Anhand der Vorgehensmodelle Rational Unified Process (RUP) und dem V-Modell XT wurde eine Aufstellung der an der Softwareentwicklung beteiligten Rollen erstellt. Aus Abbildung 3, die einen Überblick über die identifizierten Rollen gibt, ist ersichtlich, dass die Rollen grundsätzlich einer von vier Gruppen zugeordnet werden können.

Bei der Gruppe der **Anwendenden Mitarbeiter** handelt es sich um diejenigen Rollen, die den Anstoß zu dem Softwareprojekt geben, das Projekt finanzieren, die Systeme administrieren und letztendlich den Nutzen für ihre tägliche Arbeit aus der neuen Software ziehen. Entscheidungsträger des Projektes gehören zu dieser Gruppe.

Der Gruppe der **Leitenden Mitarbeiter** sind diejenigen Rollen zuzuordnen, die für die korrekte Projektdurchführung verantwortlich sind. Ihre Aufgabe ist die Koordination des Projektteams und die Planung einer sinnvollen Umsetzung, der vom Anwender vorgegebenen Anforderungen. Im Rahmen dieser Vorgaben werden Entscheidungen von dieser Gruppe selbstverantwortlich getroffen.

In der Gruppe der **Produktionsmitarbeiter** sind alle Rollen zusammengefasst, die mit der eigentlichen Erstellung des Softwareprodukts beschäftigt sind.

Angehörige der Gruppe der **Einkaufs-/Vertriebsmitarbeiter** sind für die betriebswirtschaftliche und organisatorische Unterstützung des Projektteams zuständig.

