

Reihe: Telekommunikation @ Mediendienste · Band 14

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, Dr. Rudolf Pospischil, Brüssel, Prof. Dr. Claudia Löbbecke, Köln, und Prof. Dr. Christoph Zacharias, Köln

PD Dr.-Ing. habil. Martin Engelien
Dipl.-Inf. Jens Homann (Hrsg.)

Virtuelle Organisation und Neue Medien 2002

Workshop GeNeMe2002
Gemeinschaften in Neuen Medien

TU Dresden, 26. und 27. September 2002



JOSEF EUL VERLAG
Lohmar · Köln

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Virtuelle Organisation und Neue Medien 2002 / Workshop GeNeMe 2002 – Gemeinschaften in Neuen Medien – TU Dresden, 26. und 27. September 2002. Hrsg.: Martin Engeliens ; Jens Homann. – Lohmar ; Köln : Eul, 2002

(Reihe: Telekommunikation und Medienwirtschaft ; Bd. 14)

ISBN 3-89936-007-9

© 2002

Josef Eul Verlag GmbH

Brandsberg 6

53797 Lohmar

Tel.: 0 22 05 / 90 10 6-6

Fax: 0 22 05 / 90 10 6-88

<http://www.eul-verlag.de>

info@eul-verlag.de

Alle Rechte vorbehalten

Printed in Germany

Druck: RSP Köln

Bei der Herstellung unserer Bücher möchten wir die Umwelt schonen. Dieses Buch ist daher auf säurefreiem, 100% chlorfrei gebleichtem, alterungsbeständigem Papier nach DIN 6738 gedruckt.



Technische Universität Dresden
Fakultät Informatik • Institut für Angewandte Informatik
Privat-Dozentur Angewandte Informatik

PD Dr.–Ing. habil. Martin Engelen

Dipl.–Inf. Jens Homann

(Hrsg.)

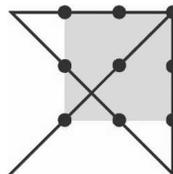


an der

Fakultät Informatik der Technischen Universität Dresden

in Zusammenarbeit mit der
Gesellschaft für Informatik e.V.,
GI-Regionalgruppe Dresden

gefördert von der Klaus Tschira Stiftung
gemeinnützige Gesellschaft mit beschränkter Haftung



am 26. und 27. September 2002

in Dresden

<http://pdai.inf.tu-dresden.de/geneme>

Kontakt: Thomas Müller (geneme@pdai.inf.tu-dresden.de)

D.3. Eine Rahmenanwendung für die informelle Teamarbeit in Dokumentenbeständen

Alexander Lorz

Heinz-Nixdorf-Stiftungslehrstuhl für Multimediatechnik

Technische Universität Dresden

Zusammenfassung: Die Zusammenarbeit an gemeinsam genutzten Dokumenten ist ein zentraler Aspekt virtueller Gemeinschaften und verteilt arbeitenden Teams. In diesem Beitrag werden die Konzepte und der Prototyp einer Rahmenanwendung vorgestellt, welche die informelle Zusammenarbeit in den Dokumentenbeständen virtueller Gemeinschaften unterstützt. Die Kernidee der vorgestellten Lösung liegt in der Einführung einer Dokumenten-Metapher in Verbindung mit einem feingranularen Rechtesystem, einem Konzept zur Strukturierung des Dokumentenraums und einer Strategie zur Verteilung von Dokumenten zwischen mehreren Speicherorten und zur Gewährleistung der Verfügbarkeit von Dokumenten bei nicht-permanenter Netzwerkanbindung. Auf der Grundlage dieser Konzepte werden Workflows auf spezielle Dokumente abgebildet, die durch die gleichen Mechanismen wie die eigentlichen Inhalte und Dokumente gespeichert, verwaltet und zwischen den einzelnen Nutzern verteilt werden.

