

Informatikprojekte und Mediation

Diplomarbeit von Andreas Duppenhaler

*Nachdiplomkurs Mediation in Wirtschaft, Umwelt und
Verwaltung*

*Fachhochschule Aargau
Nordwestschweiz*

Birsfelden, 13. November 2001

Inhaltsverzeichnis

Vorwort	3
Mein Bezug zur Informatik	3
Mein Bezug zur Mediation	3
Informatikprojekte und Mediation – eine Kombination, die sich aufdrängt	3
Die Wahl des Themas	3
Konflikte in Informatikprojekten	4
Informatikprojekte	4
Verunglückte Projekte	4
Art der Konflikte und ihre Ursachen.....	4
<i>Sachverhaltskonflikte</i>	5
<i>Interessenskonflikte</i>	6
<i>Beziehungskonflikte</i>	6
<i>Wertekonflikte</i>	7
<i>Strukturkonflikte</i>	7
Die Konfliktbeteiligten	8
Innere und äussere Konflikte.....	8
Weshalb es in Softwareprojekten Konflikte gibt	9
<i>Softwareprojekt als Betriebsstörung</i>	9
<i>Erwartungen nicht erfüllt</i>	9
<i>Einander nicht verstanden</i>	10
Mediation als Konfliktbearbeitungsmodell	11
Sind Konfliktlösungen bei Informatikprojekten gesucht?	11
Was ist anders in der Mediation bei Informatikprojekten?	11
Sind die Voraussetzungen für eine Mediation gegeben?	12
Welche Feldkompetenz braucht der Mediator?	13
Hat die Mediation bei Informatikprojekten eine Zukunft?	14
Mediation als Prävention	15
Können Konflikte in Informatikprojekten durch Mediation vermieden werden? ...	15
Welche Phasen im Softwarezyklus eignen sich für präventive Mediation?	15
Welche Rolle soll der Mediator im Softwareprojekt einnehmen?	15
Nachwort	17
Softwareentwicklung – Kommunikation – Mediation	17

Vorwort

Zur Wahl des Themas "Informatikprojekte und Mediation"

Mein Bezug zur Informatik

Während meiner Ausbildung zum Ingenieur in Fachrichtung Heizungs-, Lüftung- und Klimatechnik wurde mein Interesse an der Mathematik geweckt. Beim nachfolgenden Studium als Mathematiker-Ingenieur an der ETH Lausanne habe ich die Fachrichtung Informatik eher zufällig gewählt. Seitdem beschäftige ich mich beruflich mit Informatikprojekten.

In den letzten 15 Jahren konnte ich in diversen Softwareentwicklungsprojekten Erfahrungen sammeln. Zuerst als Projektleiter an einem Forschungsinstitut und dann als Miteigentümer in der Geschäftsleitung einer kleinen Softwareentwicklungsfirma.

Mein Bezug zur Mediation

Über die Mediation bin ich just zu einer Zeit "gestolpert", als ich den Wunsch nach einer Horizonterweiterung verspürt habe. Die Techniken, sich zurückhaltend und einfühlsam auf die Probleme anderer einzulassen und mit Phantasie zur Lösung von Konflikten beizutragen, weckten mein Interesse. Ich wollte sie erlernen und sie im privaten und beruflichen Bereich nutzbar machen.

Zu meinem Interesse für die Mediation im beruflichen Umfeld kommt die wachsende Einsicht, dass meine Stärken in der Softwareentwicklung mit zunehmendem Alter nicht auf derselben Ebene liegen, wie die der jungen Informatiker. Die Ausbildung für Mediation in Wirtschaft, Umwelt und Verwaltung schafft ideale Voraussetzungen für die Anwendung der Mediation im Bereich der Informatik.

Informatikprojekte und Mediation – eine Kombination, die sich aufdrängt

Das Management jedes Informatikprojekts hat nur ein Ziel; eine Lösung herbeizuführen. Von Problemen spricht man nicht gerne, sie sind nur da, um gelöst zu werden. Und trotzdem sind Konflikte in Informatikprojekten, insbesondere in Softwareentwicklungsprojekten, quasi vorprogrammiert.

Eine der Grundvoraussetzungen für die Mediation, dass die Konfliktbeteiligten eine gemeinsame Zukunft vor sich haben und dadurch automatisch ein grosses Interesse an der Lösung eines Konflikts haben, ist bei Informatikprojekten praktisch immer gegeben. Gleich ob es sich um einen inneren oder einen äusseren Konflikt handelt, die Bindung der Konfliktbeteiligten untereinander ist - wie in einer "Schicksalsgemeinschaft" - hoch.

Bei Softwareentwicklungsprojekten haben im allgemeinen alle Beteiligten ein hohes Interesse einvernehmliche Lösungen zu finden. Alle wissen, dass die Entwicklung einer Software nicht erstritten werden kann; auch dann nicht, wenn es um streitbare Forderungen geht.

Die Wahl des Themas

Diese Arbeit bildet eine natürliche Synthese aus meinem Bezug zur Informatik und zur Mediation.

Konflikte in Informatikprojekten

Von der Schwierigkeit einander zu verstehen

Informatikprojekte

Heute stellt die Informatik, das Fachgebiet der Informations- und Datentechnik und der Einsatz von Informatikmitteln (Computer, Datennetze, usw.) ein Gebiet des Wissens und der Allgemeinbildung dar [Ze 01]. Bei den Informatikprojekten in dieser Arbeit geht es jedoch um den praktischen Einsatz der Informatik in Industrie, Wirtschaft und Verwaltung, dem sogenannten *professionellen* Einsatz.

Ein Projekt ist die Menge aller Tätigkeiten, Interaktionen und Resultate, die mit dem Versuch zusammenhängen, ein bestimmtes Ziel mit begrenzten Mitteln und innerhalb begrenzter Zeit zu erreichen [FrüLudSan 00]. In diesem Papier geht es speziell um die Bereitstellung von Softwaresystemen resp. Softwareprodukten und dementsprechend um Softwareprojekte. Auf die Definition aller verwendeten Begriffe wird verzichtet, dies würde den Umfang der Arbeit sprengen. Die meisten Begriffsdefinitionen sind in "Informatik-Projektentwicklung" [Ze 01] zu finden.

Verunglückte Projekte

"Weit mehr als die Hälfte aller IT-Projekte verlaufen nicht reibungslos und viele davon scheitern ganz. ..." lautet der einführende Satz für ein Seminar des Bundesamts für Berufsbildung und Technologie (BBT). Mit dem Seminar "Einführung in die Formen rascher Konfliktbeilegung bei IT-Projekten" im Rahmen von Soft[net] dürfte das Bundesamt in der Schweiz ein allgemeines Bedürfnis aufgegriffen haben.