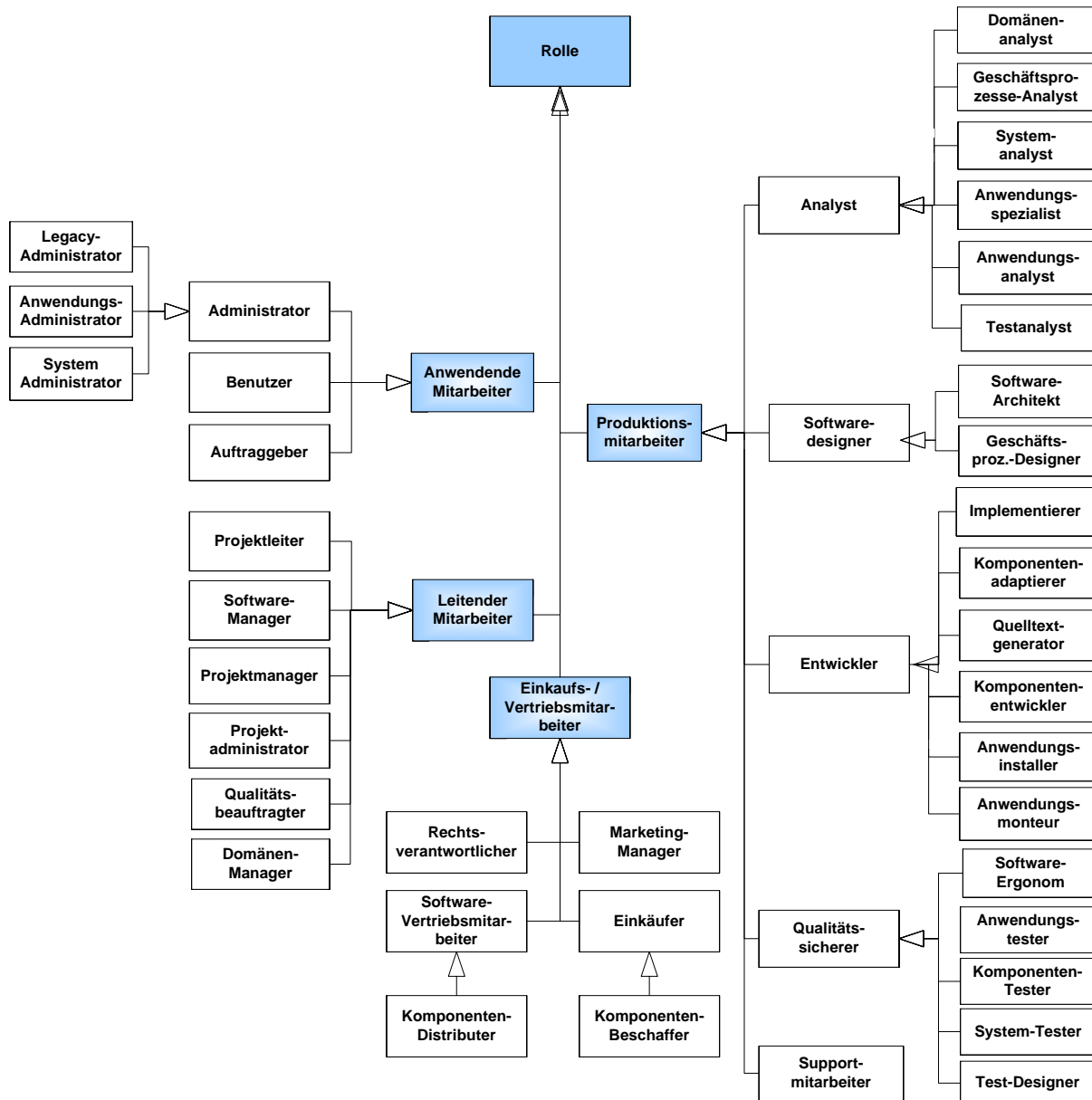


Abbildung 3: Meta-Rollenmodell in der Softwareentwicklung

Abbildung 4 stellt die Verbindung zwischen der Rolle aus dem Prozessmetamodell der Softwareentwicklung und dem Unternehmen her. Akteure arbeiten bei einer Organisation und können Mitglied in einer oder mehreren Arbeitsgruppen sein. Jeder Akteur nimmt dabei eine oder mehrere Rollen ein. Eine Organisation kann z.B. aus mehreren Unternehmen bestehen. Somit kann sich beispielsweise für die Rolle des Geschäftsprozessanalysten folgende Konstellation ergeben: Unternehmen A und Unternehmen B arbeiten gemeinsam an der Erstellung einer Software und beauftragen Frau X (Unternehmen A) und Herrn Y (Unternehmen B) mit der Analyse der Geschäftsprozesse. Die Rolle wird folglich durch mehrere Akteure aus unterschiedlichen Unternehmen besetzt.

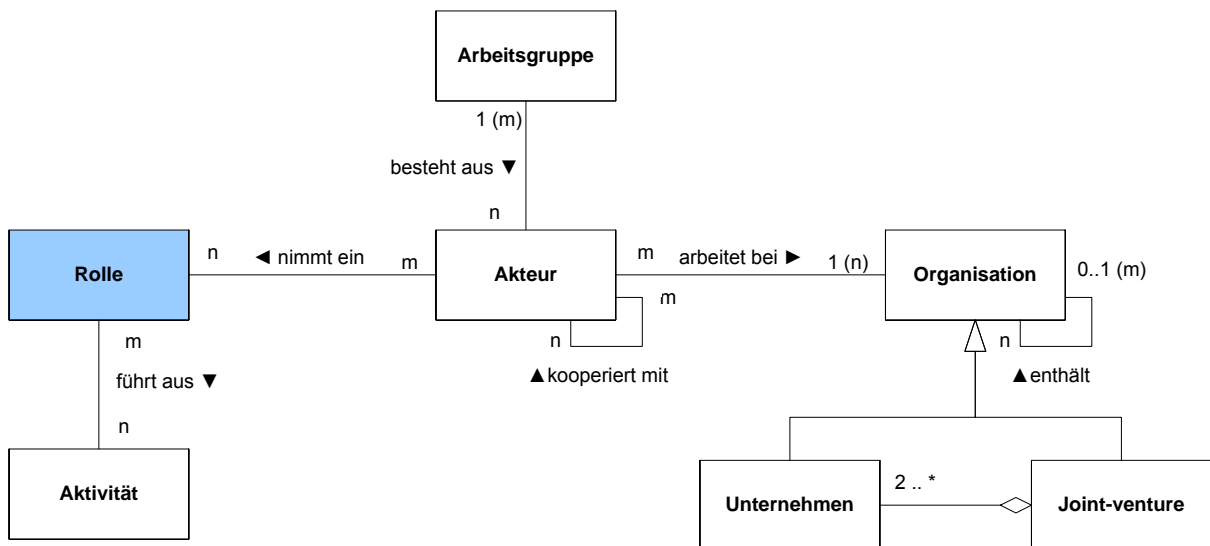


Abbildung 4: Die Rolle im Organisationsumfeld

Zu einem kollaborativen Prozess kommt es auf der Ebene, der an der Softwareentwicklung beteiligten Personen, wie in der Definition in Kapitel 2 vorgestellt. Diese nehmen dabei eine der im Meta-Rollenmodell beschriebenen Rollen ein. Mit dem Ziel, die Kollaboration im Softwareentwicklungsprozess zu strukturieren und einen Überblick über die Punkte zu geben, an denen Kollaboration prinzipiell stattfinden kann, wird ein Ansatz gewählt, der die Kollaboration zwischen den beteiligten Gruppen von Akteuren untersucht. Eine Identifikation von Kollaborationspunkten aus der schier unübersichtlichen Zahl von Einzelaktivitäten in der Softwareentwicklung bietet hierfür keinen Mehrwert und wird deshalb nicht weiter verfolgt. Die Gruppierung von Akteuren erfolgt entsprechend ihrer Hauptaufgabe im Projekt. Dabei kristallisieren sich die in Abbildung 5 dargestellten Arbeitsgruppen *Anwender*, *Systemintegrator* und *Komponentenhersteller* heraus.