1. Einleitung

Die Speicherung, der Austausch und die gemeinsame Nutzung von Dokumenten und der in ihnen abgelegten Informationen bilden die Grundlage von Zusammenarbeitsvorgängen in virtuellen Gemeinschaften und Unternehmen. Daher sind elektronische Werkzeuge zur Unterstützung der mit diesen Vorgängen verbundenen Arbeitsabläufe und Kommunikationsprozesse von grundlegender Bedeutung für den Erfolg virtueller Gemeinschaften. An solche Werkzeuge existieren viele, einander teilweise widersprechende Anforderungen. Diese haben letztendlich zu einer großen Palette unterschiedlichster Produkte und Lösungsansätze geführt, die von spezialisierten Nischen-Applikationen über generische Middleware-Lösungen bis hin zu großen, auf verschiedene Einsatzzwecke anpassbaren Universalsystemen reicht.

In dem vorliegenden Beitrag sollen die Konzepte und der Prototyp des „Enabling Informal Teamwork Medienmanagementsystems“ (EITMMS) dargestellt werden. Das EITMMS ist eine Rahmenanwendung, die eine virtuelle Gruppe verteilt arbeitender Personen beim Austausch von Dokumenten und bei der Koordinierung ihrer

Arbeitsvorgänge über das Internet unterstützt. Die Konzepte und der Prototyp entstanden in Zusammenarbeit mit dem Cornelsen Verlag im Rahmen des Projekts „Enabling Informal Teamwork“ (EIT) innerhalb der vom BMBF geförderten Initiative „Globale Elektronische und Multimediale Informationssysteme“ (Global Info).

Bei der Entwicklung des EITMMS stand der Gedanke im Vordergrund, ein elektronisches Werkzeug zu konzipieren, welches als Lösung „off-the-shelf“ sofort eingesetzt werden kann. Verteilt arbeitende Gruppen und virtuelle Gemeinschaften können mit seiner Hilfe ohne größeren initialen Aufwand gemeinsam Daten nutzen, miteinander kommunizieren und Arbeitsprozesse koordinieren. Darüber hinaus ist es möglich, durch eine übersichtliche und leicht zu handhabende Programmierschnittstelle das EITMMS modular zu erweitern, um auf spezielle Bedürfnisse und Anforderungen des jeweiligen Nutzerkreises zu reagieren. Auf diese Weise entstehen gewachsene Lösungen, die eine bessere Unterstützung für den jeweiligen Anwendungsfall bieten als Standardanwendungen und mit geringerem Aufwand realisierbar sind als vollständig maßgeschneiderte Spezialanwendungen.

Im folgenden Abschnitt werden die Zusammenarbeits- und Kommunikationsvorgänge in virtuellen Teams betrachtet und Anforderungen an Werkzeuge dargestellt. Exemplarisch steht dabei der Prozess der Erstellung multimedialer elektronischer Publikationen, insbesondere im Bereich der elektronischen Lehr- und Lernmaterialien im Vordergrund.

Danach wird auf ausgewählte Anwendungen und Forschungsansätze eingegangen, wobei auf Grund der großen Vielfalt nur einige, für das Thema besonders relevante Lösungen behandelt werden. Im darauf folgenden Abschnitt wird das Konzept des EIT-Medienmanagementsystems vorgestellt, wobei zunächst auf die zugrundeliegenden Ideen und Metaphern und anschließend auf die Anwendungssicht des EITMMS eingegangen wird. Abschließend werden die gewonnenen Erfahrungen zusammengefasst und es wird auf weitere Forschungsziele eingegangen.

2. Informelle Teamarbeit in Dokumentenbeständen

Ein Ziel des Projektes EIT war die Betrachtung von Zusammenarbeitsprozessen bei der Erstellung elektronischer multimedialer Publikationen im Bereich der Lehr- und Lernanwendungen. Aus diesem Problemkontext heraus sollen im Folgenden sowohl konkrete Anforderungen an eine geeignete informationstechnische Werkzeugumgebung dargestellt als auch über den speziellen Anwendungsbereich hinausgehende

Anforderungen an Anwendungen für die Erstellung und Bearbeitung von elektronischen Dokumenten in kleinen und mittleren Teams formuliert werden.

