

MEMORANDUM

Der Software-Architekt – eine neue Form der Arbeitsteilung in der IT-Welt?

Sylter Runde, 23./24. September 2004

Fragestellung

Reifere Produktionsorganisationen zeichnen sich durch Arbeitsteilung und den damit einhergehenden Spezialisierungsgewinnen aus, wobei insbesondere die Gestalt der Schnittstellen zwischen den Komponenten sowie des übergreifenden Vorgehensmodells Resultat vielfältiger Erfahrung ist. In der Bauindustrie heißen die Komponenten Gewerke, deren Zusammenspiel von der Planung am Reißbrett bis zum Controlling der Ausführung auf der Baustelle der Verantwortung von Architekten unterliegt.

Die inzwischen mehr als 40 Jahre alte Software-Industrie hat bei ihren immateriellen, aber ebenfalls hoch komplexen Aufgabenstellungen ähnliche Anforderungen zu bewältigen. Halten aber auch die übrigen Analogien, so dass es Sinn macht, die Rolle eines Software-Architekten nach diesem Muster zu spezifizieren? Wenn ja, sollte deutlich werden, welche Randbedingungen für diese Rolle bestimmend sind, welche beruflichen Voraussetzungen man mitbringen muss und welche Erfolgsfaktoren als treibende Kraft wirken. Darüber hinaus ist zu diskutieren, wo Besonderheiten der Software-Industrie die Analogiebildung etwa zur Bauindustrie behindern und mit welchen speziellen Maßnahmen man die hieraus entstehenden Probleme überwinden kann. Dabei scheint klar, dass nicht jeder Software-Ingenieur zum Software-Architekten befördert werden sollte, während umgekehrt jeder Software-Architekt ein guter Software-Ingenieur sein muss.

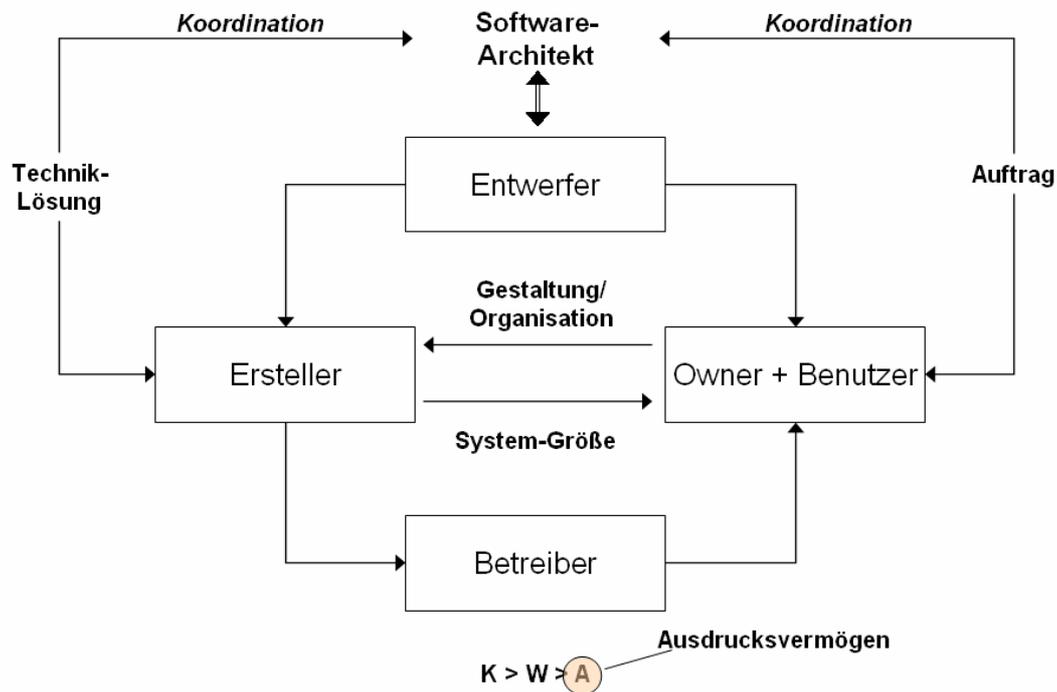
Inwieweit verfolgen die Hochschulen wie auch die Unternehmen im Falle von In-House-Fortbildungen in ihren Curricula dieses Leitbild? Unsere These ist, dass gegenüber den Bedürfnissen des Marktes heute schon und weiterhin zunehmend eine substanzielle Angebotslücke existiert und dass wir von den klassischen Architekturstudiengängen noch viel für die Gestaltung unserer Software-Engineering-Ausbildung lernen können – auch etwa für das "On the job Training" und gezielte Fortbildungsmaßnahmen.