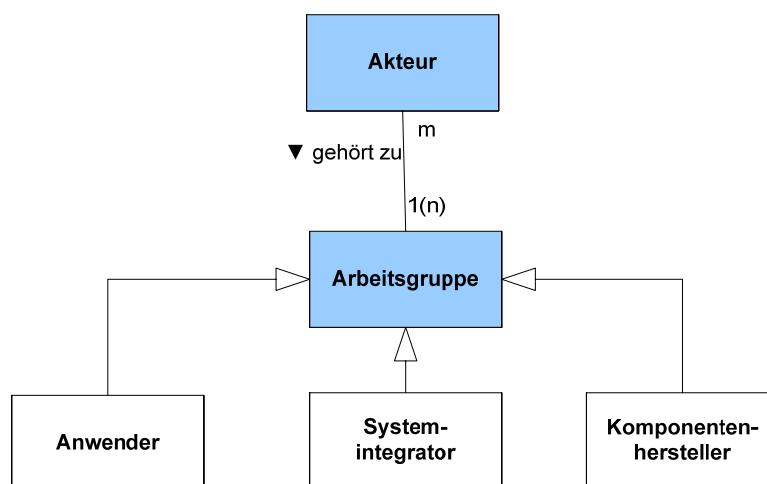


Abbildung 5: Arbeitsgruppen

Frau X und Herr Y, denen im bereits genannten Beispiel die Rolle des Geschäftsprozessanalysten zugewiesen wurde, können nun aufgrund dieses Ansatzes jeweils

4 KOLLABORATIONSPUNKTE IM SOFTWAREENTWICKLUNGSPROZESS

Zwei – eigentlich unterschiedliche - Zitate treffen den Kern der Kollaboration, den diese Arbeit adressiert:

„Software engineering is inherently a team-based activity.“ [CoCh03, S. 290] und

„The truth is that even in tightly knit task-teams the members perform the majority of activities separately.“ [Andr03, S. 6]

Auch wenn viele Arbeiten durch einzelne Personen durchgeführt werden, so treten im Softwareentwicklungsprozess immer wieder Aktivitäten auf, die nur in Zusammenarbeit mit anderen Personen erfolgreich abgewickelt werden können. Der Austausch von Informationen in diesen Aktivitäten ist entscheidend für den Projekterfolg, da die Abarbeitung der nachfolgenden Aktivitäten von der korrekten Verarbeitung der ausgetauschten Informationen abhängt.

Das folgende Kapitel beschäftigt sich mit der Kollaboration zwischen Arbeitsgruppen, wie sie in Kapitel 3 vorgestellt wurden. Zwischen diesen Arbeitsgruppen wurden fünf Kollaborationspunkte identifiziert, die in

Abbildung 7 dargestellt und in diesem Kapitel detailliert beschrieben werden. Die einzelnen kollaborativen Aktivitäten des Softwareentwicklungsprozesses können jeweils einem der vorgestellten Kollaborationspunkte zugeordnet werden.

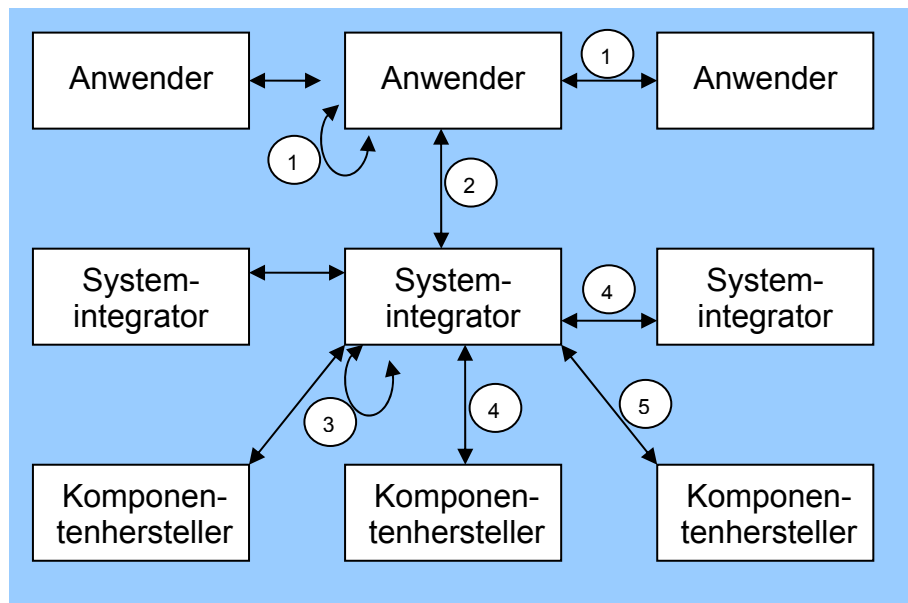


Abbildung 7: Kollaborationspunkte in der Softwareentwicklung

4.1 Kollaboration zwischen Anwendern

Kollaboration zwischen Anwendern findet statt, wenn mehrere Anwender gemeinsam z.B. eine Anforderungsspezifikation erstellen. Hier wird unterschieden zwischen der Kollaboration von Anwendern im selben Unternehmen und der Kollaboration von Anwendern aus unterschiedlichen Unternehmen. Im zweiten Fall ist das Zustandekommen einer Kollaboration in starkem Maße davon abhängig, ob sich die Unternehmen, denen die Anwender angehören, in einer Konkurrenzsituation gegenüberstehen oder nicht.

