

Herausgegeben von Norbert Szyperski, Golo Winkler, Dietrich Seibt, Kai-In Voigt
und Rudolf Pospischil

Martin Engelen/Jens Homann (Hrsg.)

Virtuelle Organisation und Neue Medien

Workshop GeNeMe99
Gemeinschaften in Neuen Medien

TU Dresden, 28./29.10.1999



JOSEF EUL VERLAG
Lohmar · Köln

Reihe: Telekommunikation @ Mediendienste · Band 6

Herausgegeben von Prof. Dr. Dr. h. c. Norbert Szyperski, Köln, Prof. Dr. Udo Winand, Kassel, Prof. Dr. Dietrich Seibt, Köln, Prof. Dr. Rainer Kuhlen, Konstanz, und Dr. Rudolf Pospischil, Brüssel

PD Dr.-Ing. habil. Martin Engelen
Dipl.-Inform. (FH) Jens Homann (Hrsg.)

Virtuelle Organisation und Neue Medien

Workshop GeNeMe99
Gemeinschaften in Neuen Medien

TU Dresden, 28./29.10.1999



JOSEF EUL VERLAG
Lohmar · Köln

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

GeNeMe <1999 Dresden> :

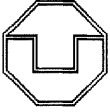
GeNeMe 99 : Gemeinschaften in neuen Medien ; Dresden, 28./29.10.1999, an der Fakultät Informatik der Technischen Universität Dresden / Technische Universität Dresden, Fakultät Informatik, Institut für Informationssysteme, Forschungsgruppe "Entwurfsmethoden und Werkzeuge für Anwendungssysteme". Martin Engeliien ; Jens Homann (Hrsg.). – Lohmar ; Köln : Eul, 1999

(Reihe: Telekommunikation @ Mediendienste ; Bd. 6)
ISBN 3-89012-710-X

© 1999

Josef Eul Verlag GmbH
Brandsberg 6
53797 Lohmar
Tel.: 0 22 05 / 91 08 91
Fax: 0 22 05 / 91 08 92
<http://www.eul-verlag.de>
eul.verlag.gmbh@t-online.de
Alle Rechte vorbehalten
Printed in Germany
Druck: Rosch-Buch, Scheßlitz

**Gedruckt auf säurefreiem, 100% chlorfrei gebleichtem,
alterungsbeständigem Papier nach DIN 6738**



Technische Universität Dresden

Fakultät Informatik • Institut für Informationssysteme

Forschungsgruppe „Entwurfsmethoden und Werkzeuge für Anwendungssysteme“

PD Dr.-Ing. habil. Martin Engeliem
Dipl.-Inform. (FH) Jens Homann
(Hrsg.)

Dresden, 28./29.10.1999

GENEME99

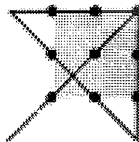
Gemeinschaften in Neuen Medien



*Workshop zu Organisation, Kooperation und Kommunikation
auf der Basis innovativer Technologien*

Forum für den Dialog zwischen Wissenschaft und Praxis

an der
Fakultät Informatik der Technischen Universität Dresden



Gefördert von der Klaus Tschira Stiftung,
gemeinnützige Gesellschaft mit beschränkter Haftung

sowie unter Mitwirkung der
GI-Regionalgruppe Dresden

am 28./29.10.1999
in Dresden

B. Ausgewählte Ansätze zur Entwicklung und Technologie von virtuellen Gemeinschaften

B.1. Gruppenorientiertes Requirement Engineering auf der Basis von Lotus Notes

Prof. Dr. R. Liskowsky

Dipl.-Ing. R. Pjater

Dipl.-Inf. H. Stelter

Technische Universität Dresden

Zusammenfassung

Alle Projektarbeit insbesondere zur Schaffung verteilter Gemeinschaften, die mit neuen multimedialen Informationen arbeiten, beginnt mit einer tiefgründigen Anforderungsanalyse. Wenn diese frühe Phase der Softwareentwicklung ebenso von einem verteilt arbeitendem Team, bestehend aus Auftraggebern, Auftragnehmern, Systemanalytikern, Endnutzern u.a., durchgeführt wird, sprechen wir von einem gruppenorientierten Requirement Engineering. Gegenwärtig liegen für dessen rechentechnische Unterstützung noch keine befriedigenden Lösungen vor.

Ausgangspunkt der zu entwickelnden Groupwarelösung ist ein allgemeingültiges Objektmodell für Anforderungsdokumente, das alle denkbaren Elemente und Bearbeitungsmethoden des Requirement Engineering enthält. Zusammen mit verschiedenen Möglichkeiten teamorientierter Arbeitsweisen stellt es die Grundlage für die Entwicklung einer verteilten Requirement Lösung mit Lotus Notes dar. Im Beitrag wird ausgeführt, wie von unterschiedlichen Nutzern(Rollen) Pflichtenheft-Elemente bearbeitet, weitergegeben(Workflow), gesteuert und geschützt werden können. Die realisierte Notes-Lösung ist offen, was durch eine sinnvolle Kopplung mit CASE-Tools demonstriert wird. Auf der Basis des Domino-Servers ist eine verteilte Arbeit von Internet-Nutzern möglich.

Die Vorstellung der gruppenorientierten Requirementlösung soll dazu beitragen, sie im nächsten Schritt im Rahmen eines größeren Projektes unter Praxisbedingungen zu erproben.

0 Einleitung

Auf dem Wege zu virtuellen Unternehmen sind viele Anwendungssysteme neu zu überdenken und zu überprüfen, ob sie die weiterentwickelten technischen Möglichkeiten wie Vernetzung, effektivere Architekturen und Kommunikation

ausschöpfen. Das Hauptkriterium ist dabei stets höchste Gebrauchstauglichkeit, also größter Effekt und Komfort für den bedienenden Menschen.

Ob für die Weiterentwicklung bestehender Systeme oder die Schaffung neuer Anwendungen, stets sind systematisch Analysen durchzuführen. Alle Tätigkeiten dieses (frühen) Abschnittes der Systementwicklung bezeichnet man als Requirement Engineering. Unabhängig ob es sich dabei um konventionelle Lösungen oder multimediale Anwendungen handelt, existieren für diesen Prozeß bewährte Vorgehensmodelle, Anforderungsdokumente mit den zugehörigen Spezifikationsmitteln und Methoden [1],[2],[3]. Das Ziel besteht darin alle, sich aus der Analyse ergebenden Anforderungen an das neue System zu ermitteln, mit ihren Abhängigkeiten zu dokumentieren und zu verwalten, so daß sie für den Auftraggeber meßbar und gleichzeitig später Basis für die Validation des entstandenen Systems werden.

In diesem wichtigen Prozeß der Anforderungsermittlung arbeiten normalerweise viele Personen zusammen. Sie bringen unterschiedliche Sichtweisen beruhend auf verschiedenen Geschäftsfeldern, andersgearteten Erfahrungen und differenziertem Wissen aber mit dem gleichen (Gruppen-) Ziel ein, das neue Anwendungssystem entsprechen ihren Einzelanforderungen optimal zu gestalten.

In diesem Beitrag soll ein rechnergestütztes Requirement Management System vorgestellt werden, das speziell für die Gruppenarbeit mehrerer im Netz (Intranet oder Internet) verteilter Teilnehmer ausgerichtet ist. Ausgangspunkt sind im ersten Abschnitt die derzeitig bekannten Techniken des Requirement Engineering, wie sie beispielsweise auch in CASE [4],[5] unterstützt werden. Sie reichen von Formaten für Anforderungsdokumente [2] über bekannte formale Spezifikationsmethoden [6] bis zur Überwachung der Requirement-Erfüllung. Gewissermaßen als Zusammenfassung wird ein allgemeines Modell für Anforderungsdokumente in UML-Notation [7] angegeben, mit dem sich alle Techniken vereinen lassen. Es stellt die Grundlage für die nachfolgende kooperative Rechnerlösung dar. Die Anforderungen, die an eine Gruppenbearbeitung von Requirements zu stellen sind, beinhaltet der zweite Abschnitt. Ausgehend von bekannten Vorgehensmodellen, wobei das V-Modell [1] dominiert, werden die unterschiedlichen Rollen der Teammitglieder postuliert. Je nach deren typischen Aufgabengebieten lassen sich differentielle Zugriffsrechte vergeben, die bei gezielter Arbeitsteilung sogar die Basis für einen Workflow zur Aufgabenbearbeitung bilden können [8].