Zusammenfassung

Die Softwareentwicklung folgt zwei grundverschiedenen Pfaden. Sie ist einmal traditionell auf individuelle Maßanfertigung für dedizierte Anwender ausgerichtet und produziert andererseits im besten industriellen Stil Massenprodukte für einen im Prinzip anonymen Markt. Im letzteren Falle dauert die professionelle Entwicklung und Einführung grundsätzlich neuer Produkte in der Regel fünf Jahre. Bei der Maßanfertigung von Software-Systemen wird die Entwicklungsdauer regelmäßig unterschätzt. Trotz relativ langer faktischer Entwicklungszeiten werden die *Benutzer* bzw. *Owner* dieser Produkte immer wieder mit unerwarteten und eigentlich nicht zu vertretenden System-Flops und fehlgeschlagenen Projekten, insbesondere bei bereichsübergreifenden Aufgabenstellungen, konfrontiert. Liegt dies daran, dass sich die Software-Branche noch immer in einem frühen Evolutionsstadium befindet? Wo in anderen konstruktiven Branchen in frühen Stadien auch Bauteile handgefertigt wurden, gibt es heute Normteile, die zu einer Erleichterung bei der Umsetzung von Konstruktionsplänen (z.B. in der Immobilienbranche, im Flugzeugbau, Automobilbau etc.) und schlussendlich zu einer Steigerung der Zuverlässigkeit der Endprodukte geführt haben (Gewährleistungsarchitekturen). Die zentrale Frage aber ist, ob eine derartige Analogiebildung denn überhaupt hilfreich sein kann. Um jedoch die Probleme bei der Erstellung von Softwareprodukten professionell und industriell in den Griff zu bekommen, sollte jeder Versuch unternommen werden, der Aussicht auf Erfolg verspricht. Praktische Erfahrungen mit Software-Architekten im professionell arbeitsteiligen Einsatz weisen einen möglicherweise erfolgsträchtigen Weg dieser neuen Form der Arbeitsteilung in der IT-Industrie.

Im Rahmen des zweitägigen Gesprächskreises wurde entlang des Software-Entwicklungsprozesses diskutiert (siehe Abbildung), welche Rollen uns heute vorliegen und welche Aufgaben sowie Anforderungen zukünftig für einen Software-Architekten im Netz der Produktentwicklung zu definieren sind, um den kreativen Koordinations- und Kommunikationsprozess bei der Erarbeitung einer technischen Lösung im Rahmen eines Auftrags zu optimieren. Dabei wurde festgestellt, dass die Analogie zwischen den Artefakten Software-Architektur zur Gebäude-Architektur nur bedingt trägt, hingegen die Analogie bei den Rollen Software-Architekt und Gebäude-Architekt recht fruchtbar ist, wie im folgenden erläutert wird.