Im Rahmen des Projektes CollaBaWue wurden Interviews mit Vertretern mehrerer namhafter Banken in Baden-Württemberg geführt. Auch wenn aufgrund der Anzahl der Interviewpartner keine Repräsentativität für die ausgewerteten Ergebnisse abgeleitet werden kann, so erlauben sie doch eine Beschreibung der aktuellen Situation in der Softwareentwicklung bei Finanzdienstleistungsunternehmen in Baden-Württemberg, die im folgenden näher vorgestellt wird.

Im Finanzdienstleistungssektor wird in zwei Kategorien Software unterschieden – in Standardsoftware (Infrastruktur und Commodity), welche die Software umfasst, die für die Aufrechterhaltung des laufenden Betriebs und die Einhaltung der gesetzlichen Vorschriften benötigt wird, und in Individualsoftware. Benötigt der Anwender eine Software aus dem Bereich der Standardsoftware, so wird diese am Markt, so verfügbar, zugekauft und eingeführt. Eine Zusammenarbeit mit Anwendern aus anderen Finanzdienstleistungsunternehmen wird nicht zustande kommen. Ist die benötigte Software jedoch nicht verfügbar, so ist durchaus der Fall denkbar, dass mehrere Finanzdienstleistungsunternehmen versuchen, einen Hersteller von Standardsoftware davon zu überzeugen, die benötigte Software gemeinsam mit den Anwendern zu entwickeln. Eine gemeinsame Entwicklung bedeutet in diesem Kontext, dass der Hersteller ebenfalls einen Teil der Entwicklungskosten übernimmt. In diesem Fall ist eine Kollaboration unter Anwendern mehrerer Finanzdienstleistungsunternehmen

durchaus realistisch, da bei Standardsoftware die Konkurrenzsituation sekundär ist. Eine Kollaboration unter den Anwendern des Finanzdienstleistungsunternehmens selbst findet immer statt, da sie gemeinsam die Einführung der neuen Software im Unternehmen bewältigen müssen.

Individualsoftware wird für die Vermarktung und Abwicklung institutseigener Produkte, mit deren Hilfe sich das Finanzdienstleistungsunternehmen von seinen Konkurrenten abheben möchte, benötigt. Der Einsatz von Individualsoftware dient folglich der Schaffung eines Marktvorteils gegenüber seinen Mitstreitern. Dieser Individualsoftware liegt Wissen zugrunde, das wegen der Konkurrenzsituation nicht mit anderen Unternehmen geteilt werden kann. Es kann demzufolge auch nicht zu einer Kollaboration mit Anwendern aus anderen Unternehmen kommen. In diesem Fall ist die Kollaboration der Anwender eines Unternehmens noch ausgeprägter als bei der Einführung der Standardsoftware, da gemeinsam die Anforderungen spezifiziert und die neuen Abläufe in das Tagesgeschäft integriert werden müssen.

Es lässt sich folglich die Frage ableiten, ob es überhaupt eine Situation im Bereich der Finanzdienstleistungsunternehmen gibt, in der es zu einer Kollaboration unter Anwendern unterschiedlicher Unternehmen bei der Entwicklung von Individualsoftware kommen kann. Diese Situation kann dann eintreten, wenn die Unternehmen unterschiedliche Kundengruppen bzw. Zielgruppen anvisieren. Die Vorteile einer solchen Kollaboration sind dieselben wie beim Einsatz von Standardsoftware im Vergleich zur Eigenentwicklung. Dies ist darauf zurückzuführen, dass für den Softwarehersteller mehr als eine Kundenverbindung an dem Projekt hängt. Vorteile sind beispielsweise Kostenersparnisse durch eine größere Anzahl der Abnehmer und die Gewährleistung der Programmwartung und –weiterentwicklung durch den Softwareentwickler [Krcm05, S.137]. Darüber hinaus kann aus dieser Kollaboration ein Qualitätsgewinn resultieren, da sowohl in die Spezifikation als auch in den Test das Know How von mehreren Unternehmen einfließt. Diese Situation ist in zahlreichen Organisationen zu beobachten:

- im genossenschaftlichen Verbund, der Sparkassenorganisation
- im Vorfeld geplanter Fusionen und Outsourcing, sowie
- bei kleineren und mittleren Finanzdienstleistungsunternehmen, wie z.B. freie Versicherungsmakler, die evtl. unter einer Dachorganisation organisiert sind.

Hinderlich für eine Kollaboration ist immer der damit verbundene erhöhte Verwaltungs- bzw. Koordinationsaufwand. Darüber hinaus haftet dem Ergebnis einer Kollaboration derselbe Nachteil an, der beim Einsatz von Standardsoftware auftritt, nämlich die evtl. „unvollständige Abdeckung unternehmensspezifischer Anforderungen, sowie die unvollständige Integration in die Gesamtheit bereits im Unternehmen implementierter Anwendungen z.B. wegen Schnittstellenproblemen und durch Orientierung an allgemeiner Verwendbarkeit eventuell schlechtes Betriebsverhalten in unternehmensspezifischen Situationen“ [Krcm05, S.137].