Seit dem die sog. Softwarekrise an der berühmten Garmisch-Konferenz 1968 diagnostiziert wurde [Hu 99], ist das Scheitern von Softwareprojekten thematisiert. Trotzdem sind kaum Publikationen darüber zu finden. Kein Wunder, wer möchte schon über Entwicklungsprojekte berichten, die abgebrochen werden mussten, oder über Projekte deren Nutzen in keinem Verhältnis zu den Kosten standen, und/oder deren Terminüberschreitungen ins Unermessliche stiegen?

In einem Artikel über die wichtigsten Ursachen für Misserfolge in Softwareprojekten [May 98], werden erschreckende Zahlen genannt. Allein für das Jahr 1995 haben Staat und Wirtschaft der USA 81 Milliarden US\$ für abgebrochene Softwareprojekte und zusätzlichen 59 Milliarden US\$ für überzogene Budgets in Softwareprojekten bezahlt.

Die Verluste durch verunglückte Softwareprojekte in Europa dürften eine vergleichbare Grössenordnung erreichen, da die angewandten Verfahren, Methoden und Tools bei der Softwareentwicklung dieselben sind. Die Dokumentation von Misserfolgen in Softwareprojekten wäre nicht nur aus rein organisatorischer und technischer Sicht von Interesse, sondern auch für die Analyse der daraus zwangsläufig resultierenden Konflikte.

Art der Konflikte und ihre Ursachen

Wenn wir davon ausgehen, dass es verunglückte Softwareprojekte gibt, gibt es auch Konflikte in Softwareprojekten. Mit der folgenden Definition eines Konflikts,

- *Konflikt* := eine Differenz, die unangenehme Gefühle auslöst,

ist zudem anzunehmen, dass der Konflikt in einem Softwareprojekt schon existiert, bevor ein Projekt definitiv Schiffbruch erleidet. Oder anders formuliert, wer möchte behaupten, dass unangenehme Gefühle nicht schon vor einem Schiffbruch aufkommen?

Um adäquate Interventionen bei Konflikten anzuwenden, müssen die Ursachen eines Konflikts erkannt werden. Die folgende allgemeine Typisierung von Konflikten und ihren Ursachen nach [Glasl 00] wird bezüglich Softwareprojekten kommentiert.

Sachverhaltskonflikte

sind verursacht durch:

- *Mangel an Information*
Nicht ausreichende Information führt in jedem Softwareprojekt zu Problemen. Die aktive und kontinuierliche Information der Mitarbeiter, auch über das Umfeld und die Rahmenbedingungen, und die Wahrnehmung von Stimmungsschwankungen im Projekt ... [FrüLudSan 00] gehört zu den Aufgaben eines Projektleiters.
- *Fehlinformation*
Fehlinformationen in den frühen Projektphasen (die zu Fehlern in der Software führen) können nicht nur Konflikte, sondern auch erhebliche Kosten verursachen. Die Kosten eines Fehlers in einem Softwaresystem steigen im Durchschnitt exponentiell mit der Dauer, die er unentdeckt im System ist.
- *unterschiedliche Einschätzung darüber, was wichtig ist*
Bei Softwareprojekten ist - im Gegensatz zu konventionellen Vorstellungen über Personalführung - eine Doppelprojektleitung verbreitet. Die projektinternen und projektexternen Aufgabenbereiche der Projektleitung werden auf zwei Personen aufgeteilt. Die Einschätzung darüber, was die "richtigen" Prioritäten sind, kann schon in einer konventionellen Projektorganisation zu Differenzen führen; im Falle einer Zweierprojektleitung ist die Wahrscheinlichkeit dazu noch grösser.
- *unterschiedliche Interpretationen von Informationen*
Informationen werden von Menschen in Softwareprojekten unterschiedlich wahrgenommen und folglich auch unterschiedlich interpretiert. Diesbezüglich unterscheiden sich Projektmitarbeiter in der Informatik nicht von anderen Menschen, auch wenn "den Informatikern" oft pauschal einheitliche Eigenschaften nachgesagt werden. Wesentliche Parameter der Wahrnehmung, wie fachliches Wissen, Erfahrungshintergrund, persönliche Wertvorstellung sind bei Beteiligten an einem Softwareprojekt individuell ausgeprägt.
- *unterschiedliche Vorgehensweise*
Eine unterschiedliche Vorgehensweise von Projektmitarbeitern in bestimmten Situationen ist für Softwareprojekte geradezu typisch. In Softwareprojekten wird eine Idee oft erstmalig verwirklicht, was vielfach einen Rückgriff auf Bestehendes erschwert. Seit der Softwarekrise wird versucht, die Vorgehensweise in der Softwareentwicklung zu standardisieren. Software soll - wie ein anderes technisches Produkt - ingenieurmässig entwickelt werden. Dieses Ziel wird seit über 30 Jahren verfolgt durch den Einsatz von Methoden und Werkzeugen, durch Projektorganisation und Projektmanagement, durch Ausbildung und permanente Schulung. Trotz enormer Fortschritte im Software-Engineering ist die Softwareentwicklung nicht mit industriellen Produktionsmethoden vergleichbar. Die unterschiedlichen Vorgehensweisen werden auch in Zukunft Konflikte verursachen.

Interessenskonflikte

sind verursacht durch angenommene oder tatsächliche Konkurrenz:

- *von realen (inhaltlichen) Interessen*
Die Interessen von Auftraggeber (im amerikanischen umfassender mit "Stakeholder" bezeichnet [May 98]), Anwender, Projektleitung und Projektteam sind durch ihre jeweiligen Rollen naturgemäss verschieden. Es gäbe beliebig viele spezifische Beispiele unterschiedlicher Interessen der Beteiligten in einer (Informatik-) Projektorganisation aufzuzählen. Eines davon ist die verbreitete Illusion von Auftraggebern/Anwendern, dass jeder das - von einer Software - bekommen könne, was er sich wünscht. Dies kollidiert mit vielen Interessen der Projektleitung/Entwickler im Softwareprojekt: schlanke, wartbare, erweiterbare, widerspruchsfreie, etc. Software in einem kontrollierbaren, kostengünstigen Projekt zu erstellen.
- *von Verfahrensinteressen*
Bei der Entwicklung von Software arbeiten Menschen aus verschiedenen Organisationen in einem länger dauernden Entwicklungsprozess miteinander zusammen. Im allgemeinen sind die Verfahrensinteressen durch die unterschiedlichen Kulturen in den Organisatoren geprägt und nicht einheitlich.
Ein für Softwareprojekte spezifisches Beispiel ist das Verfahren zur Erstellung des Pflichtenhefts. Ihm kommt eine spezielle Bedeutung zu, da nach der rechtlichen Situation für die Beurteilung des Projekterfolgs das Pflichtenheft heranzuziehen ist [Vorn 99]. Einerseits gibt es aus naheliegenden Gründen ein grosses Interesse, ein verbindliches Pflichtenheft möglichst noch vor Vertragsabschluss zu erstellen. Andererseits hat die Erfahrung gezeigt, dass es selbst bei den besten Absichten aller Beteiligten im Allgemeinen nicht möglich ist, ein Pflichtenheft in nur einem Anlauf abschliessend zu erstellen. Die "Mitwirkungspflicht" der Anwender beim Erstellen des Pflichtenhefts und moderne Vorgehensmodelle im Softwareengineering (Spezifikation durch Prototyping) sind zwei Beispiele von divergierenden Verfahrensinteressen bezüglich dem Interesse, möglichst früh im Projekt, eine ausführliche Beschreibung der Leistungen zu erstellen, die nötig sind, damit die Ziele erreicht werden.
- *von psychologischen Interessen*
Beteiligte an einem Softwareprojekt dürften sich diesbezüglich von anderen Menschen, die an technikorientierten Projekten arbeiten, kaum unterscheiden.

Beziehungskonflikte

sind verursacht durch:

- *starke Gefühle*
Wer denkt, dass diese Ursache in Softwareprojekten keine Rolle spielt, wird im Aufsatz "Die nicht-technische Seite des Software-Engineering" [DeMarco 97] von Tom DeMarco eines Besseren belehrt: "Methoden und Techniken schienen keine so grosse Rolle zu spielen, wie er erwartet hatte. Sie machten nicht annähernd so viel aus wie bestimmte nicht-technische Faktoren: zum Beispiel Harmonie innerhalb des Teams, geschickte Rollenverteilung und Zielorientierung in der Gruppe. Liebeskummer beeinflusste das Projekt stärker als die beste Technologie."
- *Fehlwahrnehmungen oder Stereotypen*
Der Misserfolg vieler Softwareprojekte sei darauf zurückzuführen, dass Projektleiter keine Wahrnehmung hätten, wer letztlich darüber befindet, ob ein Projekt als Erfolg oder als Misserfolg deklariert wird [May 98]. Diese Aussage ist für die

Analyse von Konflikten besonders interessant, da (Fehl-) Wahrnehmungen, unterschiedliche Einschätzungen und Interpretationen zusammenhängen.

- *mangelnde Kommunikation oder Fehlkommunikation*
Zu den wichtigsten Aufgaben des Projektmanagements zählen Frühauf et al. [FrüLudSan 00] die Kommunikation. Fehlende Kommunikationsfähigkeiten werden auch als eine Ursache für Misserfolge in - vor allem grösseren - Softwareprojekten [May 98] genannt.
- *wiederholtes negatives Verhalten*
In einem Entwicklungsteam führt wiederholtes negatives Verhalten einzelner Mitglieder über kurz oder lang zwangsläufig zu Konflikten.

Wertekonflikte

sind verursacht durch:

- *verschiedene Kriterien zur Bewertung von Ideen oder Verhalten*
- *ausschliessliche Ziele von innerem Wert*
- *unterschiedliche Lebensformen, Ideologien und Religionen*

Seit jeher gelten Programmierer eher als Künstler denn als Handwerker. Dies ist aus Sicht der Softwareindustrie, die professionelle Lösungen mit den typischen Qualitätsmerkmalen von technischen Produkten erstellen will, natürlich unerwünscht. Andererseits wird von den Softwareingenieuren – vor allem beim Design – ein hohes Mass an Kreativität verlangt. Die Kriterien zur Bewertung von unterschiedlichen Ideen können in einem Team von kreativen Mitarbeitern durchaus verschieden sein. Auch die unterschiedlichen Lebensformen der Projektmitarbeiter können - in Zeiten hoher Beanspruchung - in einem Projektteam zu Konflikten führen.

Strukturkonflikte

sind verursacht durch:

- *destruktive Verhaltens- oder Interaktionsmuster*
- *ungleiche Kontrolle, Eigentumsverhältnisse oder Verteilung von Ressourcen*
- *ungleiche Macht und Autorität*
- *geographische, physische oder umweltbezogene Faktoren, welche Zusammenarbeit behindern*
- *Zeitwänge*

Projekte sind im allgemeinen für Strukturkonflikte anfällig; Softwareprojekte bilden diesbezüglich keine Ausnahme. Auf eine Kommentierung der ersten vier Ursachenpunkte wird deshalb verzichtet.

Zeitwänge hingegen, gehören zu einem Softwareprojekt wie der Zaun zum Garten. Wenn über Softwareprojektmanagement debattiert und publiziert wird, sind Themen wie Zeit, Zeitaufwand, Zeitplan, Zeitüberschreitung, usw. als Evergreens mit dabei.

In seinem Aufsatz über "Management-unterstütztes Softwareengineering" mahnt uns Tom DeMarco [DeMarco 97], unsere wichtigste Ressource – Zeit – vernünftig zu managen und im Essay "*Warum ist Software so teuer?*" fragt er provokativ: "Woher kennen Manager den 'richtigen' Zeitplan?"

Eine der bekanntesten Aussagen zum Thema stammt von F.P. Brooks: "Adding manpower to a late software project makes it later", die sogar als *Brooks' law* bezeichnet wird.

Aus den Zitaten lässt sich erahnen, wie viel Sprengkraft das Thema beinhaltet. Als Konfliktursache werden Zeitzwänge in Softwareprojekten auch weiterhin Dauerbrenner bleiben.