Abbildung: Software im Netz der Entwicklung



K = Können; W = Wissen; A = Ausdrucksvermögen

Ein wesentliches Problemfeld ist dabei in der sprachlichen Abbildung zu sehen, die eine Zusammenarbeit in diesem Netz ermöglicht. Bisher wurde keine einheitliche und ausreichend tragfähige Sprache entwickelt, die eine ergebnisorientierte Kommunikation zwischen den am Entwicklungsprozess beteiligten Personen gewährleistet. Bei den Plänen der Software-Architekten handelt es sich häufig lediglich um ikonische Abbildungen, die Kommunikationsprobleme verursachen und den Entwicklungsprozess behindern. Eine ausreichend funktionierende Symbolik liegt folglich bisher nicht vor. Auch wissen wir bis heute relativ wenig über die verschiedenen Rollenmuster, die ein Software-Architekt in dem Entwicklungsprozess eines Softwareprodukts spielen kann. Es fehlen allgemein anerkannten Rollendefinitionen im Berufsbild des Software-Architekten. Gleichwohl wissen wir analog aus der Welt der die Städte planenden und die Individualbauten in diesen Stadtregionen konzipierenden Architekten sehr viel über die damit verbundenen Berufsbilder. Wir sind uns auch durchaus der Tatsache bewusst, dass Architekturen in einer erdbebenreichen Region nur dann Sinn machen, wenn dabei die Anforderungen an die Statik der Gebäude und Brücken nicht vernachlässigt werden. Genauso wie ein Architekt bei der Planung eines Hauses die Kundenwünsche mit den technischen Möglichkeiten und den regionalen Umfeldbedingungen in Einklang bringen muss, so sollte ein „Requirements Engineering“ bereits in der Vorphase der Software-Entwicklung zum Tragen kommen. Gerade in der Vorphase kommt es aber zu gravierenden Verständnisproblemen zwischen dem, was der *Auftraggeber* (Owner oder Benutzer) gemäß seinen Wunschvorstellungen eigentlich umgesetzt haben will, und jenem, was letztendlich realisierbar ist. Es ist daher - wie schon seit langem immer wieder gefordert wird – auch bei der Entwicklung einer Software empfehlenswert, noch vor der Grob- und Feinplanung die Kundenwünsche zu analysieren und eine Auswahl der realisierbaren Möglichkeiten festzulegen sowie für diese Aufgaben notwendige Rollen zu definieren. Die Wirkung eines analogen Vorgehens sollte es daher sein, über die Entwicklung von differenzierten Rollenbeschreibungen, Implementierungsregeln und Qualitätsmaßstäben die Ebene des Ausdrucksvermögens in der Software-Industrie zu steigern. Auch in Bezug auf die aktuelle Situation in der Softwareindustrie kann das Verhältnis zwischen **Können**, **Wissen** und **Ausdrucksvermögen** wie folgt zusammengefasst werden:

$$K > W > A$$

Diese Formel basiert auf einem einfachen, aber doch sehr interessanten Gedanken: Je mehr wir etwas auszudrücken vermögen, desto besser kann das Wissen formalisiert werden und desto eher sind positive Rückkopplungs-Effekte auf das allgemeine Können in der Branche zu erwarten.

Kritisch festzuhalten ist in diesem Zusammenhang, dass sich auf der Ausbildungsebene bisher nur wenige Hochschulen mit dem Berufsbild des Software-Architekten eingehend befasst haben und es weitestgehend versäumt wurde, ein Fundament für die Entwicklung bzw. Definition differenzierterer Rollen zu schaffen. Die Curricula der deutschen Hochschullandschaft weisen in dieser Hinsicht vor allem dort massive Defizite auf, wo die informatische, die systemtechnische Seite der betrieblichen Informationssysteme im Vordergrund steht. So hat die Informatik bisher kaum qualifizierte Ausbildungsangebote entwickelt, die zu einer richtungweisenden Entwicklung des Berufsbilds der Software-Architektenzunft geführt haben.

Um analog zum Gebäude-Architekten die Rolle des Advokaten des Kunden wahrzunehmen und seiner strukturierenden und koordinierenden Rolle gerecht werden zu können, benötigt der Software-Architekt fundiertes Wissen aus den verschiedenen Anwendungsdomänen von Software. Im Bereich des betrieblichen Einsatzes von Software ist v.a. die Wirtschaftsinformatik zu nennen, die sich schon länger mit Entwicklung betrieblicher Anwendungsarchitekturen beschäftigt. Besonders prominente Beispiele sind u.a. die Arbeiten von Mertens und Scheer. Wünschenswert wäre daher eine Verbreitung von Ausbildungsprogrammen, wie sie in der deutschen Wirtschaftsinformatik typisch sind: Vorgehensmodelle und Methoden für Anforderungsbeschreibung und -analyse, die auch der Kunde versteht, branchenspezifische Anwendungsarchitekturen oberhalb der systemtechnischen Ebenen sowie Prozess- und Business-Architekturen, die den Spezifika der IT-unterstützten Unternehmung Rechnung tragen. Im Bereich der Anwendung von Software in technischen Systemen (der sog. embedded software) können die Curricula der Ingenieur-Ausbildung an Technischen Hochschulen mit ihrer Verankerung von praktischen Anteilen und starker Anwendungsfeldspezialisierung als Vorbild dienen. Unabhängig vom Anwendungsgebiet sind fundierte Kenntnisse im Software Engineering unabdingbar.