4.2 Kollaboration zwischen Anwendern und Systemintegrator

Die Entwicklung der Vorgehensmodelle und Methoden in der Softwareentwicklung zeigt, dass der Anwender immer stärker in den Entwicklungsprozess integriert wird,

da die Einhaltung strenger Phasen in der Softwareentwicklung, wie es ursprünglich im Wasserfallmodell vorgesehen war, nicht garantiert werden kann. Die endgültigen Systemanforderungen kristallisieren sich erst mit wachsendem Erkenntnisstand im Verlauf des Entwicklungsprozesses heraus und äußere Einflüsse bedingen schon während der Systementwicklung wiederholt Änderungen der Anforderungen [Stah05, S.218f.]. Die Softwareentwicklung erfolgt daher in Zyklen, so dass der Anwender immer wieder mit den Aufgaben konfrontiert wird, noch offene Spezifikationen zu konkretisieren bzw. Tests durchzuführen und Teilabnahmen zu machen. Im Rahmen der agilen Softwareentwicklung wird sogar davon ausgegangen, dass der Anwender in seiner Eigenschaft als Experte über die gesamte Laufzeit der Softwareentwicklung als Ansprechpartner ständig zur Verfügung steht [Cock03, S.224].

Als Problem an diesem Kollaborationspunkt ergibt sich z.B., dass der Benutzer in der Regel nicht Technik affin ist und eine „andere Sprache“ spricht als die Mitarbeiter des Systemintegrators und somit nicht von einer gemeinsamen Wissensbasis der Kollaborationsbeteiligten ausgegangen werden kann. In den Vorgehensmodellen wird der Versuch unternommen, diese Problematik mit der Erstellung eines Glossars zu beseitigen. Auch können z.B. aus in EPK³ formulierten Geschäftsprozessen UML-Modelle erzeugt werden, so dass das Problem der „anderen Sprache“ abgeschwächt werden kann. Da jedoch die Geschäftsprozesse nicht von dem Benutzer selbst, sondern in Zusammenarbeit mit einem Analysten erstellt werden, kann hierdurch nicht das Problem im Ganzen gelöst werden. Selbst, wenn der Systemintegrator und die Benutzer im selben Unternehmen angestellt sind, arbeiten sie trotzdem meistens in einem unterschiedlichen Kontext, der sich z.B. in verschiedenen Arbeitsweisen, Interessenslagen und Sichtweisen niederschlägt und sich aus unterschiedlicher Aufgabenstellung und Einbindung in den betrieblichen Produktionsprozess ergibt.

Der Kollaborationspunkt zwischen Anwender und Systemintegrator ist eine 1:1 Verbindung, da es sich um das Verhältnis Auftraggeber und Auftragnehmer handelt. Es ist aus Sicht der praktischen Ausgestaltung vor allem hinsichtlich der oben beschriebenen Probleme dabei unerheblich, ob der Auftraggeber im selben Unternehmen tätig ist wie der Auftragnehmer oder nicht. Die Kommunikation erfolgt zielgerichtet über einen festgelegten Kommunikationskanal. Stehen auf der Seite der Anwender mehrere Gruppen u. U. aus unterschiedlichen Unternehmen, so wird davon ausgegangen, dass die Gruppe der Anwender einen Sprecher hat, über den die Kommunikation abgewickelt wird. Die Kommunikation innerhalb der Gruppe fällt dann wieder unter den Kollaborationspunkt 1, der die Kollaboration zwischen Anwendern und Anwendergruppen beschreibt.

Denkbar ist ebenfalls die Konstellation, dass ein Anwender mit mehreren Systemintegratoren parallel kommuniziert, wenn ein Softwaresystem aus Modulen unterschiedlicher Hersteller erstellt werden soll. Dieser Fall wird wie eine Vielzahl von Einzelprojekten betrachtet, bei denen jeweils Kollaboration zwischen einem Anwenderteam und einem Systemintegrator stattfindet.

Wenn die Kommunikationssituation aufgrund geographischer, kultureller und / oder organisatorischer Gegebenheiten erschwert wird (z.B. Outsourcing, Offshoring der Softwareentwicklung [HeMo01]), so ist davon auszugehen, dass die Kommunikation

³ Ereignisgesteuerte Prozessketten

nicht direkt zwischen Anwender und Systemintegrator erfolgt. Denkbar ist in diesem Fall, dass ein produktionsnaher Mitarbeiter aus dem eigenen Unternehmen eine Vermittlungsrolle übernimmt. Die Form der Kollaboration fällt somit unter den Kollaborationspunkt 4.4 „Kollaboration zwischen Produktionsteams unterschiedlicher Unternehmen“.