Die Konfliktbeteiligten

Die potentiellen Konfliktbeteiligten bei Softwareprojekten sind in den vorherigen Ausführungen schon genannt worden:

- *Auftraggeber (Stakeholder)*
Der Auftraggeber ist das einem Projekt übergeordnete Organ; es kann Betriebsintern oder extern sein. Der Auftraggeber will und zahlt das Projekt, das den künftigen Anwendern nützen soll. Er erteilt auch den Projektauftrag.
In einem Konflikt kann sich überraschend herausstellen, dass "der wahre Auftraggeber" ein anderer ist, als der angenommene. Nicht unwesentlich ist zudem, welche Personen "den Auftraggeber" vertreten.
- *Projektleitung*
Diese besteht aus einer oder mehreren Personen, die ein Projekt führen und dem Auftraggeber verantwortlich sind. Eine Aufteilung in Benutzer-Projektleitung und Informatik-Projektleitung tritt häufig auf.
- *Projektteam*
Das sind jene Personen, welche zur Durchführung eines Projekts benötigt werden; Informatiker und/oder Organisatoren, sowie Mitarbeiter aus dem Anwendungsbereich.
- *Benutzer oder Anwender*
Die Benutzer oder Anwender sind jene Personen, welche die zukünftige Anwendung (Software) für ihre Arbeit oder persönlichen ihre Bedürfnisse einsetzen und nutzen.

Auftraggeber, Projektleitung, Projektteam und Anwender können verschiedenen Organisationen angehören, verschiedene Rollen inne haben, sowie in unterschiedlichen Abhängigkeiten und Beziehungen zueinander stehen. Dies muss für jeden Konflikt individuell analysiert werden.

Innere und äussere Konflikte

Wenn wir uns ein Softwareprojekt als System vorstellen und die Akteure danach unterscheiden, ob sie zu diesem System gehören (also innerhalb der Systemgrenze agieren) oder zur Systemumwelt (also ausserhalb der Systemgrenze agieren), können wir *innere* und *äussere* Konflikte unterscheiden. Dabei ist zu beachten, dass ein solches System aufgrund der begrenzten Projektdauer eine zeitlich begrenzte Lebensdauer hat.

- *innerer Konflikt*
Wenn die Konfliktbeteiligten ausschliesslich zum Softwareprojekt gehören, bezeichnen wir den Konflikt als "innerer Konflikt".
- *äusserer Konflikt*
Ein "äusserer Konflikt" liegt vor, wenn die Differenzen zwischen Beteiligten im Softwareprojekt und Akteuren ausserhalb des Projekts auftreten.

Die Unterscheidung in innere und äussere Konflikte ist unabhängig von der Konfliktart (Sachverhaltskonflikt, Interessenskonflikt, usw.). Bei der Suche nach einem angemessenen Konfliktmanagement kann diese Unterscheidung hilfreich sein.

Weshalb es in Softwareprojekten Konflikte gibt

Unabhängig von der obigen Konflikttypologie und den bereits genannten Ursachen von Konflikten werden im folgenden noch weitere typische Gründe für Konflikte in Softwareprojekten beschrieben.

Softwareprojekt als Betriebsstörung

Die Ausgangs- und Motivationslage der verschiedenen Betroffenen eines Softwareprojekts ist unterschiedlich. Der Auftraggeber, der eine neue Informatiklösung beauftragt, sieht die Dinge anders als die künftigen Anwender.

Die Interessen der Anwender sind unter anderen: reibungslose Arbeitsabläufe, klare Arbeitsverhältnisse, Sicherheit, Stabilität und Komfort. Sie sind normalerweise auf Dauer in ihrer Funktion tätig und haben gelernt, mit einer bestehenden Lösung zu "leben". Ein Softwareprojekt stört ihren gewohnten Arbeitsalltag, ähnlich wie Umbauarbeiten in einem Haus, in dem man wohnt. Die Befürchtungen, dass durch ein Softwareprojekt Mehrbelastungen entstehen und Einführungsspannen auftreten, sind nicht unbegründet.

Das primäre Interesse der Projektmitarbeiter ist, das Resultat des Projekts unter den gegebenen Rahmenbedingungen zu erreichen. Der Einsatz der Softwareentwickler in einem Projekt ist zeitlich befristet. In der Endphase eines Projekts werden die Entwickler hektisch, schlafen wenig und die Anspannung nimmt zu. Wenn der Fertigstellungstermin näher rückt, helfen die Projektmitarbeiter einander vermehrt; es gilt das Ziel zu erreichen und das Projekt abzuschliessen.

Die Softwareentwickler sehen ihr Tun – Entwerfen, Schreiben, Testen, Dokumentieren und Debuggen von Software – als kreativen Akt, während aus der Haltung der Anwender klar wird, dass der Betrieb schon vor der "Betriebsstörung", Softwareprojekt, funktioniert hat und ihr Interesse dem reibungslosen Betrieb nach dem Projekt gilt.

Erwartungen nicht erfüllt

"Wie passt es zusammen, dass die Softwareindustrie als Ganzes mit ihren Produkten die kühnsten Erwartungen übertrifft, während jedes Softwareprojekt für sich weit hinter den Projektzielen zurückzubleiben scheint?" Dies fragt Tom DeMarco [DeMarco 97] und gibt auch gleich eine Antwort: "Die Projekte bleiben nicht hinter den Erwartungen zurück, und das wissen die Kunden auch ganz genau. (...) Die Leute klagen über uns [Softwareentwickler], weil sie wissen, dass wir dann härter arbeiten. Wir haben ihnen beigebracht, sich so zu verhalten. Bis heute reagieren wir auf Beschwerden mit vermehrten Anstrengungen".

Was uns hier ein erfahrener Softwareingenieur der ersten Stunde aufischt, mag Aussenstehenden absurd vorkommen und dem Bild von gutverdienenden, unverzichtbaren Programmierern widersprechen. Aus meiner persönlichen Erfahrung kann ich jedoch bestätigen, dass wir in Folge des Drucks von Kunden immer härter arbeiten.

Der erste Teil der Antwort von DeMarco ist leider ein Wunschdenken, denn viele Softwareprojekte erfüllen die Erwartungen heute nicht. Wichtig dabei ist, dass Erwartungen und Anforderungen nicht gleichgesetzt werden. Die Anforderungsspezifikation ist für ein Softwareprojekt von grosser Bedeutung; sie sollte in einer geordneten Darstellung - eventuell halbformal - vorliegen, sowie auf Vollständigkeit und Widerspruchsfreiheit geprüft sein. Dies unterscheidet Anforderungen grundsätzlich von (stillen) Erwartungen, die Auftraggeber und Anwender an Softwarelösungen

haben. Stille Erwartungen werden nur selten zufällig erfüllt und Softwareingenieure können nicht formulierte Bedürfnisse nicht erahnen.

Bei Auftraggebern und Anwendern herrscht grosse Unsicherheit beim Evaluieren/Auswählen eines Softwareprodukts, resp. beim Formulieren von Anforderungen an ein Softwaresystem. Im allgemeinen wird dann ein Berater engagiert, um den richtigen Partner für die Lösung in einem Ausschreibungsverfahren zu suchen. Mit dem Auftreten eines neuen Akteurs kommen auch neue, eigene Interessen dazu, die das Konfliktrisiko nicht mindern.