Auf Basis der vorstehenden Anforderungsanalyse wird im dritten Abschnitt die Lösungskonzeption für das gruppenorientierte Requirement Management System entwickelt. Es zeigt sich, daß mit dem Groupwaresystem Lotus Notes/Domino die geforderten Funktionen sehr gut zu realisieren sind. Der gesamte Aufbau der Notes-Applikation für das Requirement Engineering ist dann Inhalt des vierten Abschnittes.

Gelöst wurde die Erarbeitung von Requirement-Dokumenten in der Gruppe, die Zustandsüberwachung der Einzelforderungen und das Versenden zu Teammitgliedern bis hin zur Definition von Workflows. Speziell für die Einbeziehung formaler Methodenergebnisse und zum Export von Requirementdaten wurde die Kopplung zu einem CASE [9] exemplarisch hergestellt. Mit der Bereitstellung der Notes-Ansichten und Dokumente im Internet wird gewährleistet, daß alle Projektbeteiligten plattformunabhängig gemeinsam an der Gestaltung von Anforderungsdokumenten arbeiten können.

1 Bewährte Techniken des Requirement Engineering (RE)

Die nachfolgende Aufzählung bewährter Techniken oder Spezifikationsmittel ist quasi als Anforderungssammlung für die neue gruppenorientierte Requirementlösung zu verstehen. Die zusammengetragenen Mittel, die in langjähriger RE-Arbeit entstanden sind, sollen auch die Basis des kooperativen Wirkens der beteiligten Nutzer und Analytiker sein.

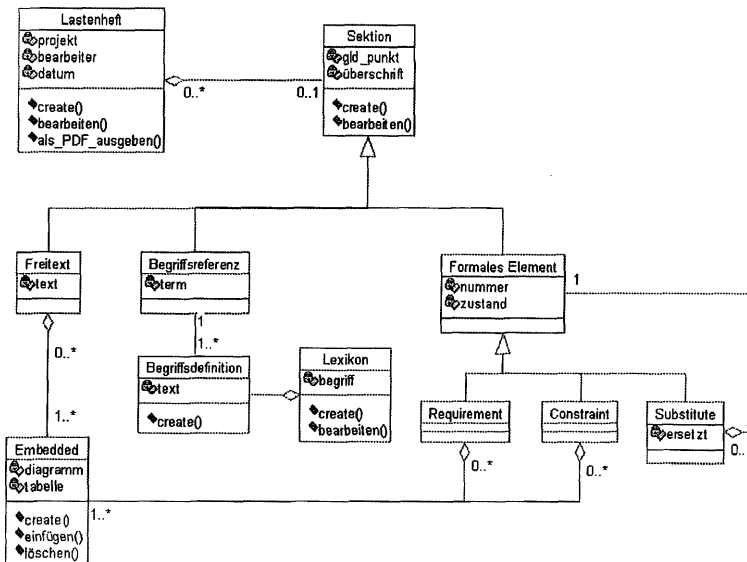


Abb. 1: Dokumentenmodell eines Lastenheftes

Das Anforderungsdokument, das alle Requirements mit ihren Abhängigkeiten untereinander enthält, kann beispielsweise ein Lasten- oder Pflichtenheft [2] sein. Eine

vorgegebene Gliederung stellt stets eine Richtschnur zum Handeln dar. Viele CASE lassen auch beliebig gegliederte Anforderungssammlungen zu. Unabhängig von der konkreten Ausprägung kann man dafür ein sehr variabel zu handhabendes Dokumentenmodell ableiten. In UML-Notation [7] ist es zusammen mit den wichtigsten Objekten, hier am Beispiel eines Lastenheftes, in Abb. 1 dargestellt.

Jedes Objekt hat neben dem eindeutigen Namen bestimmte Attribute und Methoden, die auf sie angewandt werden können. In Abb. 1 sind zum besseren Verständnis nur einige wesentliche enthalten. Die hierarchische Gliederung des Dokumentes folgt aus den dezimal klassifizierten *Sektionen*, die einzelne Gliederungspunkte präsentieren. Diese wiederum bestehen aus *freiem Text*, *Begriffsreferenzen* zur eindeutigen *Begriffsdefinition* in einem *Lexikon* und aus *formalen Elementen*. Letztere sind bezeichnete (als *Requirement* oder *Constraint*) referenzierbare Abschnitte, die mittels des dritten formalen Elements *Substitute* durch andere (neue) Requirements oder Constraints ersetzt werden können. Der Bearbeitungsstand formaler Elemente wird im Attribut *Zustand* festgehalten. In Freitext bzw. formalen Elementen können auch graphische oder Tabellenelemente (*Embedded*) eingebettet sein. In der Regel sind dies Anforderungsspezifikationen in Form von Diagrammen oder Matrizen, wie z.B. Use Case, Entity Relationship Diagramme und Zustandsübergangsmatrizen. Sie können aus CASE stammen, womit ein direkter Anknüpfungspunkt zu entsprechenden Tools gegeben wäre. Das Modell in Abb. 1 läßt sich leicht um weitere Objekte erweitern. Als ein Beispiel ließen sich *Kategorien* als zusätzliches Kennzeichnungsobjekt einführen, mit dem Sektionen, formale Elemente und Begriffe übergeordneten Sachbegriffen zugeordnet und damit nach Wunsch gruppiert werden könnten.

Die Requirement-Erfüllung im Laufe der Aufgabenbearbeitung kann man relativ einfach über eine Zustandsabfrage sämtlicher formalen Elemente erhalten. Von der Semantik her gilt, daß Requirements unbedingt einzuhaltende Ziele bzw. Anforderungen bezeichnen, während Constraints den Charakter von Voraussetzungen, Nebenbedingungen haben. Bei der Zustandsdefinition sollte man sich zweckmäßig an standardisierte Stufen anlehnen, wie sie das V-Modell vorschreibt [1].

Die Erfassung der Abhängigkeiten zwischen formalen Elementen ist in Abb. 1 nicht explizit dargestellt. Solche Abhängigkeiten können existieren, wenn die Bearbeitung eines (nachfolgenden) Requirements von der Erfüllung eines oder mehrerer (vorhergehender) Requirements/Constraints abhängt. Diese logischen Verkettungen sind bei der Pflichtenheftumsetzung in der Vorgehenssteuerung entsprechend zu beachten. Wenn es das rechnergesteuerte Prozeßmanagement nicht berücksichtigt, kann man auch sogenannte Anforderungsverfolgungs-Matrizen aufbauen, in denen die

Matrixelemente die Abhängigkeitsbeziehungen verdeutlichen. Unter Bereitstellung verschiedener Sichten wird dieser Weg in RequisitePro [10] gegangen.

In dieser kurzen Aufzählung sind unserer Meinung nach alle technischen Elemente enthalten, die gegenwärtig die Arbeit mit Anforderungsdokumenten bestimmen. Im Mittelpunkt steht das sehr variable Dokumentenmodell, das es gestattet, alle Arten von Spezifikationen, seien sie textuell, graphisch oder semigraphisch, flexibel zu erfassen, ihre Abhängigkeiten zu verfolgen sowie im Laufe der Projektdurchführung zu verwalten. Dieses allgemeine Modell, das auf Lauber [11] zurückgeht, stellt damit eine sehr gute Grundlage für die gruppenorientierte Arbeit im Requirement Engineering dar.