4.3 Kollaboration beim Systemintegrator

Bei diesem Kollaborationspunkt handelt es sich um die Zusammenarbeit innerhalb eines Produktionsteams, d.h. dem Team, das für das Design und die Implementierung zuständig ist. Dies kann sowohl im Rahmen eines gemeinsamen Projekts sein, als auch projektübergreifend, wenn z.B. für ein Anschlussprojekt eine Projektübergabe stattfinden muss, oder einfach nur Wissen über ähnlich gelagerte Probleme ausgetauscht wird.

Bei der Kollaboration unter Softwareentwicklern in einem Unternehmen kann davon ausgegangen werden, dass alle eine „gemeinsame Sprache“ sprechen, eine gemeinsame Wissensbasis vorliegt, die Organisationsform modular und somit meist flach und offen gestaltet und das Umfeld kommunikationsfreundlich ist. [Pico03, S.234] Bei der näheren Untersuchung dieses Kollaborationspunktes können daher viele Rahmenbedingungen, die bei der Kollaboration an der Anwender beteiligt sind, Relevanz haben, vernachlässigt werden. Ein ausschließlich werkzeugorientierter Ansatz ist jedoch auch hier nicht ausreichend, denn Forschungsergebnisse aus dem soziologisch-technischen Umfeld, wie z.B. die Erkenntnisse über Informationspathologien, sind auch an dieser Stelle zutreffend.

Das Wissensmanagement [RuBo99], das (Multi-)Projektmanagement und das Konfigurationsmanagement [Grin95] sind kollaborative Tätigkeiten mit großen Auswirkung auf die Effektivität eines Produktionsteams, so dass der Lösung dieser Aufgaben eine hohe Beachtung bei der Auswahl der Kollaborationswerkzeuge zukommen muss.

4.4 Kollaboration zwischen Produktionsteams unterschiedlicher Unternehmen

Grundsätzlich muss bei diesem Kollaborationspunkt unterschieden werden, ob es sich um Softwareerstellung im Auftrag (Kollaboration zwischen Systemintegrator und Komponentenhersteller) handelt, oder ob mehrere Systemintegratoren gemeinsam an der Entwicklung eines Produktes arbeiten, wobei der Markt die Rolle des Anwenders übernimmt.

Der Kontext, in dem die Softwareentwickler bei diesem Kollaborationspunkt arbeiten, ist dem des dritten Kollaborationspunktes („Kollaboration beim Systemintegrator“, s. S. 13) ähnlich, die Kollaboration kann aber durch einen oder mehrere der folgenden Punkte erschwert werden:

- Strategische Ausrichtung der Unternehmen
- unterschiedliche technische Voraussetzungen
- unterschiedliche kulturelle Voraussetzungen
- Konkurrenzsituation auf dem Markt

4.4.1 Softwareerstellung im Auftrag

Der Auftraggeber legt in der Regel Wert darauf, nur einen Ansprechpartner für seinen Auftrag zu haben, der Systemintegrator dagegen möchte meist nicht die Gesamtverantwortung für das System übernehmen, so dass er versuchen wird, Verträge zwischen dem Auftraggeber und dem Komponentenhersteller zu vermitteln. Wenn der Systemintegrator die Gesamtverantwortung übernimmt, wird er vom Auftraggeber berechtigt, einen Komponentenhersteller in festgelegtem Umfang zu beauftragen oder Komponenten zuzukaufen, falls dies erforderlich oder gewünscht ist. Aus Sicht der Softwareerstellung findet die Kollaboration in beiden Fällen zwischen dem Systemintegrator und dem Komponentenhersteller statt, da es sich um eine Zusammenarbeit in Bezug auf fachliche Inhalte handelt (Schnittstellenanpassungen, Spezifikationen der Komponenten, Unterauftragsvergabe), ungeachtet der vertraglichen Regelung.

Auch wenn die Auftragsvergabe an ein virtuelles Unternehmen erfolgt, hat der Auftraggeber grundsätzlich nur einen Ansprechpartner. Der Untersuchung von Gabriel und Kerlen [GaKe02, S.85ff] lässt sich entnehmen, dass die Führung dann in der Regel bei dem das jeweilige Projekt akquirierenden Unternehmen liegt, das Unteraufträge an die anderen Partnerunternehmen vergibt. Die Partnerunternehmen sind dabei rechtlich und wirtschaftlich selbständig. „Der Nutzen von virtuellen Unternehmen wird darin gesehen, dass sich die Vorteile kleiner, flexibler und kostengünstiger Unternehmenseinheiten verbinden lassen mit der Möglichkeit, größere Aufträge abwickeln zu können, da die erforderlichen Kompetenzen passend zum Volumen eines Auftrags zusammengeführt werden“ [GaKe02, S. 86].

Im Falle einer Unterauftragsvergabe (Systemintegrator vergibt Unterauftrag an einen anderen Systemintegrator) übernimmt der Systemintegrator des ursprünglichen Auftrags die Rolle des Anwenders. Die Mehrstufigkeit in der Unterauftragsvergabe lässt sich auf diese Weise im Modell iterativ abdecken.