Einander nicht verstanden

Warum haben sie nicht gesagt, dass ...? Was meinen sie mit ...? Das sind oft gehörte Fragen, die in Softwareprojekten nicht selten auch noch bei der Einführung einer Lösung - am Ende eines Projekts - gestellt werden, also dann, wenn die Missverständnisse schon "programmiert" worden sind.

Die ingenieurmässige Bewältigung der Probleme als Reaktion auf die Softwarekrise hat die Verständnisprobleme zwischen Anwendern und Entwicklern nicht beseitigt. Die "neue" Empfehlung, "Einbezug der Anwender in das Projektteam zur Vermeidung eines möglichen Grabens zwischen Projektentwicklern und Anwendern ..." in [Ze 01], ist nur eine von zahlreichen – weniger dramatisch anmutenden – Anstrengungen, die Zusammenarbeit zwischen Entwicklern und Anwendern zu institutionalisieren. Die Partizipation von Anwendern wird mittels Prototypen, Simulationen, Kommunikationsmethoden, usw. gefördert.

Nicht nur die Software sondern auch die Prozesse bei ihrer Entwicklung werden zunehmend abstrakter und komplexer, so dass Missverständnisse vorprogrammiert zu sein scheinen.

Mediation als Konfliktbearbeitungsmodell

Der Konflikt ist da - eine einvernehmliche Konfliktlösung wird gesucht

Sind Konfliktlösungen bei Informatikprojekten gesucht?

Wer diese Frage an dieser Stelle in diesem Papier noch nicht eindeutig mit "Ja" beantworten kann, sei an folgende Schlagzeilen erinnert:

- *Bahnhof SBB: Totaler Systemausfall* (erschieden in der Basler Zeitung am 15.6.01)
Knapp zwei Stunden lang lief gestern im Bahnhof Basel SBB gar nichts mehr: Weil ein Computersystem im Stellwerk vollständig ausfiel, kam es zwischen 9.32 und 11.15 Uhr zu einem Stillstand. Die Züge konnten weder ein- noch ausfahren, sie wurden in den Bahnhöfen Muttenz, Pratteln und Münchenstein gewendet. Mehrere tausend Passagiere und 50 Züge waren von der Panne betroffen. Im Extremfall mussten Verspätungen von bis zu zwei Stunden hingenommen werden.
- *Vontobel begräbt Internet-Bank* (erschieden in der Basler Zeitung am 28.2.01)
Aus Kostengründen hat die Vontobel-Gruppe ihr Internet-Bank-Projekt y-o-u abgeblasen. (...) Das Ende des Projekts kam für die an der Realisierung Beteiligten recht überraschend. Doch der Verwaltungsrat hatte am Montag entschieden, das Ganze abzublasen. Gestern wurden die 40 Mitarbeiter und die Medien informiert. Die Begründung für den Stopp der Internet-Bank, mit deren Aufbau die Unternehmensberatung PricewaterhouseCoopers beauftragt war: «Kostenrahmen und Zeithorizont haben sich als nicht realisierbar erwiesen.» Mit 151 Mio. Fr. hat das ambitionöse Projekt die Konzernrechnung 2000 belastet.
- *Mit neuen Kräften gegen das Paket-Chaos* (erschieden im Tages-Anzeiger am 22.9.99)
Der 40-jährige Informatiker Michel Kunz soll das Durcheinander bei der Paketpost entwirren. (...) Kunz kenne die Probleme der Paketpost bestens, die ja in erster Linie Softwareschwierigkeiten seien. Und mit dem Informatikbereich habe er eine Querschnittaufgabe inne gehabt. Er könne daher über den Tellerrand hinausblicken. Das sture Bereichsdenken wird als Postkrankheit und als eine Ursache für das Päckelchaos gesehen.

Die Liste könnte mit weiteren, prominenten Beispielen und vielen anonymen, unbekanntem ergänzt werden. Erst kürzlich wurde bekannt, dass der Lieferant einer Softwarelösung, um deren Realisierung wir uns (mit der eigenen Softwarefirma) auch beworben haben, mit dem Auftraggeber in einem Gerichtsverfahren steht.

Statt zu Fragen: "Sind Konfliktlösungen bei Informatikprojekten gesucht?" , müssen wir fragen: "Ist Mediation als Konfliktlösungsmethode bei Informatikprojekten bekannt und erwünscht?"

Was ist anders in der Mediation bei Informatikprojekten?

Man kann sich fragen, was an der Mediation bei Informatikprojekten speziell ist. Ich denke, dass die Mediation in diesem Umfeld als Wirtschaftsmediation verstanden werden muss. Die beschriebenen Konflikte decken eine breite Palette ab, wie sie in der Wirtschaftsmediation im allgemeinen bekannt sind: Konflikte mit Kunden, Lieferanten und Verbrauchern, Konflikte zwischen Mitarbeitern, Teams und Organisationseinheiten, Fragen der Haftung und Gewährleistung, usw.

Die Mediation bei Informatikprojekten unterscheidet sich nicht grundsätzlich von der Mediation in anderen Wirtschaftsbereichen. Sie ist auch in diesem Anwendungsbereich eine moderne, kooperative Konfliktlösungsmethode, bei der eine neutrale dritte Person, der Mediator, die Parteien bei der Lösung eines Konflikts unterstützt. Das Mediationsverfahren ist eine strukturierte Vorgehensweise unter Führung des Mediators, bei dem die Parteien aktiv und eigenverantwortlich an einer

interessengerechten Lösung ihres Konflikts wirken. Die Entscheidungsgewalt über Fortführung und Ausgang des Mediationsverfahrens bleibt dabei jederzeit bei den Konfliktparteien.

Beim Erstellen des Mediationsplans und bei der Wahl des Ablaufs des Mediationsverfahrens kann - wie in anderen Konflikten im Wirtschaftsleben - vorgegangen werden. Es müssen Fragen geklärt werden: Handelt es sich um einen inneren oder um einen äusseren Konflikt? Besteht der Konflikt zwischen Organisationen, oder zwischen Mitarbeitern? Sind grössere Gruppen in den Konflikt involviert? Man wird das Handwerkszeug [FHA 01] einsetzen und die persönlichen Erfahrungen einbringen, um in jedem neuen Fall ein massgeschneidertes Vorgehen zu konzipieren.