Dabei muß aber deutlich zwischen verschiedenen Software-Architekturen für definierte Anwendungsgebiete, methodischen und verfahrenstechnischen Hilfen für Architektur-Entwürfe bzw. -Realisierungen einerseits und den Aufgaben und Fähigkeiten des Software-Architekten bzw. die Anforderungen an seine Rolle andererseits unterschieden werden. Die notwendige personelle Arbeitsteilung findet in der curricularen Szene der deutschen Informatik zu wenig Beachtung. In den anwendungsorientierten Facetten der Wirtschaftsinformatik gibt es zahlreichere Bestandteile in Lehre und Forschung, die in die hier vorgeschlagene Richtung weisen. Doch erst in der nunmehr bundesweit angestrebten Differenzierung von speziellen Master-Programmen kann die Spezialisierung eines Studienganges für Software-Architekten sichtbar werden, wie dies z.B. an der Fachhochschule in Furtwangen angeboten wird. Hier haben die Studenten der Wirtschaftsinformatik die Möglichkeit, auf der Grundlage eines 6-7 semestrigen Bachelor-Studiengangs und in Verbindung mit einem Praxisjahr, einen weiteren Abschluss zum „Application Architecture Master“ bzw. „Business Consulting Master“ zu erwerben.

Aus Praxissicht sind bereits erste Tendenzen einer arbeitsteiligen Software-Architektenwelt zu erkennen. So teilt sich z.B. die Rollenvergabe im sd&m-Modell für die Peer-Organisation eines Kunden in die zwei Aufgabenbereiche „Technik“ und „Anwendung“. Beide Aufgabenbereiche werden von einem bereichsverantwortlichen Chefdesigner betreut und sind in weitere für Designer-Rollen vorgesehene Aufgabenbereiche unterteilt. Basierend auf den Projekterfahrungen von sd&m hat sich in der Praxis ein hausintern entwickeltes Anforderungsmanagement als besonders hilfreich erwiesen, das in enger Zusammenarbeit zwischen dem Chefdesigner und dem Kunden ein iteratives „Herantasten“ an die Anforderungen des für den Kunden zu erstellenden Gewerkes vorsieht.

Damit derartige Best-Practice-Beispiele in Deutschland Verbreitung finden und somit zu einer branchenweiten Qualitätsverbesserung beizutragen vermögen, müssen die Bedingungen – unter denen ein Informations- und Erfahrungsaustausch in der Softwarebranche stattfinden kann – verbessert werden. Deutschland, als ein traditionell qualitätsbewusstes Land, hat jedoch auch die Chance, Qualitätsstandards zu setzen und den Aufbau eines arbeitsteiligen Berufsbilds des Software-Architekten aktiv mitzugestalten. Es wäre bedauerlich, wenn diese Chance nicht genutzt würde. Leistungen deutscher Software-Architekten könnten ein beachtenswerter Exportartikel in einer globalisierten Welt werden, die unsere besonderen Tugenden wie Kreativität, Zuverlässigkeit, Chaosresistenz und Glaubwürdigkeit zu schätzen weiß. Zugleich wäre eine mögliche Basis geschaffen, auf der es eine europäische Anstrengung zur Entwicklung einer international wettbewerbsfähigen eigenen Software-Industrie zu schaffen. Um noch eine Analogie zu bemühen: Der Aufbau eines Unternehmens vergleichbar den „AIRBUS-Industries“.

Forderungen

- **Definition der notwendigen Fähigkeiten von Software-Architekten**

Um sowohl die Aufgaben als auch die verschiedenen Rollen im Berufsbild des Software-Architekten präzise definieren zu können, müssen seine verschiedenen Fähigkeiten transparent dargelegt werden können. Nachfolgend seien einige der wesentlichen Fähigkeiten eines Software-Architekten exemplarisch aufgeführt:

- Generell ein „Sound sense of business and technical strategy“.
- Tiefes Verständnis der fachlichen Anforderungen.
- Genaue Kenntnisse der verwendeten Technik.
- Delegations- und Durchsetzungsfähigkeit.
- Teamführungsgeschick bezogen auf variierende und teilweise virtuelle Gruppen.
- Unternehmenspolitisches Verständnis.
- Glaubwürdigkeit kraft fachlicher Kenntnisse und Fähigkeiten.
- Überzeugungskraft und womöglich ein kreatives Charisma.

Verständlicherweise werden die hier genannten Fähigkeiten nicht immer in einer Person zu finden sein. Daher sind die speziellen Fähigkeiten durchaus auch Architektenteams zuzuordnen, so wie etwa auch Entwurfs- und Ausführungsarchitekten zusammen arbeiten.