2 Anforderungen an das gruppenorientierte Requirement Engineering

Die Problematik des RE, besonders für größere Projekte, besteht in der optimal organisierten Zusammenarbeit der für diese Phase des SW-Entwicklungsprozesses verantwortlichen Projektbeteiligten. Je intensiver und effektiver die Zusammenarbeit zwischen den Beteiligten des Auftragnehmers sowie dem Auftraggeber ist, um so effektiver können die nachfolgenden Projektphasen realisiert werden. In der Literatur mehren sich in letzter Zeit Ansätze, die versuchen, in der Phase des RE die Teamarbeit rechnergestützt zu organisieren [12], [13]. Trotzdem sind systematisch alle Anforderungen zu untersuchen, die die gruppenorientierte Arbeit in dieser frühen Phase der Softwareentwicklung stellt. Entsprechend des durch den Auftragnehmer genutzten Vorgehensmodell (z.B. V-Modell der Bundeswehr) können die beteiligten Rollen und Aktivitäten definiert werden. Sie sind die Basis für eine Spezifizierung der daraus entstehenden Workflows.

Ausgangspunkt für die Zusammenstellung der Anforderungen an die Gruppenarbeit und an die entsprechende rechentechnische Unterstützung, speziell auch für das RE, sind die unterschiedlichen Ausprägungen und wichtigsten Ebenen der Zusammenarbeit zwischen den Gruppenmitgliedern. [14], [15].

Ebenen:

Kommunikation - Vermittlung von Informationen (Mitteilung, Aufforderung, Anfrage...)

Koordination - Kontrollieren oder Handhaben von wechselseitigen Abhängigkeiten zwischen Aktionen der Partner mit evtl. gemeinsamer Ressourcennutzung

Kooperation - Tätigsein von zwei oder mehr Partnern, die bewußt planvoll, aufeinander abgestimmt ein gemeinsames *Gruppenziel* anstreben

Auf dieser Basis muß eine Unterstützung der Gruppenmitglieder (Projektbeteiligten) durch ausgewählte Werkzeuge (Software) innerhalb des RE erfolgen. Diese Werkzeuge werden allgemein als Groupware bezeichnet. Unter Groupware wird nach [16] sowohl

die hardware- als auch softwaremäßige Realisierung der Gruppenarbeit verstanden, wobei der Begriff Groupware oft nur die softwaremäßige Realisierung beinhaltet.

Nach bekannten Klassifikationsschemata von Groupware [14] sind für das gruppenorientierte RE folgende nach den Zusammenarbeitsebenen ausgewählte Werkzeugklassen von Bedeutung:

1. Elektronisches Postsystem (E-mail)

Die E-mail ist das Kernsystem zur Unterstützung der Kommunikation. Einbezogen sind dabei auch die Diskussionsforen („Schwarzes Brett“). Schwerpunkt ist die zielgerichtete und schnelle Verteilung von Informationen/Dokumenten, wobei zwei Prinzipien angewendet werden. Nach dem „Send-Prinzip“ leitet man die Informationen per E-mail direkt an den Empfänger. Damit ergeben sich folgende Vorteile:

- der Nutzer wird vom System direkt auf die ankommenden Informationen hingewiesen
- eine einfache Protokollierung der Verteilung aller Informationen und
- die Reduzierung des Informationsvolumens auf das Wesentliche.

Das „Share-Prinzip“ zentralisiert die Daten in einer gemeinsamen Informationsbasis, wobei der Nutzer bei Bedarf nach neuen Informationen suchen muß. Dieses Prinzip ist sehr stark von der aktiven Mitarbeit der Nutzer abhängig.

Auf Grundlage beider Prinzipien können u.a. im RE der Nachrichtenaustausch innerhalb der Gruppe als auch mit dem Auftraggeber realisiert werden. Es können Objekte (z.B. Anforderungsdokumente) übermittelt und diskutiert werden.

2. Koordinationssysteme (Workflow-Management-Systeme)

Diese Systeme unterstützen die Verteilung, die Steuerung und Kontrolle von Aufgaben (mit einem hohen Strukturierungsgrad) innerhalb der Projektgruppe. Ein wichtiger Bestandteil ist dabei die Realisierung von Workflows, denn jedes Dokument im Pflichtenheft unterliegt einem Bearbeitungsablauf. Allgemein kann man sagen, ein Workflow ist eine zum Teil automatisiert ablaufende Gesamtheit von Aktivitäten, wobei die Folge der Aufgabenschritte durch Ereignisse ausgelöst und beendet wird. Ein Aspekt bei der Modellierung des Workflows ist die Zuordnung von Rollen zu Aktivitäten, wobei eine Rolle durch eine Menge von Fähigkeiten und Kompetenzen definiert ist, welche alle Personen aufweisen, die dieser Rolle zugehörig sind(vgl.[14]). Für das gruppenorientierte RE muß wie für jede Aufgabe analysiert werden, welche Aktivitäten notwendig sind und durch welche Rollen sie ausgeführt werden sollen.

In Anlehnung an das V-Modell wurden beispielhaft einige repräsentative Rollen und Aktivitäten ausgewählt.

Rollen:

- Auftraggeber

- Projektmanager
- Systemanalytiker
- Prüfer
- Endnutzer

Diese Rollen können bei Benutzung anderer Vorgehensmodelle erweitert und beliebig modifiziert werden

Aktivitäten:

Grundlage für die benötigten Aktivitäten sind neben dem größeren Vorgehensmodell natürlich die zu bearbeitenden Objekte, hier speziell die des Lasten-/Pflichtenheftes [2].

Typische Aktivitäten sind dort:

- Ausgangssituation/Istzustand beschreiben
- Aufgabenstellung/Sollzustand spezifizieren
- Qualitätsanforderungen definieren
- Systemtechnische Lösung erstellen
-

Im RE durchlaufen beispielsweise die Sektionen des Pflichtenheftes vorgegebene Arbeitsschritte. Diese sind gewissermaßen als untergeordnete Aufgaben zu betrachten. In Anlehnung an [8] können diese Komponenten einen Workflow bilden. In Abb. 2 sind die Aktivitäten Begriffsbestimmung, Auffinden von Aktoren, Definition von Use Case, Requirement-Entwicklung und -Verwaltung enthalten. Eine mögliche Zuordnung der Rollen zu den gewählten Aktivitäten mit den vorhandenen tatsächlichen Ressourcen ist in Abb. 2 ebenfalls gezeigt.

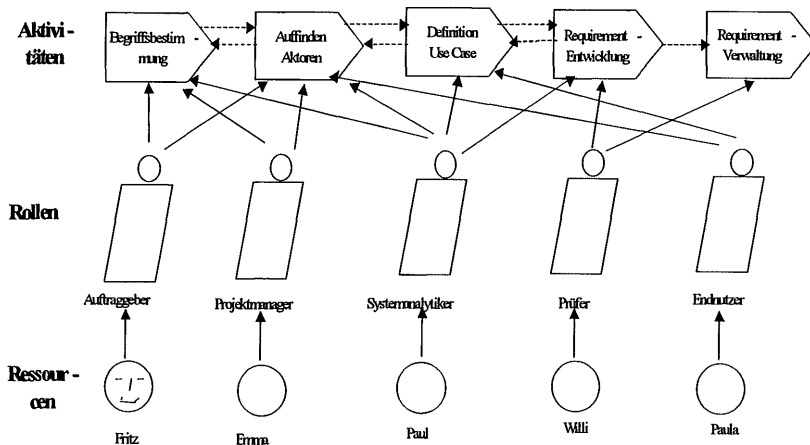


Abb. 2: Prinzip-Workflow des Requirement Engineering

3. Kooperationsysteme (Workgroup-Computing)

Diese Systeme unterstützen die Kooperation von Projektmitgliedern bei der Lösung von Aufgaben mit einem mittleren bis geringen Strukturierungsgrad. Zu diesen Systemen gehören die Entscheidungs- und Sitzungsunterstützungssysteme, die Gruppeneeditoren und Planungssysteme wie z.B. Terminverwaltungssysteme[14]. Sie sind im Requirement Engineering insbesondere bei zeitgleicher synchroner Arbeit der Teammitglieder an Formulierungen von Anforderungsdokumenten bzw. bei der Erstellung formaler Modellkomponenten durchaus denkbar. Nach Meinung der Autoren wird aus Gründen der Effizienz zunächst die asynchrone Zusammenarbeit vorherrschen, bevor später bei positiver Bilanz vielleicht stärker zur zeitgleichen Telearbeit übergegangen wird.