Einige Spezialitäten weist die Mediation in diesem Anwendungsbereich aber trotzdem auf. Wenn man die Konflikte in Informatikprojekten analysiert, fällt auf, dass "der Welt der Informatik" im heutigen Wirtschaftsleben noch immer etwas "Exotisches" anhaftet. Der Hintergrund der Konflikte und das Umfeld der Konfliktparteien passen nicht in die Norm der Wirtschaft im allgemeinen.

Sind die Voraussetzungen für eine Mediation gegeben?

Jedes Unternehmen weiss, wie schnell Probleme mit Informatikprojekten zu einem Rechtsstreit eskalieren können. Der Gang zum Gericht bedeutet erheblichen zeitlichen sowie finanziellen Aufwand und führt dennoch in vielen Fällen nicht zu den erhofften Ergebnissen. Die Mediation bietet den Konfliktparteien die Chance, selbst eine einvernehmliche Lösung ihrer Probleme zu erarbeiten.

Gewisse Voraussetzungen müssen gegeben sein, damit Mediation sinnvoll ist. Nachfolgend werden einige davon kommentiert, die für die Wirtschaftsmediation und im Zusammenhang mit Informatikprojekten von Bedeutung sind.

- *Die Konfliktparteien haben ein Interesse an guten zukünftigen Beziehungen:*
In den meisten äusseren Konflikten ist diese wichtige Voraussetzung erfüllt. In inneren Konflikten hängt es davon ab, wie lange das Projekt noch dauert und wie die Mitarbeiter in zukünftigen Projekten eingesetzt werden.
- *es geht nicht allein um eine Rechtsfrage:*
Ansonsten müsste der Rechtsweg bevorzugt werden.
- *Es geht um eine Vielzahl von Konfliktpunkten:*
Gemeint ist, dass der Konflikt nicht - oder nur schlecht - in direkten Gesprächen gelöst werden kann.
- *Die wichtigsten Konfliktparteien sind vertreten:*
Der Mediator sollte mit Nachdruck darauf hinwirken.
- *Alle Beteiligten streben eine einvernehmliche Lösung an:*
Diese Voraussetzung ist vor allem bei inneren (Mitarbeiter-) Konflikten nicht immer erfüllt.
- *Die Konfliktparteien haben Verhandlungsgeschick:*
Da bei einem Informatikprojekt viel verhandelt werden muss, ist die Wahrscheinlichkeit hoch, dass diese Voraussetzung erfüllt ist.
- *Es stehen ausreichende Ressourcen zur Verfügung:*
Für die Informatikprojekte ist die Klärung der zeitlichen Verfügbarkeit besonders wichtig. Natürlich muss auch die zeitliche Belastung von allfälligen Alternativen geprüft werden.
- *Die Konfliktparteien akzeptieren die Vertraulichkeit:*
Dies ist im wirtschaftlichen Umfeld eine wichtige Voraussetzung.

- *Die Konfliktparteien nehmen freiwillig an der Mediation teil:*
In Organisationen kann eine Mediation auch verordnet werden. Der Mediator muss dann einschätzen können, ob der sanfte Zwang das Verfahren verunmöglicht.

In den meisten äusseren Konflikten haben die Konfliktparteien bei Informatikprojekten zwangsläufig ein grosses Interesse an guten zukünftigen Beziehungen. Dadurch ist die Chance für eine Mediation gross. Die übrigen Voraussetzungen sind sicher nicht schlechter als in anderen Bereichen der Wirtschaft.

Welche Feldkompetenz braucht der Mediator?

Unter Feldkompetenz eines Mediators wird seine Kompetenz im Anwendungsgebiet der Mediation, in diesem Falle auf dem Gebiet der Informatikprojekte, verstanden. Durch die Verwendung dieses Begriffs wird der Unterschied zwischen der eigentlichen Fachkompetenz in der Mediation und der Kompetenz im Anwendungsgebiet deutlich gemacht.

Die Aufgabe des Mediators ist es, die Parteien bei der Konfliktbearbeitung und der Problemlösung zu führen und zu unterstützen. Um diese Aufgabe professionell zu erfüllen, stehen ihm eine Vielzahl von Methoden zur Verfügung. Seine Fachkompetenz bezieht sich auf das Mediationsverfahren, für das er verantwortlich ist. Der Mediator steuert die Verhandlungen nicht inhaltlich sondern nur verfahrensmässig. Es liegt an den Parteien selbst, eine ihren Interessen optimal entsprechende Problemlösung zu erarbeiten.

Eine verbreitete Lehrmeinung in der Mediation besagt, dass ein Mediator "nur" für den Mediationsprozess - nicht für die Lösung - verantwortlich ist. Die Frage: "welche Feldkompetenz braucht der Mediator", wäre damit beantwortet, nämlich keine.

Da mich die Frage schon seit längerem beschäftigt, habe ich sie schon verschiedenen Mediatoren gestellt. Die Antworten weisen eine eindeutige Tendenz auf. Mediatoren mit einem grossen Selbstbewusstsein und/oder einem anerkannten Renommé wagen sich auch ohne Feldkompetenz an Mediationsfälle. Anfänger brauchen im Allgemeinen die Sicherheit der Feldkompetenz. Dieses einleuchtende Resultat zeigt aber nur die eine Seite der Medaille, die Sicht von innen.

Würde man die Konfliktparteien fragen, ob sie einen Mediator mit Feldkompetenz bevorzugen, würde die Antwort sicher ebenso eindeutig ausfallen. Vorausgesetzt wird natürlich, dass die Fachkompetenz der wählbaren Mediatoren äquivalent ist. Ein Mediator mit Feldkompetenz wird von den Beteiligten besser akzeptiert und wird es leichter haben, ihr Vertrauen zu gewinnen.

Wie verhält es sich nun im Speziellen mit der Feldkompetenz für die Mediation bei Informatikprojekten? Könnte es sein, dass den Mediatoren ähnliches passiert, wie den Informatikern mit den Anwendern?

Die Informatik hat lange darum kämpfen müssen, als eigene Ingenieurdisziplin und Wissenschaft anerkannt zu werden. Auf der einen Seite waren die Anwender (insbesondere wenn sie selbst Ingenieure waren) der Meinung, dass das Wissen im Anwendungsgebiet das Wesentliche sei und jeder programmieren (lernen) könne. Andererseits waren die Informatiker der Überzeugung, dass die Kunst der Softwareentwicklung über die Niederungen der Anwendung erhaben sei. Äusserungen wie, "ich verstehe zwar nichts von dem, was ihr tut, aber programmieren kann ich trotzdem, was ihr wollt", waren nicht selten zu hören. Der alte Spruch "für den Ingeniör ist nichts zu schwör" wurde in beiden Richtungen appliziert.