Im universitären Umfeld erfolgt die Bereitstellung von Inhalten für Lehr- und Lernanwendungen oder andere elektronische Publikationen meist durch eine Person oder einen sehr kleinen Kreis fest angestellter Mitarbeiter und Hochschullehrer. Diese organisieren in ihrer Rolle als Projekt-Koordinator auch die Umsetzung der Inhalte in das Präsentationsmedium. Dabei wird auf die Unterstützung durch nicht festangestellte Mitwirkende, z.B. studentische Hilfskräfte und freiberufliche Designer, zurückgegriffen. Diese freien Mitarbeiter werden entsprechend den anfallenden Arbeiten und der vorhandenen Qualifikation flexibel für unterschiedliche Tätigkeiten eingesetzt.

In Verlagen übernimmt ein Redakteur die Koordination unterschiedlicher Experten und kontrolliert die Qualität der Zwischen- und Endprodukte. Inhalte werden von meist externen Autoren zugeliefert und können in den unterschiedlichsten Formen, vom handgeschriebenen Manuskript bis zu druckfertig formatierten Textdokumenten, vorliegen. Der Redakteur sichtet und ordnet diese Rohfassung der Inhalte und leitet sie zur weiteren Bearbeitung an z.B. Grafiker, Wortredakteure, Programmierer und Web-Designer weiter. Diese ergänzen das vorhandene Material und bringen es in ein präsentationsfähiges Format, z.B. Word- und PDF-Dokumente, die dem Endkunden zum Download angeboten werden. Andere gängige Vertriebswege sind interaktive multimediale Präsentationen zur Verbreitung per Internet oder CD-ROM, wobei für letztere oft noch ergänzende Materialien über das Internet bereitgestellt werden. Dabei erfolgt eine fortlaufende Qualitätskontrolle durch den Redakteur.

Der Austausch von Zwischenergebnissen und die Koordination der Zusammenarbeit erfolgt i.d.R. per E-Mail und über gemeinsam genutzte Verzeichnisse. Besonders nachteilig an dieser Vorgehensweise ist, dass Zwischenergebnisse immer wieder neu in den Gesamtkontext eingeordnet werden müssen. Die bei der Bearbeitung implizit anfallenden Metadaten, z.B. über einzelne Bearbeiter, deren Auslastung, den Fortschritt von Teilaufgaben etc., können nicht genutzt werden, da sie mit den Arbeitsobjekten nicht verknüpft sind und beim Transport von Dokumenten zwischen den Mitarbeitern verloren gehen. Es entsteht ein nicht zu unterschätzender Aufwand für die koordinierende Person, die den Gesamtkontext des Projekts verwaltet. Es muss insbesondere darauf geachtet werden, dass erreichte Zwischenschritte nicht versehentlich überschrieben werden und Teilergebnisse bzw. Projektstände nachträglich wieder rekonstruiert werden können.

In der geschilderten Anwendungsdomäne treten zwei Problemstellungen besonders deutlich zu Tage:

Dokumentenmanagement-Aspekte: Wie wird der dezentrale Zugriff auf einen gemeinsam benutzten Datenbestand durch verschiedene interne und externe Mitarbeiter mit teilweise sehr heterogener technischer Infrastruktur gewährleistet?

Workflow-Aspekte: Wie kann der organisatorische Aufwand für die Koordination der einzelnen Mitarbeiter und die Kommunikation mit ihnen minimiert werden?

Im Folgenden soll auf diese beiden Aspekte ausführlicher eingegangen werden. Eine stichpunktartige Zusammenfassung der resultierenden Anforderungen erfolgt in Abschnitt 2.3.