- **Klare Abgrenzung der Aufgabenbereiche**

Damit zur professionellen Abwicklung von Softwareprojekten Aufgaben eindeutig vergeben werden können, ist klar zu definieren, welche davon einem Software-Architekten zuzuordnen sind. Die des Software-Architekten sind dabei von jenen abzugrenzen, die in den Verantwortungsbereich eines Projektmanagers oder eines Projektcontrollers fallen.

- **Präzisierung der Rollendefinition innerhalb des Berufsbilds des Software-Architekten**

Eine Präzisierung der Rollenbeschreibung innerhalb des Berufsbildes der Software-Architekten geht mit der zuvor genannten Forderungen nach der Abgrenzung von Aufgabenbereichen einher. So müssen auf der Basis der eingeforderten Fähigkeiten konkrete Aufgabenbeschreibungen präzisiert werden. Aus diesem Procedere ist auch die Rolle des Software-Architekten in seiner Beauftragung und Verantwortung für den jeweiligen Auftraggeber herzuleiten.

- **Einschaltung von unabhängigen Software-Architekten bei Ausschreibungsprozessen**

Sowohl bei öffentlichen Projekten, als auch bei Projekten von Wirtschaftsunternehmen und NPOs (Non Profit Organisation) macht es Sinn, bei der Vergabe größerer Aufgabenstellungen, einen unabhängigen Software-Architekten einzuschalten. Diesem unabhängigen Software-Architekten kommt u.a. die Aufgabe zu, sich bereits bei der Formulierung der Kundenwünsche kritisch über die Realisierungsmöglichkeiten zu äußern, Risiken entsprechend abzuwägen und mit dem Kunden auf der Basis eines architektonischen Entwurfs eine zukünftige Projektdefinition zu erarbeiten.

- **Abbau von Ausbildungsdefiziten**

Bisher gibt es nur wenige Hochschulen und Universitäten, die qualifizierte Curricula im Rahmen der Ausbildung zum Software-Architekten anbieten. Eine Tradition der „Bauhütten“ wie sie auch in der IT-Welt zu beobachten ist, reicht zur dringend notwendigen Industrialisierung der Softwarebranche nicht aus. Zur Behebung der aktuellen Ausbildungsdefizite in Deutschland ist es daher zwingend notwendig, dass Ausbildungsangebot auf diesem Gebiet konsequent und auf ein hohes professionelles Niveau zu erweitern. Wie an den Technischen Universitäten in vielen ingenieurwissenschaftlichen Fächern üblich, sollten die Professoren auch selbst erfolgreiche und womöglich weiterhin praktizierende Software-Architekten sein. Aus der o.a. Notwendigkeit von Ausbildung in Anwendungsgebieten zusätzlich zu einer fundierten Software Engineering Ausbildung ergibt sich eine Spezialisierung der Ausbildung entsprechend der verschiedenen Anwendungsgebiete. Dies entspricht auch der Ingenieur- und Architektenausbildung, die nach Vermittlung allgemeiner Grundlagen sehr anwendungsfachbezogen verläuft. Die Curricula der Wirtschaftsinformatik mit ihrem Bezug zur betrieblichen Anwendung von Software sind daher eine gute Ausgangsbasis für die Ausbildung von Software-Architekten für betriebliche Anwendungen, sofern sie eine fundierte software-technische Ausbildung beinhalten. Software Engineering Curricula eignen sich als Basis zur Ausbildung von Software-Architekten, wenn sie Elemente aus der Anwendungsdomäne verankern. Im Bereich der Ausbildung von Architekten für eingebettete (Software-) Systeme bietet die ohnehin technisch orientiertere Kerninformatik eine gute Ausgangsbasis, sofern auch elektrotechnische Grundlagen und Software Engineering Wissen vermittelt werden.

Unabhängig vom Anwendungsgebiet ist die Architektenausbildung aus dem Bauwesen insofern als vorbildlich zu betrachten, weil zur Zertifizierung neben einem Hochschulstudium auch Praxiserfahrung durch die erfolgreiche Bearbeitung von Projekten nachgewiesen werden muss.