Aus diesem Grunde beschränken wir uns auf die zeitversetzte Gruppenarbeit, wobei die zu schaffende gruppenorientierte Requirement-Lösung folgende weitere allgemeine Anforderungen erfüllen muß[vgl. [17],[18)].

Ausgereifte Sicherheitskonzepte:

Gruppenarbeit erfordert ein hohes Maß an Datenschutz und Datensicherheit. Um die Sicherheit innerhalb des RE zu gewährleisten, ist der Schutz der Einzelobjekte mit entsprechend gestaffelten Zugriffsberechtigungen für die Rollen zu realisieren.

Realisierung von Import- und Exportmechanismen:

Für die weiteren Phasen des Softwareentwicklungsprozesses, die u.a. durch CASE-Werkzeuge unterstützt werden, wäre eine Kopplung zwischen den Informationen und konkreten Daten des Pflichtenheftes und einem entsprechenden CASE sinnvoll. So könnte beispielsweise die Erfüllung der Requirements nach der Entwurfs- oder Implementierungsphase innerhalb eines CASE kontrolliert und ins RE-System zurück übertragen werden. Weiterhin lassen sich zum besseren Verständnis (UML-)Diagramme aus dem Systementwurf in das Pflichtenheft einbinden. In umgekehrter Richtung sollten aus dem Export von Requirements aus dem RE-System erste Entwurfsentscheidungen in CASE ableitbar sein.

Realisierung im Intranet bzw. Internet

Die Zusammenarbeit innerhalb eines Unternehmens bzw. eines virtuellen Unternehmens über lokale Grenzen hinaus kann zu Problemen sowohl hinsichtlich der Infrastruktur als auch der eingesetzten Groupware führen. Eine Forderung des RE ist u.a. die Gewährleistung einer effektiven Zusammenarbeit zwischen allen beteiligten Rollen insbesondere zwischen Auftragnehmer und Auftraggeber. Wenn die Lösung von vornherein unter der Prämisse der Einbeziehung des Internets entwickelt wird, ließe sich das Problem der verteilten Requirement Bearbeitung sehr elegant lösen.

Projektmanagement

Ein Bestandteil des Projektmanagement ist die Terminkontrolle und –überwachung. Von Bedeutung ist die Terminkontrolle der Aktivitäten durch den Projektleiter sowie eine möglichst automatisierte Kontrolle der Einhaltung der Termine für alle Projektbeteiligten. Zusätzlich sollte jeder Bearbeiter seine eigenen Termine und Aufgaben verwalten können. Falls das RE-System das Projektmanagement nicht direkt unterstützt, werden zumindest offengelegte Schnittstellen gefordert, über die externe Werkzeuge anschließbar sind.

3 Entwicklung einer Requirement Engineering Applikation auf der Basis von Lotus Notes/Domino

3.1 Lotus Notes

In den nächsten beiden Kapiteln soll auf Grundlage der Ausführungen in den Abschnitten 1 und 2 der Entwurf einer Software für ein gruppenorientiertes Requirement Engineering erfolgen. Die Merkmale des Requirement Engineering werden in diesen Entwurf einfließen. Der Aufbau der Requirements beruht auf dem Dokumentenmodell, wie es in Abschnitt 1 (Abb. 1) erläutert wurde.

Ein aktuelles und marktführendes Softwareprodukt für die Nutzung von Groupwareapplikationen ist Lotus Notes¹. Es ist ein plattformunabhängiges offenes System für die Entwicklung von Groupwareanwendungen. Lotus Notes bietet gegenüber anderen Softwaresystemen folgende Vorteile:

1. Client-Server-Netzwerk

Für ein gruppenorientiertes Requirement Engineering ist es notwendig, daß den Nutzern alle relevanten Informationen zur Verfügung gestellt werden. Daraus ergibt sich die Forderung, die Daten und Informationen an zentralen Punkten zu verwalten und den Nutzern über Workstations (Clients) diese Informationen zugänglich zu machen. Diese Art der Datenhaltung wird durch ein Client-Server-Netzwerk abgebildet.

Jeder Nutzer der Datenbank besitzt einen Client (Workstation) und hat Zugriff auf mindestens einen Server im Netzwerk. Nur so kann sichergestellt werden, daß auf neue bzw. vorhandene Informationen von jedem Client und von jedem Standort aus zugegriffen werden kann.

2. Webfähigkeit

In der heutigen Zeit hat das Internet in bezug auf die Bereitstellung von Informationen zu den Nutzern an großer Bedeutung gewonnen. Die Clients werden durch Browser und die Datenhaltungsserver durch Webserver ersetzt. Der Notes-Domino-Server kann pa-

¹Copyright © 1985-1999 Lotus Development Corporation

rallend als Webserver betrieben werden. Die Anwendungen können von dem Lotus Notes-Client und dem Webbrowser ausgeführt werden. Die Lotus Notes-Informationen und das Aussehen dieser Informationen werden dynamisch bei einer Internetanfrage als HTML-Seite konvertiert.

3. Datenverteilung in einer Anwendung

Eine Anwendung in Lotus Notes kann aus einer bzw. aus mehreren Datenbanken bestehen und auf unterschiedlichen Servern plaziert sein. Diese Datenbanken sind durch Aktionen oder interne Mechanismen miteinander verknüpft. Es ist aber darauf zu achten, daß bei einem benötigten Datenbankzugriff auch die Berechtigung des Zugriffs auf den jeweiligen Server benötigt wird.

4. Datensatzspeicherung

Die Datenbanken in Lotus Notes enthalten nicht nur die Datensätze sondern auch Gestaltungs-, Steuerungs- und Zugriffsschutzinformationen. Die Datensätze werden in den sogenannten Dokumenten gespeichert. Lotus Notes als Datenbanksystem hat den Vorteil, daß nicht nur strukturierte Daten, also Text- oder Zahleninformationen in den Dokumenten gespeichert werden können, sondern auch unstrukturierte Informationen, z.B. Grafiken, Videos, Charts oder OLE-Objekte aus anderen Programmen. Diese Datensätze werden als „Compounds Documents“ bezeichnet

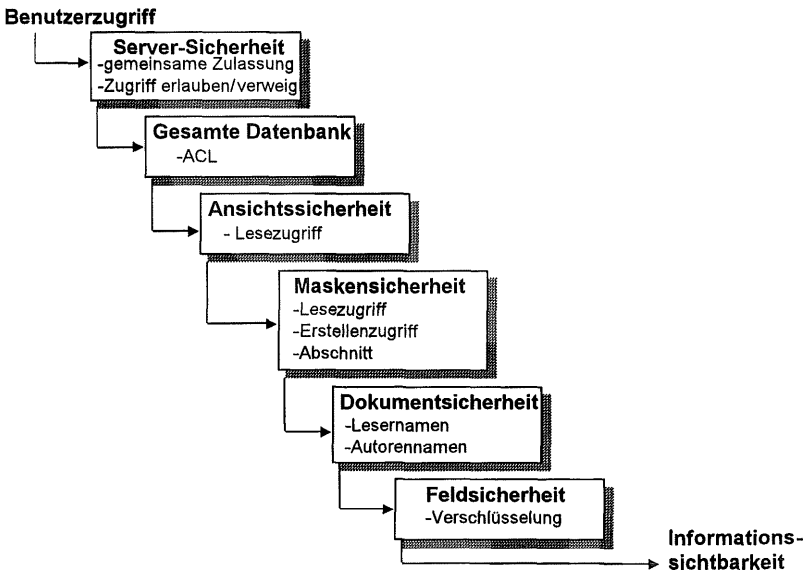


Abb. 3: Sicherheitsstufen der Notes-Objekte

5. Zugriffssicherheit und Datenschutz

Die Systemsicherheit wird in Lotus Notes auf den unterschiedlichsten Ebenen angeboten und realisiert.