Diese Beziehung wird durch die folgenden Merkmale charakterisiert:

- Rahmenvertrag (Kooperation)
- Vorgehen ist geregelt / evtl. Anwendung eines Vorgehensmodells
 - Beauftragung / Lieferung – Austausch von Artefakten
 - Zuständigkeiten
- Statusverfolgung
- permanenter Austausch, der unter Umständen mehr formale als fachliche Elemente enthält
 - geringer Projektumfang: Beauftragung – Lieferung
 - großer Projektumfang: Iterationen mit ständigem Austausch

Wenn das Produkt aus einer Beauftragung resultiert ist eine Kollaboration zwischen Produktionsteams weiterer Unternehmen unüblich, da der Auftraggeber aus Gründen der Gewährleistung und der Haftung einen einzigen Vertragspartner bevorzugt.

4.4.2 Softwareerstellung ohne direkten Auftrag

Entwicklungen ohne Kundenauftrag von mehreren Softwareentwicklern gemeinsam, werden meist in der Kooperationsform der Coopetition [BrNa95, ThLo04] durchge-

führt. „Coopetition ist eine Wortschöpfung aus dem Englischen (Mischung aus Competition und Cooperation) und bedeutet „kooperatives Konkurrieren“. Die Coopetition zielt mehr darauf hin, aus Kooperationen in erster Linie eigene Vorteile zu erlangen als gemeinsame Ziele zu verfolgen. [...] Es gilt, den eigenen Mehrwert zu maximieren und den der anderen Marktteilnehmer zu begrenzen. Um dies durchzusetzen, können Verträge und herrschende Gesetze als Regeln herangezogen werden“ [ThLo04]. Zu Kooperationen kommt es dabei vorwiegend in internen Bereichen wie z.B. Forschung, Entwicklung oder Produktion, während die Unternehmen am Markt weiterhin als eigenständige und unabhängige Unternehmen auftreten und im Wettbewerb miteinander stehen.

Coopetitions können entstehen, wenn sich mehrere kleine, konkurrierende Unternehmen zu einem Konsortium zusammenschließen, um gemeinsam ein Großprojekt durchführen zu können, das keines der Unternehmen alleine bewältigen könnte. Ein anderer Grund für die Entstehung einer Coopetition ist der Bedarf nach einer Einigung auf bestimmte Regeln oder Standards, die notwendig sind, damit die an der Coopetition beteiligten Unternehmen ihre angebotenen Produkte besser am Markt durchsetzen oder Netzwerkeffekte nutzen können.

Coopetitions sind z.B. im Rahmen von Open Source Communities bekannt. Die Entwicklung von OS/2 1984 durch Microsoft und IBM ist ebenfalls ein gutes Beispiel für diese Konstellation. Auch wenn das Betriebssystem nicht die ursprünglich geplante Verbreitung gefunden hat und letztendlich wieder verschwand, so war es doch ein lauffähiges System.

4.5 Kollaboration zwischen Systemintegrator und Komponentenhersteller

Wie in Kapitel 4.4.1 bereits ausgeführt wurde, kann ein Systemintegrator vom Auftraggeber berechtigt werden, am Markt verfügbare Komponenten zu kaufen und in das System zu integrieren. Die Kollaboration erfolgt in diesem Fall nicht nur direkt zwischen Vertretern der beteiligten Gruppen, sondern auch über ein Repository bzw. ein Portal über das der Komponentenhersteller seine für den Markt erstellten Komponenten anbietet. Zentrale Aktivitäten sind das Auffinden der Komponenten, das Überprüfen, ob die Funktionalität der angebotenen Komponente sich mit den vorliegenden Anforderungen deckt, und ein möglichst problemloses Integrieren der Komponente in das zu erstellende System. Dies setzt eine standardisierte, formale Beschreibung der Komponente voraus, über die entsprechende Suchalgorithmen fündig werden können.

Ein Kontakt zwischen Vertretern des Systemintegrators und dem Komponentenhersteller kommt nur dann zustande, wenn Fragen oder Probleme auftreten oder wenn Änderungen an der Komponente benötigt werden. In letzterem Fall würde die weitere Kollaboration dann unter den vierten Kollaborationspunkt „Kollaboration zwischen Produktionsteams unterschiedlicher Unternehmen“ fallen.

4.6 Kritische Untersuchung weiterer möglicher Kollaborationspunkte

Komponentenhersteller und Systemintegratoren sind grundsätzlich erst einmal Softwarehäuser, die Software für ihre Auftraggeber oder den Markt herstellen bzw. zusammenpassende Systeme integrieren. Ein Softwarehaus kann dabei sowohl Komponentenhersteller als auch Systemintegrator sein. Wenn mehrere Softwarehäuser an einem Produkt zusammen arbeiten, wird es immer mindestens ein Softwarehaus unter den kollaborierenden Partnern geben, das eine integrierende Funktion übernimmt. Dieses Softwarehaus agiert in diesem Fall nicht als Komponentenhersteller sondern als Systemintegrator. Das gilt auch, wenn es für dasselbe Produkt zu einem späteren Zeitpunkt - beispielsweise wenn das Produkt nach der Fertigstellung über ein Repository angeboten wird - als Komponentenhersteller auftreten wird. Eine Kollaboration unter Komponentenherstellern kann es somit nicht geben.