2.1 Dokumentenmanagement-Aspekte

In virtuellen Gemeinschaften und verteilt arbeitenden Teams bilden gemeinsam genutzte Dokumente einen wesentlichen Kernpunkt der Zusammenarbeit. Sie sind sowohl Gegenstand der gemeinsamen Arbeit als auch Mittel zur Kommunikation und bilden das „gemeinsame Gedächtnis“ der Gruppe. Ein nicht zu unterschätzender Aspekt ist die informelle Kommunikation an Arbeitsartefakten, da aus diesen der momentane Arbeitsbereich und die Intentionen einzelner Gruppenmitglieder abgeleitet werden können. Damit stellen Dokumente und die Meta-Informationen über ihre Bearbeitung ein wichtiges Medium zur Schaffung von Group-Awareness dar. Diese wiederum ist zur Vermeidung und Auflösung von Konflikten in synchronen Kooperationsszenarien unentbehrlich.

In den beschriebenen Szenarien werden Dokumente primär in Form von Dateien gehandhabt. Dieser Ansatz ist in der Historie der Informationstechnologie verwurzelt, da Dateien die essenzielle Metapher zum Umgang mit digital gespeicherten Informationen darstellen. Ein wesentlicher Nachteil dieses Ansatzes besteht in der Trennung von Nutzdaten bzw. Inhalten und Meta-Informationen, die Aussagen über die Inhalte treffen. Es ist daher eine wichtige Aufgabe von Dokumentenmanagement-Systemen, Daten und Meta-Daten zusammen zu halten, Meta-Daten automatisch zu erfassen und deren Integrität sicher zu stellen.

Dateien werden durch ihren Speicherort, d.h. durch die Angabe eines physikalischen oder logischen Laufwerks in Verbindung mit einer Pfadangabe und einem Dateinamen eindeutig identifiziert. In verteilten Szenarien ist das ein Nachteil, da z.B. für externe Mitarbeiter oder bei der Benutzung mobiler Geräte nicht jederzeit Zugang zum Speicherort einer Ressource besteht. Daher ist es eine verbreitete Vorgehensweise, mit lokalen Kopien dieser Dateien zu arbeiten. Allerdings führt die manuelle Verwaltung dieser Kopien beinahe zwangsläufig zu Konflikten und zum Verlust von Arbeitsergebnissen durch das Überschreiben von zwischenzeitlich geänderten „Originalen“ durch zeitgleich von einem anderen Mitarbeiter bearbeitete Kopien. Um Gruppenarbeit zu unterstützen, muss ein Werkzeug Mechanismen anbieten, um Daten von ihrem Speicherort zu lösen und Replikate zu verwalten. Konflikte sind entweder durch z.B. Sperrmechanismen zu vermeiden oder durch geeignete Strategien automatisch aufzulösen.

In Dateisystemen wird die Gesamtmenge aller vorhandenen Dokumente, der Dokumentenraum, durch Verzeichnisse strukturiert. Damit wird allen Nutzern zwangsläufig die gleiche hierarchische Sicht auf den Gesamtdatenbestand aufgezwungen, obwohl eine andere Strukturierung aus Sicht einzelner Mitarbeiter durchaus sinnvoll wäre und die Arbeit vereinfachen würde. In Verbindung mit der Löslösung der Dokumente von ihrem Speicherort ist es an dieser Stelle sinnvoll, individuelle Sichten auf den Dokumentenraum anzubieten. Diese sollen es den einzelnen Team-Mitgliedern gestatten, sich eine eigene Ordnungshierarchie aufzubauen und die von ihnen benutzten Dokumente nach eigenen Gesichtspunkten zu gliedern, ohne jedoch anderen ihre Ordnungskriterien aufzuzwingen.