3.2 Eigenschaften des Entwurfs

Die aufgeführten Eigenschaften der Lotus Notes-Anwendungen beeinflussen die Programmierung und die Ausführung dieser Anwendungen. Die Applikation zum gruppenorientierten Requirement Engineering soll auf der Basis dieses Groupwaresystems modelliert werden. Das bedeutet, daß die Eigenschaften und die Möglichkeiten von Lotus Notes in diesem Entwurf verstärkt Anwendung finden.

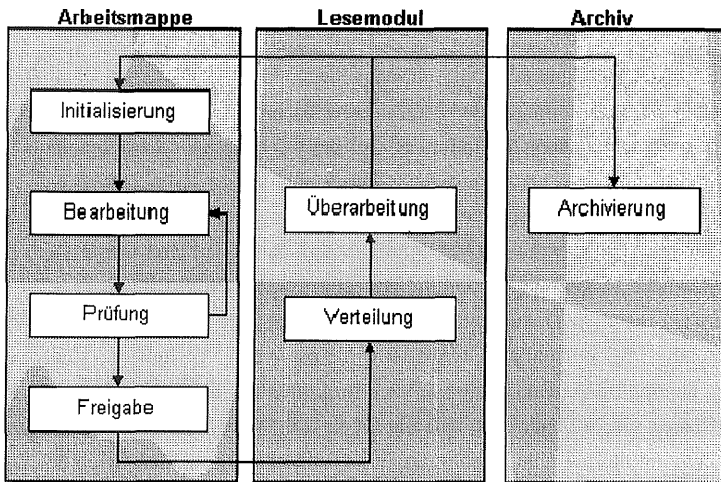


Abb. 4: Zustände der Elemente von Anforderungsdokumenten

Bevor das Modell der Anwendung zum gruppenorientierten Requirement Engineering näher beschrieben wird, müssen Zielfunktion und Ausführungsverhalten der Anwendung eindeutig definiert werden:

Jedes einzelne Element aus dem Lasten- oder Pflichtenheft wird in der Anwendung als ein Lotus Notes Dokument gespeichert. Jedes Requirement und jeder freie Text wird als einzelner eigenständiger Datensatz den Projektbeteiligten zur Verfügung gestellt. Diese Eigenschaft ist notwendig, damit das Pflichtenheft von mehreren Projektbeteiligten erarbeitet werden kann.

Jeder Nutzer erhält eine Arbeitsmappe (Ansicht), in der Dokumente sichtbar sind, für die der Nutzer Aufgaben erledigen muß, z.B. das Erstellen der Pflichtenheftelemente.

Der Initiator des Dokumentes entscheidet, in welcher Gliederungsebene dieses Dokument abgelegt werden soll.

Gleichzeitig muß er die Bearbeiter und Prüfer für dieses Dokument eintragen. Damit ergeben sich für diese Benutzer gewisse Zugriffsregeln zum Bearbeiten des Dokumenteninhaltes und für die Prüfung dieses Inhaltes. Das Dokument besitzt den Status „Initialisierung“. Der Initiator gibt manuell das Dokument in den Status „Bearbeitung“. Der Bearbeiter kann jetzt den Inhalt des Dokumentes zum Pflichtenheft für das Softwareentwicklungsprojekt eintragen und das Dokument entweder zur Prüfung oder zu anderen Bearbeitern schicken. Damit verliert der Bearbeiter den Zugriff auf dieses Dokument. Mit dieser Methode wird verhindert, daß andere Nutzer unbemerkt Informationen eintragen und damit das Dokument und den aktuellen Bearbeitungsstand fälschen können.

Die Prüfung der Dokumente erfolgt auf manuelle Weise. Der Prüfer entscheidet, ob das Dokument den Anforderungen der Auftraggeber entspricht. Bei Bestätigung gelangt es wieder in die Arbeitsmappe des Initiators, welcher das Dokument freigeben kann.

Durch die Freigabe des Dokumentes kann dieses nicht mehr bearbeitet werden.

Sollen trotz Prüfung Änderungen erfolgen, wird von dem alten Dokument eine neue Version erstellt. Nur dieses neue Versionsdokument kann bearbeitet werden.

Über jedes neue Dokument werden die Nutzer entweder durch E-mails (Send-Prinzip) oder durch vorhandene Dokumente in ihrer Arbeitsmappe (Share-Prinzip) hingewiesen.

Da nicht alle Projektbeteiligten registrierte Nutzer dieser Anwendung sind, wird ein Medium benötigt, daß unabhängig von den Strukturen des Lotus Notes-Netzwerkes ist. Heute existiert dieses Medium als Internet. In diesem Entwurf werden die freigegebenen Dokumente im Internet sichtbar sein. Somit ist es möglich, von allen Orten der Welt auf diese Informationen zuzugreifen. Alle Projektbeteiligten, Auftraggeber und Auftragnehmer können an der Erstellung des Pflichtenheftes teilnehmen. Über das Internet können die Kritiken an den vorhandenen freigegebenen Dokumenten erstellt werden. Der Initiator eines Dokumentes kann danach diese Kritiken und Kommentare auswerten und entscheiden, ob das Dokument als eine neue Version überarbeitet werden soll.

4 Funktionen der Notes Applikation für das Requirement

4.1 Aufbau von Requirement - Dokumenten

Da in den Lotus Notes-Datenbanken nur die Dokumente Träger von Informationen sind, müssen neben den Inhalten des Requirement Engineering auch die Steuerungsinformationen mit gespeichert werden. Aus diesem Grund besteht das Lotus Notes-Requirement-Dokument aus folgenden Abschnitten:

- Gliederung,
- Requirementstyp,
- Bearbeitungsworkflow mit dem Zugriffsschutz und
- Inhalt der Requirement-Objekte(vgl. Abb.1).

Jedes Dokument in einem komplexen Anforderungsdokument unterliegt einer Gliederung. Die Gliederung, die auch die Einordnung der Requirements im Lasten- bzw. Pflichtenheft bestimmt, ist von den Systemadministratoren parametrisierbar. Selbständig können die Anwender entscheiden, wie tief ihre Dokumente hierarchisch gegliedert sein sollen.

Jedes Dokument in dem Lasten- oder Pflichtenheft kann ein formales Element, eine Begriffsreferenz oder ein Freitext sein. Das Requirement-Objekt bei einem formalen Element ergibt sich aus Abbildung 1.

Jedes Dokument beispielsweise des Pflichtenheftes durchläuft während der Erarbeitung festgelegte Status und unterliegt innerhalb dieser Status bestimmten Zugriffsschutzmechanismen. Da sich die Status in einem Workflow von Projekt zu Projekt oder von Unternehmen zu Unternehmen unterscheiden, sind diese ebenfalls frei wählbar. Die Status werden durch einen Bezeichner, einen Identifikator sowie die Verkettung der Status definiert.

Die Requirement-Dokumente, der zentrale Teil dieser Anwendung, ergeben als Ganzes z. B. ein Pflichtenheft. Es besteht aus Texten, welche bestimmten Kapiteln oder Abschnitten zugeordnet werden können. Während des gesamten Lebenszyklus wird vom Dokument auf die Module „Gliederung“ und „Workflow“ zugegriffen und die benötigten Informationen in das Dokument geschrieben. Für die Nutzer bleiben allerdings die Administrationsmodule transparent.

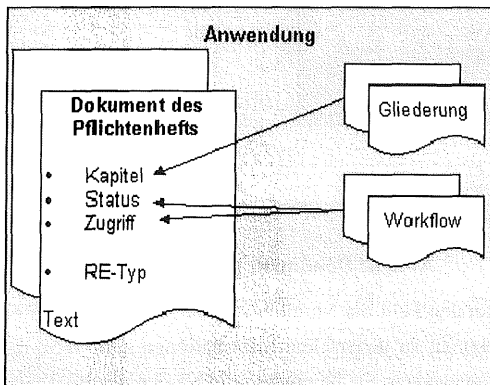


Abb. 5: Dokumentenaufbau im Pflichtenheft

Jeder Nutzer besitzt eine Arbeitsmappe. Es ist eine Ansicht in der Lotus Notes-Datenbank. In der Arbeitsmappe erscheinen für jeden Nutzer seine persönlich zugeordneten Aktivitäten. Diese Aktivitäten sind in diesem Entwurf den Lotus Notes Dokumenten des Pflichtenheftes gleichzusetzen. Die Aktivitäten richten sich nach der Definition des Workflows.