- **Förderung des Informations- und Erfahrungsaustauschs**
Im internationalen Vergleich schneidet Deutschland bei der Förderung des industriellen Informations- und Erfahrungsaustauschs leider nicht immer gut genug ab. Eine Förderung von Round-Table-Initiativen, wie sie etwa aus der Bay Area der nordamerikanischen Westküste bekannt sind, könnte auch hierzulande dazu beitragen,entwicklungsspezifische Probleme und Herausforderungen in der Software-Industrie besser zu thematisieren. Es ist daher eine Aufgabe der einschlägigen Industrie, den Austausch aktiv zu suchen und personen- oder gruppenbezogene Initiativen zu unterstützen. Innerhalb der Gesellschaft für Informatik e.V. (GI) als Berufsvertretung der Informatiker hat sich der Arbeitskreis „Software-Architektur“ etabliert, der sich mit der Herausgabe des deutschsprachigen Handbuchs zur Software-Architektur beschäftigt. Eine Weiterentwicklung dieses Arbeitskreises über dieses Ziel hinaus zu einer eigenständigen Fachgruppe der GI oder als Mit-Initiator des Software-Bauhauses gilt als wünschenswert.
- **Entwicklung und Veröffentlichung von Best Practice Beispielen**
In Unternehmen erarbeitete Best Practice Beispiele können nur dann zu einer Verbesserung von Produkten innerhalb einer ganzen Branche führen, wenn diese in geeigneten Foren und Medien publiziert werden. Die Softwareindustrie ist daher nicht nur aufgefordert, die Entwicklung von Best Practice Beispielen voranzutreiben, sondern auch dazu aufgerufen, die Erarbeitung und Veröffentlichung von Best Practice Beispielen auf diesem Gebiet zu unterstützen.
- **Stimulans der Arbeitsteilung in der Softwarebranche**
Eine Arbeitsteilung in der Softwarebranche sollte zum Ziel haben, die Bedürfnisse des Kunden (Owner oder Benutzer) besser zu verstehen und bedienen zu können. Das Berufsbild des Software-Architekten braucht in der Supply-Chain bzw. im Netz der Software-Entwicklung eine klare und geachtete Position. Es ist daher die Aufgabe von Verbänden, Unternehmen, Forschung und Lehre sowie der Medien die hier vorgestellte Arbeitsteilung maßgeblich und glaubwürdig voranzutreiben.
- **Deutschland als Vorreiter - Definition allgemein anerkannter Qualitätsbegriffe**
Weltweit fehlt es an anerkannten Qualitätsbegriffen. Daraus ergibt sich für Deutschland die Chance, eine internationale Vorreiterrolle zu übernehmen und Akzente bei der Qualitätsbestimmung zu setzen. Deutsche Software-Architekten könnten – bei all dem, was an grundlegenden und weltweit akzeptierten theoretischen Anregungen für die ganze Branche aus dem deutschsprachigen Raum hervorgebracht wurde, ein besonderes Markenzeichen werden.
- **Eine Konzentration der Entwicklungselite – ein neues Software-(System-)Bauhaus**
Die Avantgarde der Software-Architekten sollte, gemeinsam mit anderen System-Entwicklern, eine spezielle Wirkungsstätte mit ähnlicher Konzeption und Struktur wie das legendäre Weimarer Bauhaus erhalten. Damit könnte eine besondere wechselseitige Befruchtung von Hard- und Software-Entwicklungen in einer neuen funktionellen Klarheit im Interesse alter und neuer Anwendungen kreiert werden. Insbesondere die Eigenschaften der Bauhaus-Idee, wie die enge Verzahnung von akademischer Ausbildung und anwendungsfachspezialisierter Praktika, die Forderung nach Anwendbarkeit der Produkte, enge Betreuungsverhältnisse und nicht zuletzt auch der Führungs- und Innovationsanspruch in seinem Feld, sollten als Vorbild für eine Ausbildung von Software-Architekten dienen.

Westerland/Sylt, im September 2004

Gerhard Barth, Frankfurt am Main
Ulrich Dietz, Villingen-Schwenningen
Klaus Höring, Bergisch Gladbach
Cuno Pfister, Zürich (Schweiz)
Sebastian Saxe, Altenholz
Johannes Siedersleben, München
Marcus Wiebcke, Hamburg

Dirk Buschmann, Köln
Eduard Heindl, Furtwangen
Stefan Kirn, Hohenheim
Ralf Reussner, Oldenburg
Frank P. Schmitz, Berlin
Norbert Szyferski, Köln/Westerland

Dieses Memorandum wird auch von folgenden Personen unterstützt:

Oliver Günther, Berlin

Cuno Pfister, Zürich, Schweiz

Firuz Heydarpoor, Hamburg

Wolfgang Weck, Zürich, Schweiz

Matthias Bohlen, Meckenheim

Stand: 30.03.2005