5 AUSBLICK

Kollaboration in der Softwareentwicklung kann und muss von Werkzeugen unterstützt werden. Im Rahmen des Projektes CollaBaWue ist es geplant, ein Modell zur Situationsanalyse zu entwickeln, das zum einen als Hilfestellung bei der Untersuchung von Konstellationen, in denen Kollaboration stattfindet, dienen soll, zum anderen aber bei der Einführung von Werkzeugen Unterstützung bieten kann. Anhand dieses Modells zur Situationsanalyse soll basierend auf der Betrachtung des Kontextes (vgl. Kapitel 2) ermittelt werden können, welches Tool bzw. ob ein ausgewähltes Tool zur Unterstützung einer konkreten Kollaborationssituation für die Erfüllung der Anforderung im Unternehmen bzw. unternehmensübergreifend eingesetzt werden kann.

Als Grundlage für dieses so genannte Situationsanalysemodell wird in einem weiteren Arbeitspapier der Begriff des „Kontextes“ näher beleuchtet und dessen Relevanz für die kollaborative Softwareentwicklung belegt werden. Basierend auf dem Situationsanalysemodell und den fünf Kollaborationspunkten in der Softwareentwicklung soll dann versucht werden, ein Referenzmodell für die kollaborative Softwareentwicklung zu beschreiben.

LITERATURVERZEICHNIS

- [Altm99] Altmann, J.: Kooperative Softwareentwicklung – Rechnerunterstützte Koordination und Kooperation in Softwareprojekten. Dissertation, Universitätsverlag Rudolf Trauner, Linz 1999.
- [Andr03] Andriessen, J.H. E.: Working with Groupware: Understanding and Evaluating Collaboration Technology. Springer-Verlag, London 2003.
- [ArDi06] Arndt, J.; Dibbern, J.: "The Tension between Integration and Fragmentation in a Component Based Software Development Ecosystem", Proceedings of the Hawaii International Conference on Computer Science (HICCS), Kauai (Hawaii), 2006.
- [BrNa95] Brandenburger, A. M.; Nalebuff, B. J.: Co-opetition. Currency-Doubleday, New York 1996.
- [CoCh03] Cook, C.; Churcher, N.: An Extensible Framework for Collaborative Software Engineering. In: Proceedings of the Tenth Asia-Pacific Software Engineering Conference (APSEC'03), Chiang Mai, Thailand 2003.
- [Cock03] Cockburn, A.: Agile Software-Entwicklung. Verlag Moderne Industrie, 2003.
- [GaKe02, S. 86] Gabriel, P.; Kerlen, C.: Die Besonderheiten der Kooperation von Kleinunternehmen in ,virtuellen Unternehmen. In: Herczeg, M.; Prinz, W.; Oberquelle H. (Herausgeber): Mensch & Computer 2002 : Vom interaktiven Werkzeug zu kooperativen Arbeits- und Lernwelten. Berichte des German Chapter of the ACM, Band 56, 2002.
- [Grin95] Grinter, R.E.: Using a Configuration Management Tool to Coordinate Software Development. Proceedings of the Conference on Organizational Computing Systems COOCS'95, Milpitas (CA, USA), 1995.
- [HeMo01] Herbsleb, J.; Moitra, D.: Global Software Development. IEEE Software März/April 2001, IEE Computer Society, 2001.
- [Krcm05] Krcmar, H.: Informationsmanagement. 4. Auflage, Springer Verlag, 2005.
- [Mess03] Messerschmidt David G. / Szyperski Clemens, Software Ecosystem, MIT Press, 2003.
- [Pico03] Picot, Arnold / Reichwald, Ralf / Wigand Rolf T.: Die grenzenlose Unternehmung. Gabler Verlag, 2003.
- [RBHG05] Rashid Asarnusch; Behm, A.; Hildenbrand, T.; Geisser, M.: Kollaborative Softwareentwicklung – Zum Kollaborationsbegriff. Arbeitspapier des FZI Forschungszentrum Informatik an der Universität Karlsruhe, 2005.
- [RuBo99] Ruhe, G.; Bomarius, F.: Learning Software Organizations – Methodology and Applications. Proceedings of the 11th International Conference on Software Engineering and Knowledge Engineering SEKE'99, Kaiserslautern 1999.

[Stah05] Stahlknecht, P.; Hasenkamp, U.: Einführung in die Wirtschaftsinformatik. 11. Auflage, Springer Verlag, 2005.

[ThLo04] Theling, Thomas; Loos, Peter: Determinanten und Formen von Unternehmenskooperationen. Paper 18, Johannes Gutenberg-Universität Mainz, Lehrstuhl für Wirtschaftsinformatik und BWL, Mainz 2004.

[Webs03] Webster, Melissa: An End-User View of the Collaborative Software Development Market. IDC, 2003.