Während der Erstellung eines neuen Dokumentes muß der Ersteller (*Initiator*) die Entscheidung über die Einordnung des Dokumentes in das Pflichtenheft treffen. Dies geschieht über eine Dialogbox, welche zu Beginn des Erstellvorganges angezeigt wird (Abb.6).

Speichern und Schließen		Protokoll		Aufgabe		Nxt	
▼ Titelinformation							
Gliederungspunkt:		3. Aufgabeneinstellung (Sollzustand) 3.3. Hauptfunktionen					
Titel:		Konverter					
Requirementstyp:		Requirement ▾ Klasse ▾					
Status im gesamten Projekt:		Analyse ▾					
▼							
Initiator:		Heiko Stelter/USER_DD/ULC					
Bearbeiter:		IP					
Prüfer:		P					
Status:		Initialisierung					
Version:		1					
Prüfungsprotokoll:							
▶ Kommentare, Kritiken							

Abb. 6: Dokument nach Erstellung

Der Initiator hat im ersten Feld die Möglichkeit, das Dokument weiter zu untergliedern und diesem einen Namen zu geben. Die Informationen zum Workflow werden in der vorletzten Sektion eingetragen. Sie beinhalten die Namen des Bearbeiters und der Prüfer für dieses Dokument.

Beim Betätigen der Aktionsschaltfläche „*Speichern und Schließen*“ wird das Dokument gespeichert, aber noch nicht dem Bearbeiter zugänglich gemacht. Es befindet sich also noch im Bereich des Initiators und damit in seiner Arbeitsmappe.

Erst mit dem Betätigen der Aktionsschaltfläche „*Nxt*“ wird in den nächsten Workflowstatus geschaltet. Jetzt werden, je nach Definition des Workflows, die Rechte am Dokument geändert und das Dokument weitergereicht. Gleichzeitig wird der neue Inhaber, in dieser Applikation der Autor des Dokumentes, mit einer E-mail auf die neue Aktivität hingewiesen. Die Mail enthält eine Dokumentverknüpfung (Doclink) auf das Dokument des Pflichtenheftes, und der neue Inhaber kann schnell, sofort über die Mail auf dieses Dokument zugreifen.

Somit unterstützt die Anwendung zum gruppenorientierten Requirement Engineering nicht nur das Share-Prinzip für den Informationszugriff, sondern auch das Send-Prinzip (siehe Abschn. 2).

Damit größere Änderungen revisionssicher abgelegt werden können, muß der alte Informationsinhalt sichtbar sein. Für diese Funktionalität erscheint, je nach Definition des Workflows, eine Aktionsschaltfläche „*Neue Version*“. Sie ist nur für den Initiator des Dokumentes sichtbar, da dieser verantwortlich für das Dokument ist. Diese Schaltfläche speichert den aktuellen Zustand des Dokumentes als ein Antwortdokument ab (siehe [20]). Das Antwortdokument kann von den Nutzern dieser Anwendung nicht mehr geändert werden.

Nachdem die Dokumente freigegeben worden sind, kann auf diese über das Internet zugegriffen werden. Voraussetzung ist ein aktivierter HTTP-Task auf dem Lotus Notes-Server. Für den Internetnutzer, z.B. den Auftraggeber, hat die Datenbank folgendes Aussehen:

Gliederungspunkt:	1.2. Projektumfeld 1.2.2. Wirtschaftliche Zusammenhänge
Titel:	ohne Titel
Formales Element:	Freitext
<p><u>Pflichtenheft</u> Text</p> <p>▼ Kommentare, Kritiken</p>	
<input type="button" value="Bereitstellen"/>	

Abb. 7: Internet-Dokument

Der Internetnutzer sieht nur die Ansicht des Pflichtenheftes mit den freigegebenen Dokumenten. Nach dem Öffnen der Dokumente kann der Internetnutzer Bemerkungen und Kritiken zu diesem Pflichtenheft eintragen. Mögliche Änderungen werden sofort nach dem Speichern auf der Seite des Lotus Notes-Client sichtbar. Die Ersteller des Pflichtenheftes können sehr schnell reagieren und die Kritiken als Ausgangspunkt für seine weitere Bearbeitung nehmen. Damit alle Projektbeteiligten über den Status der Requirements, welche im Pflichtenheft aufgeführt sind, informiert sind, existiert im Dokument ein Statusfeld „Projektstatus“. In diesem Feld wird der eingetragene Projektstatus auch über das Internet sichtbar. Somit wird eine enge Zusammenarbeit zwischen allen Beteiligten des Requirement Engineerings gewährleistet.

Zusätzlich kann jeder Nutzer ein Dokument aus dem Pflichtenheft in Form einer Aufgabe oder einer Wiedervorlage in seine eigene Maildatenbank (Kalender) stellen.

Die gewählte Lösungskonzeption für das gruppenorientierte Requirement Engineering besitzt den Vorteil, daß durch die Konfigurierbarkeit der Programmparameter, wie z.B. der Gliederung oder des Workflows, die Anwendung universal einsetzbar ist. Bei kleinen Projekten können sowohl der Umfang der Gliederung als auch die Stationen eines Workflow geringer ausfallen. Außerdem wird durch die Verfügbarkeit im Internet das Informationsangebot allen Projektbeteiligten zu jeder Zeit bereitgestellt. Trotz dieser vielfältigen Möglichkeiten wird der Zugriffsschutz nie vernachlässigt. Die Sicherheit der Informationen ist stets gewährleistet.

4.2 Schnittstellen zu CASE

Alle Werkzeuge, welche im Entwicklungsprozeß eingesetzt werden, benutzen entweder eine gemeinsame Datenbasis (Datenteilung) oder transformieren die Daten von einem Werkzeug zu einem anderen Werkzeug (Datenaustausch oder Datenverbindung). Die gemeinsame Datenbasis erfordert ein einheitliches Datenmodell. Es stellt die Basis für die Kommunikation und Integration der Werkzeuge untereinander dar[4].

Die Informationen müssen von Werkzeug zu Werkzeug transformiert werden. Dafür streben internationale Organisationen einfache standardisierte Austauschformate an, neuerdings auf Basis XML und UML z.B. das Format XMI [21]. Leider bieten aber die wenigsten CASE-Tools zur Zeit diese Formate an, so daß man selbst für den Informationsaustausch Sorge tragen muß. Für den Fall des Transfers zwischen einem Requirement-Tool und einem mehr designorientierten CASE bieten sich theoretisch drei Arten an:

- der Export der Anforderungsspezifikation vom Pflichtenheft zu einem CASE-Tool,
- der Import der (Entwurfs-)Spezifikation von einem CASE-Tool zum Pflichtenheft und
- die Synchronisation der Informationen, d.h. der ständige Austausch der Spezifikationen zwischen dem Pflichtenheft und einem CASE-Tool.

In den folgenden Ausführungen wird untersucht, welche der Verbindungen sinnvoll sind, d.h. welche der Verbindungsarten für die Softwaretechnik insgesamt benutzt werden sollten.

Der Export in das CASE-Tool

Die Softwareentwicklung ist ein Prozeß, der in verschiedene zeitlich getrennte Phasen aufgeteilt ist. Während der Definitionsphase wird das Pflichtenheft erstellt. Dieses bildet die Voraussetzung für die späteren Entwicklungsphasen.