Solche Auswüchse gehören – hoffentlich definitiv – der Vergangenheit an, und ich unterstelle auch nicht, dass Mediatoren nur annähernd so respektlos mit ihren Klienten umgehen. Trotzdem, bei der Arbeit mit Softwareprojekten kann ein Mangel an Feldkompetenz unnötige Probleme bereiten. Mögliche Haltungen von Mediatoren ohne Feldkompetenz:

- *Ich habe von Informatik keine Ahnung und lasse mir alles erklären:*
Die Fachleute empfinden die Erklärungen als unnötige Zeitverschwendung auf ihre Kosten.
- *Ich verwende sämtliche Fachwörter der Informatik, die ich kenne:*
Die Fachleute empfinden dies als anbieten mit aufgesetzter, falsch verstandener, Fachkompetenz.
- *Ich verstehe nicht viel von Informatik, zeige aber trotzdem was ich drauf habe:*
Das Kokettieren mit Understatements wird nicht immer geschätzt.

Diese Haltungen sind der Akzeptanz des Mediators abträglich. Falls beim Mediator keine Feldkompetenz in Informatikprojekten vorhanden ist, sollte mit dem Nichtwissen/Wissen ehrlich umgegangen werden, damit alle Beteiligten die Situation einschätzen können.

Ich bin überzeugt, dass bei einem Mediator mit Feldkompetenz in Informatikprojekten die Kernkompetenz zur Steuerung des Mediationsprozesses besser zum Tragen kommt als ohne. Die "richtigen" Fragen zu stellen, ohne die Hintergründe nur annähernd zu verstehen, ist auf jeden Fall nicht einfach.

Hat die Mediation bei Informatikprojekten eine Zukunft?

Als Alternative zu gerichtlichen Auseinandersetzungen in äusseren Konflikten werden Mediationstechniken im Umfeld der Informatik in Zukunft sicher vermehrt angewendet werden. Allein schon das Interesse der Konfliktbeteiligten an nachhaltigen Lösungen spricht dafür. Ob die Methoden bei Informatikprojekten unter einem anderen Namen - z.B. Konfliktmanagement - zur Anwendung kommen werden, ist nebensächlich.

Die Schweizerische Kammer für Wirtschaftsmediation hat vom Bundesamt für Berufsbildung und Technologie (BGT) im Rahmen des Soft[net] Projekts den Auftrag erhalten, eine IT-Schlichtungsstelle aufzubauen. In diesem Zusammenhang wurde im November 2001 ein erster Kurs, "Mediation bei IT-Projekten; Konflikte schnell, effizient und für die Beteiligten gewinnbringend lösen", angeboten.

Die Frage, wer – welche Berufsgruppe – zukünftig als Mediator in der Informatik auftreten wird, ist noch offen. Es ist nicht auszuschliessen, dass der Mediation etwas ähnliches passiert, wie der Informatik (siehe oben). Wenn die Überzeugung vorherrscht, dass zur Konfliktlösung das Fachwissen der Informatik im Vordergrund steht und man sich die Mediationstechniken nebenbei aneignen kann, wäre die Frage – mindestens vorläufig – beantwortet.

Mediation als Prävention

Kick-off Meeting – ein Projekt beginnt; der Grundstein für neue Konflikte ist gelegt.

Können Konflikte in Informatikprojekten durch Mediation vermieden werden?

Bevor wir überlegen ob Konflikte in Informatikprojekten durch Mediation vermieden werden könnten, stellt sich die Frage: "Können Konflikte überhaupt vermieden werden?" Wenn wir nochmals die einfache Konfliktdefinition – eine Differenz, die unangenehme Gefühle auslöst – benutzen, würde es bedeuten, dass Differenzen vermieden werden müssten oder/und die unangenehmen Gefühle, die damit verbunden sind.

In jedem Informatikprojekt gibt es Differenzen und das wird – solange Menschen involviert sind – auch immer so bleiben. Konflikte vermeiden hiesse also, die damit zusammenhängenden unangenehmen Gefühle vermeiden. Auch dies scheint mir ein unrealistisches Ziel.

Ein erreichbares Ziel wäre hingegen, Konflikte zu erkennen und wünschbar wäre, sie möglichst früh zu erkennen. Dies bedeutet konkret, die Differenzen oder/und die unangenehmen Gefühle zu erkennen. Kann dabei Mediation helfen?

Viele bekannte Persönlichkeiten aus der Informatikprojektwelt haben schon postuliert, dass Konflikte nicht ignoriert werden dürfen. Es wird kaum jemand das anvisierte Ziel in Frage stellen. Die Methode – Mediation – wird hingegen in dieser Szene sicher nicht vorbehaltlos akzeptiert. Anders als in einem sozialen Umfeld, wie z.B. in der Schule, wo die Mediation bekannt ist und auch schon ihren festen Platz hat, wird die Institutionalisierung der Mediation in Informatikprojekten schwieriger.

Welche Phasen im Softwarezyklus eignen sich für präventive Mediation?

Die Diagnosen von Praktikern und Beratern über erkannte Ursachen für Misserfolge in Softwareprojekten [May 98] zeigen, dass die Probleme über Jahre hinweg dokumentiert werden und immer wieder plötzlich auftauchen. Die meisten erwähnten Fehlerursachen haben ihren Ursprung noch bevor die erste Zeile Programmcode geschrieben wird.

Im gleichen Artikel wird über die "Stakeholder" und ihre Konflikte in Softwareprojekten berichtet: "Sie übertünchten ihre Differenzen anstatt die Konflikte in den frühen Phasen zu lösen. Die Differenzen wurden von den Entwicklern entdeckt, weil Programmierer keine zweideutigen Systeme erstellen können."

In den Zitaten steckt eine klare Antwort auf die Frage, welche Phasen im Softwarezyklus für Konfliktprävention geeignet sind. Es sind die frühen Phasen, wo Ziele, Anforderungen, Spezifikationen, Vorgehen, Zeitpläne usw. verhandelt werden.

Welche Rolle soll der Mediator im Softwareprojekt einnehmen?

Braucht es das wirklich? Das haben wir (guten Projektleiter) doch schon immer gemacht! Das ist doch nur ein neues Modewort für altbekanntes! Wir brauchen nicht noch mehr Berater, Funktionen und Rollen in einem Softwareprojekt! Die Liste skeptischer Äusserungen gegenüber der Rolle des Mediators in einem Softwareprojekt liesse sich beliebig erweitern.