Die definierten und formulierten Requirements können aus dem Pflichtenheft in ein Werkzeug übernommen werden. Lotus Notes stellt für die Gestaltung der Requirement-Dokumente nur begrenzte Möglichkeiten zur Verfügung. Die Dokumente können nur mit freiem Text oder eingefügten Objekten, z.B. Grafiken oder OLE-Objekten, gefüllt werden. Für den Export der Daten bedeutet es, daß entweder strukturierte Informationen oder Objekte in das CASE-Werkzeug übernommen werden. Das Werkzeug selbst kann die Requirements mit eigenen Methoden nach der Übernahme bearbeiten. Zum Beispiel können alle Funktionen des neuen Softwaresystems in übersichtlichen Funktionsbäumen dargestellt werden. Durch die Übernahme der Anforderungen aus dem Pflichtenheft in ein CASE-Tool wird eine Aufwandseinsparung erzielt. Die Requirements müssen nicht noch einmal im CASE-Tool separat eingetragen werden.

Jedes Requirement wird in der Lotus Notes-Anwendung als ein eigenes Dokument gewartet. Deshalb sollte für den Export der Requirements in das CASE-Tool ein Mechanismus vorhanden sein, der jedes Pflichtenheft-Objekt erkennt und semantisch kontrolliert in das Repository des CASE überträgt.

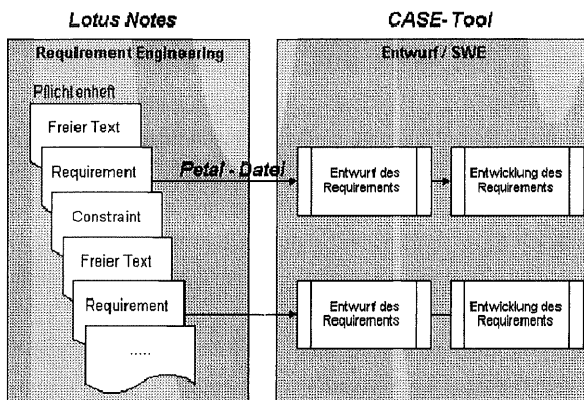


Abb. 8: Verbindung Requirement-Lösung und CASE-Tool

Der Mechanismus muß folgende Funktionalität aufweisen:

- alle Requirements werden mit ihren Attributen im Pflichtenheft gesammelt,
- die Liste der Requirements wird entweder direkt in das Repository des CASE-Tools geschrieben oder in einer Datenbasis zwischengespeichert,
- das CASE-Tool übernimmt die neuen Requirements aus der eigenen Datenbasis oder liest den Zwischenspeicher aus.

Nach der Übergabe können die Projektbeteiligten, z.B. die Entwurfs- oder die Entwicklungsabteilung, die Requirements für die nächsten Projektphasen als Grundlage nutzen.

Als Beispiel eines CASE fungierte in [17] die Softwareentwicklungs-Umgebung Rational Rose[9]. Diese CASE-Umgebung basiert auf der Definition von Klassen- und Objektstrukturen aus der Analyse sämtlicher informationeller und funktionaler Anforderungen. Alle ermittelten Anforderungen stellen die Grundlage für die Ermittlung der Klassen und Objekte dar. Nach der UML-Semantik stehen in Rational Rose Symbole zur Repräsentation der Anwendungsfälle(use cases), der Bedienungseinheiten(actors) und deren Beziehungen zur Verfügung. Es können Beziehungen zwischen Anwendungsfällen und Bedienungseinheiten sowie Beziehungen zwischen Anwendungsfällen untereinander dargestellt werden.

Rational Rose verwaltet die Projektdaten (Modelle) in externen Dateien mit der Dateierweiterung „.mdl“. Dazu gehören strukturierte Daten sowie Diagramme und

Ansichten. Die strukturierten Daten können von Rational Rose in sogenannten Petal-Dateien gespeichert werden. Alle Requirements für das zu entwickelnde Softwaresystem, z.B. die Klassendefinitionen oder Funktionen, werden im ASCII-Format in diesen Dateien gespeichert. Diese Petal-Dateien können von Rational Rose importiert und exportiert werden.

Das zu modellierende Softwaresystem wird in Subsysteme gekapselt, welche in Klassen modelliert werden können. Die Instanzen dieser Klasse besitzen Eigenschaften, Attribute, und sie können Aktivitäten, die Funktionen oder Operationen, ausführen.

Für die Datenübergabe aus der gruppenorientierten Requirement-Anwendung zum CASE muß der Exportmechanismus in die Notes-Applikation programmiert werden. Lotus Notes besitzt dafür die *Agents* als Designelemente. Die Agents sind Programme in der Lotus Notes-Anwendung, die manuell oder durch Scheduler vom Server gestartet werden können.

Der Agent heißt „*Export zu Rational Rose*“ und wurde mit der Programmiersprache LotusScript programmiert[17]. In der gegenwärtigen Ausbaustufe bearbeitet der Agent nur die Requirementstypen Klasse mit den Daten Operationen und Attribute (Abb. 10).

Typ	Datenelemente
Klasse	<ul style="list-style-type: none"> • Klassenname • Beschreibungsfeld
Operation	<ul style="list-style-type: none"> • Zugeordnete Klasse • Operationsname • Rückgabewert • Abarbeitungsart (Concurrency) • Sichtbarkeit (Export Control) • Parametername • Beschreibungsfeld
Attribut	<ul style="list-style-type: none"> • Zugeordnete Klasse • Attributsname • Attributtyp • Initialisierungswert • Beschreibungsfeld

Abb. 9: Exportierte Datenelemente in Rational Rose

Zusätzliche Datenelement-Erweiterungen sind jedoch ohne Probleme möglich. Die übergebenen Datenelemente finden sich dann in Rational Rose-Dialogen wieder und können dort bearbeitet werden..

Nach dem Export mit dem Notes-Agent kann die Petal-Datei von Rational Rose importiert werden. Dazu wird ein neues oder ein vorhandenes Modell geöffnet und die Petal-Datei über das Menü „File-Import“ ausgewählt. Die Daten werden aus der Datei in die Modellstruktur eingepaßt.

Weitere Typen an Requirements können ebenfalls problemlos in den Export integriert werden. Dazu müssen die Lotus Notes-Masken angepaßt und der Agent muß um weitere Exportfunktionen erweitert werden.

Der Import aus dem CASE-Tool

Der entgegengesetzte Weg, der Import der Entwurfsspezifikationen in das Pflichtenheft, kann für den Analyseprozeß ebenfalls von Bedeutung sein.

Zum einen kann der Bearbeitungszustand der Requirements im Stadium der Entwurfs- oder Implementationsphasen übernommen und im ursprünglichen Anforderungsdokument überprüft werden. Das Pflichtenheft dient somit als Checkliste für die Abarbeitung aller Requirements. Zum zweiten können, zur besseren Veranschaulichung die Diagramme oder andere Modelldarstellungen als grafische Bilder oder Einschübe in das Pflichtenheft übertragen werden. Der Informationsaustausch kann z.B. manuell durch das Kopieren der Grafiken über die Zwischenablage erfolgen. Die direkte Übernahme dieser Grafiken durch programmierte Tools ist in Lotus Notes nicht möglich. Die Programmierschnittstelle zu den Eingabefeldern in den Lotus Notes-Dokumenten erlaubt z. Zt. nur die Einfügung von rtf-Feldern mit grafischem Inhalt aber keine grafische Bearbeitung des Inhalts derartiger Dokumente. Auf Grundlage der Betrachtungen des zweiten Verbindungstyp, dem Import, ergibt sich auch für die Synchronisation eine geringe Bedeutung für den Austausch der Informationen.

Ein sinnvolle Verbindung vom CASE-Tool zum Pflichtenheft ist das Kopieren grafischer Darstellungen, die den Inhalt des Pflichtenheftes weiter verdeutlichen und für alle Beteiligten insbesondere die Nutzer verständlich sind. Beispiele dafür sind die Einbettung von Use Cases, Entity Relationship Diagrammen zur Aufgaben- und Informationsmodellierung, Funktionshierarchiediagramme usw. Da der Lotus Notes-Client nur einen Editor zur Texteingabe bereitstellt, ist dieser Import gegenwärtig nur über das Kopieren und Einfügen mittels der Zwischenablage möglich (Abb. 10).