Ich habe mich gefragt, ob die Aufgaben eines Mediators und die typischen Anforderungen an ihn heute von Personen mit etablierten Funktionen in Projekten wahrgenommen und abgedeckt werden. Man kann die Frage noch weiter eingrenzen und voraussetzen, dass die möglichen Personen das Handwerkszeug der Mediation beherrschen und dass es nur um die Klärung der Rolle (nach [Be 01]) geht.

- Die Person des Mediators muss von allen Beteiligten akzeptiert und respektiert werden. Es muss sich um eine Person handeln, die das Vertrauen der Parteien genießt oder sich dieses erwirbt und deren Kompetenz nicht bestritten wird. Diese Anforderung würde (hoffentlich) mindestens der Projektleiter erfüllen.
- Der Mediator soll kein eigenes Interesse an einem bestimmten Konfliktausgang haben. Er soll in diesem Sinne neutral und unparteiisch sein. Er setzt sich aber für die Interessen und Belange aller Parteien ein. In diesem Sinne ist er "allparteilich".
Diese Anforderungen und Aufgaben werden von keiner mir bekannten Funktion in einem Softwareprojekt erfüllt und abgedeckt. Projektleiter, Projektmitarbeiter, Anwender, Auftraggeber, Experten, etc. haben alle ein eigenes, spezifisches Interesse, das durch ihre Rolle gegeben ist.
- Mediatoren sind für den Gang des Gesprächs verantwortlich, die Kontrahenten für den Inhalt. Die Lösungen werden nicht von den Mediatoren, sondern von den Betroffenen erarbeitet.
Eine solche Rollenteilung ist in den klassischen Rollen eines Informatikprojekts nicht vorgesehen.

Diese nicht abschliessende Gegenüberstellung von Mediator mit bekannten Funktionen in Informatikprojekten zeigt, dass die Rolle des Mediators von einer unabhängigen Person wahrgenommen werden muss.

Der Mediator im Projekt wird den Beteiligten helfen, sich über ihre Gefühle und Interessen klar zu werden und sie verständlich zum Ausdruck zu bringen. Er wird damit eine mediative Haltung der Beteiligten fördern und die Früherkennung von Konflikten ermöglichen.

Nachwort

Die Bearbeitung des Themas hat Spass gemacht.

Softwareentwicklung – Kommunikation – Mediation

Bei der Preisverleihung für "langjährige Beiträge auf dem Gebiet der Informatik" hat Tom DeMarco [DeMarco 97] in einer Skizze seines Lebenslaufs u.a. folgendes geschrieben:

"Ich habe fast von Anfang an erkannt, dass wir Softwareleute nicht eigentlich im Computergeschäft tätig sind. Wir behaupten das zwar, und wir verhalten uns ganz sicher wie Mitglieder jeder anderen Hightechgemeinde, aber Softwareentwicklung ist kein Hightechgeschäft. Bei der Softwareentwicklung geht es vor allem darum, miteinander zu reden und Dinge niederzuschreiben. Diejenigen, die die Disziplin am stärksten voranbrachten, waren in der Regel die besten Kommunikatoren, nicht die besten Techniker ihres Fachgebiets."

Die Gedanken von DeMarco gefallen mir. Mit der Kommunikation baut er die Pfeiler für eine solide Brücke zur Mediation. Für die beiden so unterschiedlichen Disziplinen – Softwareentwicklung und Mediation – ist die Kommunikation ein wichtiges Instrument. Ich hoffe, dass wir von Seite der Mediation an unserem Teil der Brücke arbeiten. Es wird sich lohnen.

Ich habe in diesem Papier die männliche Form verwendet. Dies sollte der Lesbarkeit dienen. Selbstverständlich sind damit auch alle Frauen angesprochen.

Birsfelden, 13. November 2001

Literaturverzeichnis

[Be 01]

Besemer, Christoph.: Mediation: Vermittlung in Konflikten. Stiftung gewaltfreies Leben, Königfeld; und Werkstatt für gewaltfreie Aktion, Baden; 8. Auflage April 2001

[DeMarco 97]

DeMarco, Tom: Warum ist Software so teuer? ... und andere Rätsel des Informationszeitalters. Carl Hanser Verlag München Wien 1997
Titel der englischen Originalausgabe: "Why Does Software Cost So Much? And Other Puzzles of the Information Age". 1995 by Tom DeMarco

[Dulabaum 00]

Dulabaum, Nina L.: Mediation: Das ABC - Die Kunst, in Konflikten erfolgreich zu vermitteln. Beltz Verlag, Weinheim und Basel; 2. vollständig überarbeitete Auflage 2000

[FHA 01]

Nachdiplomkurs Mediation in Wirtschaft, Umwelt und Verwaltung; Fachhochschule Aargau für Technik, Wirtschaft und Gestaltung; Kursunterlagen aus Kurs 4, 2000 - 2001

[FrüLudSan 00]

Frühauf Karol, Ludewig Jochen, Sandmayr Helmut: Software-Projektmanagement und Qualitätssicherung. B.G. Teubner, Stuttgart, und Verlag der Fachvereine, Zürich; 3. Auflage 2000

[Glasl 00]

Glasl Friedrich: Selbsthilfe in Konflikten. Verlag Paul Haupt, Bern; 2. Auflage 2000

[Hu 95]

Huber Andreas: Ist das Software Engineering gescheitert? Neue Zürcher Zeitung, 15. August 1995

[Hu 99]

Huber Andreas: Wie arbeiten Entwicklung und Anwendung bei der Gestaltung von Informationssystemen zusammen?; Institut für Informatik, Management Weiterbildung der Universität Zürich, 23. Februar 1999

[May 98]

May, Lorin J.: Major Causes of Software Project Failures, CROSSTALK The Journal of Defense Software Engineering; July 1998
CROSSTALK On-Line <http://www.stsc.hill.af.mil/Crosstalk/crosstalk.html>

[Pa 94]

Pasch, Jürgen: Software-Entwicklung im Team - Mehr Qualität durch das dialogische Prinzip bei der Projektarbeit. 1994: Springer Verlag Berlin

[Vorn 99]

Vorndamme, J.: Die Auswirkungen rechtlicher Verpflichtungen auf die Softwareentwicklung. 1999: Oldenburger Hochschulreihe.

[Ze 01]

Zehnder, Carl August: Informatik-Projektentwicklung. vdf Hochschulverlag, Zürich; 3. vollständig überarbeitete Auflage 2001