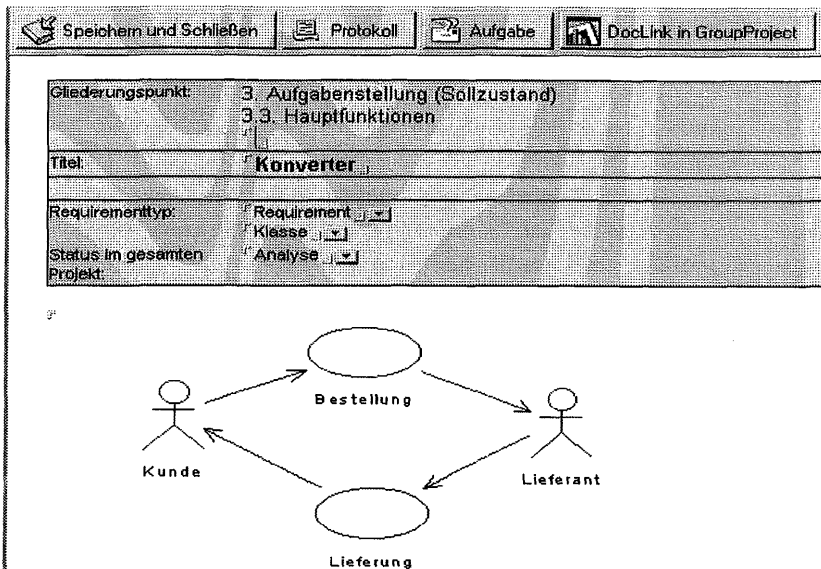


Abb. 10: Diagramm aus Rational Rose in einer Sektion des Pflichtenheftes

Allerdings muß die Größe der Diagramme beachtet werden. Die Dokumente werden in Lotus Notes bei weitgefächerten Diagrammen unübersichtlich.

5 Schlußfolgerungen und Weiterentwicklung

In diesem Artikel wurde ausgehend von einer Analyse der bewährten Techniken des Requirement Engineering ein allgemeines Modell für Anforderungsdokumente entworfen. Ein weiterer Schwerpunkt lag auf der Betrachtung des gruppenorientierten Requirement Engineering. Es wurden sowohl Merkmale als auch Anforderungen herausgearbeitet, die gemeinsam die Grundlage für die Entwicklung einer gruppenorientierten Lösung darstellten. Als die geeignetste Plattform für die Anwendung stellte sich das Groupware-System Lotus Notes heraus. Die fertiggestellte Lösung ist so gestaltet, daß die Ansichten und Dokumente im Internet lesbar sind und somit eine Zusammenarbeit unabhängig von Lotus Notes gewährleistet ist. Zusätzlich wurde experimentell eine Verbindung zwischen der Lotus Notes Applikation und dem CASE-Tool Rational Rose geschaffen.

Die hier vorgestellte Lösung ist ein erster Ansatz zur Unterstützung des gruppenorientierten Requirement Engineering. Das nächste Ziel wäre die Erprobung im Rahmen eines lohnenden Projektes der Praxis. Die Vorstellung unserer Lösung auf diesem Workshop soll dazu beitragen, einen interessierten Industriepartner zu finden.

Eine Weiterentwicklung sollte insbesondere unter dem Gesichtspunkt der Verfolgung und Erfassung der Erfüllung der Requirements insbesondere in den nachgelagerten Phasen Entwurf, Implementierung und Test durchgeführt werden. Falls dazu andere CASE, Software- oder Programmierumgebungen eingesetzt werden, ist der Datenaustausch möglichst auf der Grundlage eines standardisierten Formats (z.B. XML) weiter zu entwickeln. Zur Überprüfung der Erfüllungsstände sind die zugehörigen Zustandswerte dem Requirement-Tool zu übermitteln, damit dort zentral die Kontrolle aller signifikanten Anforderungen durchgeführt werden kann.

Die große Anpassungsfähigkeit der vorgestellten Lösung ist weiterhin im Hinblick auf die Nutzung verschiedener vorgegebener Vorgehensmodelle zu untersuchen. Der Schwerpunkt liegt hier weniger auf der Dokumentenstruktur, sondern vielmehr in der Anwendbarkeit für unterschiedliche Aktivitäts- und Prozeßstrukturen. Auf Grund der Allgemeingültigkeit der vorgestellten Lösung werden auf diesem Gebiet aber keine gravierenden Probleme erwartet. Schließlich sind weitere CASE-Tools zu analysieren, um den Datenaustausch zwischen den Entwicklungs-Datenbasen effektiver gestalten zu können. Hier wird die Erwartung geäußert, daß die weitere Standardisierung der Austauschformate [21] diesen Prozeß endlich voranbringen möge.

Literatur

- [1] Dröschel, W., Heuser, W., Midderhoff, R.: Inkrementelle und objektorientierte Vorgehensweisen mit dem V-Modell 97; Oldenbourg Verlag 1997
- [2] VDI/VDE 3694: Lastenheft/Pflichtenheft für den Einsatz von Automatisierungssystemen; VDI/VDE; April 1991
- [3] Thayer, R. H., Dorfmann, M.: Software Requirements Engineering; IEEE Computer Society Press 1998
- [4] Balzert, H.(Hrsg.): CASE-Systeme und Werkzeuge; BI-Wissenschaftsverlag (5. Auflage) Mannheim 1993
- [5] Dröschel, W.: CASE Tools–Werkzeugunterstützung im Rahmen des V-Modells; Oldenbourg Verlag 1995
- [6] Davis, A. M.: Software Requirements – Objects, Functions, and States; PTR Prentice Hall 1993
- [7] Rational Software Corporation: Unified Modeling Language; UML Specification (Draft), Version 1.3
- [8] Oberg, G., Probasco, L., Ericsson, M.: Applying Requirements Management with Use Cases; Rational Software Corporation 1998
- [9] Rational Software Corporation: Using Rose and Clear Case in a Team Environment;
<http://www.rational.com/products/rose/prodinfo/whitepapers/dynamic.jtмл>
- [10] Rational Software Corporation: RequisitePro;
<http://www.rational.com/products/repro/>
- [11] Lauber, R.: Integrated Project Support Environments; Encyclopedia of Computer Science and Technology (Kent A. and S.G. Williams, Eds.) Vol. 20 Supplement 5, New York: Marcel Dekker , 1989 pp. 262-286
- [12] Oberweis, A.: Modellierung und Ausführung von Workflows mit Petri-Netzen; Teubner Verlag 1996, S. 202ff
- [13] Jacobson, I., Booch, G., Rumbaugh, J.: Unified Software Development Process; Addison-Wesley Verlag 1999
- [14] Teufel, S., Sauter, Ch., Mühlherr, Th., Bauknecht, K.: Computerunterstützung für die Gruppenarbeit; Addison Wesley 1995
- [15] Bair, J. H.: Supporting Cooperative Work With Computers: Addressing Meeting Mania; in: Proc. 34th IEEE Computer Society International Conference
- [16] Petrovic, O.: Workgroup Computing – Computergestützte Teamarbeit; Physica Verlag 1993
- [17] Stelter, H.: Gruppenorientiertes Requirement Engineering; Diplomarbeit TU Dresden/Fakultät Informatik 1999

- [18] Nastansky, L.: Workgroup Computing, Computergestützte Gruppenarbeit in der Praxis; Betriebswirtschaft aktuell Bd. 12, Steuer- und Wirtschaftsverlag Hamburg 1993
- [19] ULC Groupware Consulting GmbH: Lotus Notes Domino und Java; IBM Lehrgangsunterlagen, Lehrgangstyp 78L59 Dresden 1998
- [20] Lotus Development Corporation: Lotus Notes - Application Developer's Guide; Lotus Development Corporation, Cambridge 1995
- [21] Jeckle, M.: XML basierter Metadatenaustausch; Vortrag auf dem 6.Fachgruppentreffen der GI-FG 2.1.9 am 27.01.99 in München http://www.krumbach.de/home/jeckle/files/xmi_mdc.pdf