Werden Dokumente zusammen mit den zugehörigen Metadaten gespeichert und verwaltet, ist es notwendig, ein Berechtigungskonzept für den Zugriff auf beiden Ebenen anzubieten. Der triviale Ansatz, identische Zugriffsrechte auf Daten und alle Metadaten zu vergeben, ist nicht immer ausreichend. Insbesondere dann, wenn personalisierte Sichten auf Dokumente durch benutzerabhängig sichtbare Metadaten benötigt werden oder Metadaten zur Steuerung von Arbeitsabläufen eingesetzt werden sollen, ist ein feingranulares Berechtigungskonzept unumgänglich. Dieses Konzept sollte die Vergabe von Berechtigungen sowohl für das Gesamtdokument als auch für einzelne Metadaten gestatten.

2.2 Workflow-Aspekte

An den geschilderten, vergleichsweise einfachen Szenarien ist bereits zu erkennen, dass sich bei der Erstellung multimedialer Publikationen noch keine klaren, universell anwendbaren Zusammenarbeitsmuster herauskristallisiert haben. Der Erstellungsprozess komplexer multimedialer Anwendungen ist durch die Verwendung einer Vielzahl neuer, sich stetig entwickelnder Technologien und demzufolge durch den flexiblen und bedarfsorientierten Einsatz von Spezialisten geprägt. Teilweise sind die Teamstruktur und die Rollen von Teammitgliedern fortlaufend Veränderungen unterworfen. Unter diesen Voraussetzungen eine umfassende und detaillierte formale Beschreibung von Zusammenarbeitsmustern vorzunehmen, ist mit hohem Aufwand verbunden. Darüber hinaus sind Gültigkeitsdauer und Verwendbarkeit eines solchen Modells in starkem Maß an die konkreten Publikationsszenarien und -technologien gebunden.

In [1] wird der Erstellungsprozess von web-basierten Lerneinheiten, also einem Teilbereich der beschriebenen Anwendungsdomäne, betrachtet. Die Autoren kommen zu dem Schluss, dass einfache sequentielle Modelle die Arbeitsabläufe nur unzureichend beschreiben können. Es wird statt dessen ein Prozesskonzept unter Anwendung der ISO 13470 („Human-centred design processes for interactive systems“) und des iterativen Spiralmodells der Softwareentwicklung vorgeschlagen. Dieses besteht im Wesentlichen aus einer Abfolge der Phasen Analyse, Entwurf, Implementierung und Einsatz, die jeweils für sich in einen Zyklus mit Review- und Evaluierungsfunktion eingebunden sind. Zwischen den einzelnen Phasen bzw. Zyklen sind Übergänge vorhanden, die bei eventuell auftretenden Schwierigkeiten ein Zurückgehen in eine vorhergehende Phase ermöglichen.

Geht man von der Prämisse aus, dass sich die Zusammenarbeit in realen und virtuellen Teams nur durch vergleichsweise komplexe Modelle umfassend beschreiben lässt, impliziert dies einen erheblichen Aufwand für die Entwicklung und Wartung von Anwendungen zur vollständigen Unterstützung von Arbeitsabläufen.

Eine sinnvolle Alternative stellt hier die Unterstützung kleinerer, generisch verwendbarer und in sich geschlossenerer Teil-Workflows mit geringer Komplexität dar. Diese beschreiben zwar nicht den vollständigen Produktionsablauf, bieten aber eine Strukturierung für Teilprozesse, wie z.B. Auftrags-Review-Zyklen, und entlasten den Projekt-Koordinator.

2.3 Anforderungen an Werkzeuge

Im Folgenden sollen die aus Sicht der Anwendungsdomäne wesentlichen Anforderungen an Werkzeuge zur Unterstützung von dokumentenzentrierter Zusammenarbeit in virtuellen Gemeinschaften und verteilt arbeitenden Teams kurz zusammengefasst werden. Im Vordergrund steht dabei die Unterstützung informaler, nicht durchgehend strukturierter Zusammenarbeit, wie sie insbesondere bei sich dynamisch entwickelnden Gruppen anzutreffen ist. Wichtige Anforderungen sind:

- Loslösung digitaler Ressourcen von ihrem Speicherort
- Verwaltung von Replikaten zur Gewährleistung der Verfügbarkeit bei nicht-permanenter Netzwerkanbindung
- Integration von Daten und Metadaten
- Konzepte zur individuellen Strukturierung des Dokumentenraums
- personalisierte Sichten auf Dokumente
- feingranulare Berechtigungsmechanismen auf Metadaten-Ebene
- Auflösung oder Vermeidung von Konflikten bei synchroner Zusammenarbeit
- Schaffung von Group-Awareness durch Erfassung und Visualisierung von Meta-Informationen über Arbeitsprozesse
- Unterstützung von Kommunikationsprozessen und Workflows
- Erweiterbarkeit zur Anpassung an gruppenspezifische Arbeitsabläufe

Darüber hinaus existieren natürlich noch weitere Anforderungen, die jedoch eher genereller Natur und nicht typisch für das geschilderte Anwendungsfeld sind. Daher sollen an dieser Stelle nur die wichtigsten genannt werden:

- Schnittstellen zu Standardanwendungen
- Einsetzbarkeit als Lösung „off-the-shelf“, ohne größeren Konfigurations- und Anpassungsaufwand
- vertraute oder leicht erlernbare Metaphern für die Benutzersicht

3. Existierende Anwendungen und Forschungsansätze

Zur Unterstützung verteilt arbeitender Gruppen existiert ein breites Spektrum kommerzieller und freier Anwendungen, die vorwiegend in die Bereiche Workflow-, Content-, Dokumenten- und Knowledge-Management-Systeme eingeordnet werden können.

Verbreitete kommerzielle Produkte sind u.a. *IBM Lotus Domino.Doc* [2], *Novell GroupWise* [3], *Vignette V6 Content Suite* [4], *Gauss VIP Content Manager* [5] und das *NPS (Network Productivity System)* [6], um nur einige zu nennen. Die genannten Systeme besitzen einen sehr großen Funktionsumfang und stellen eine Vielzahl von Werkzeugen zur Unterstützung von Gruppenarbeit bereit, wobei unterschiedliche Schwerpunkte bezüglich des Einsatzgebietes existieren. Die ersten beiden Produkte werden primär zur Unterstützung formalisierter Arbeitsabläufe, der Verwaltung von Dokumenten und Unterstützung von Teams durch z.B. Benachrichtigungsmechanismen und gemeinsame Terminkalender eingesetzt. Die letztgenannten Produkte sind auf das Web-Contentmanagement spezialisiert und dienen hauptsächlich der Erstellung und Pflege großer Web-Sites. Ein wesentlicher Nachteil dieser kommerziellen Systeme besteht in den hohen Kosten für Anschaffung und Betrieb sowie dem nicht unerheblichen Aufwand zur Anpassung an kundenspezifische Anforderungen.

Groove [7] versteht sich selbst als „Desktop Collaboration Software“ und hebt sich von den oben genannten Anwendungen dadurch ab, dass es ohne zentrale Serverkomponenten als Peer-to-Peer-Lösung eingesetzt werden kann. Anwender können mittels sogenannter „Shared Spaces“ Dokumente austauschen und gemeinsam nutzen. In die Arbeitsoberfläche sind standardmäßig verschieden synchrone und asynchrone Kommunikationswerkzeuge integriert. Eine inhaltsorientierte Verwaltung von Dokumenten und die Strukturierung der gemeinsam benutzten Dokumente mit Metainformationen ist allerdings nicht vorgesehen.

Neben den kommerziellen Anwendungen enthalten Open-Source-Lösungen, wie z.B. *Zope* [8], *OpenCMS* [9] und *Midgard* [10], weitere interessante Ansätze. Vorteile dieser Lösungen sind die freie Verfügbarkeit und die guten Erweiterungsmöglichkeiten durch den vorhandenen Quellcode. Nachteilig sind die oft fehlende Unterstützung und die meist schwierige Konfiguration.

Ein umfassenderer Überblick über existierende Anwendungen auf dem Gebiet der Content- und Workflow-Management-Systeme ist u.a. in [11], [12] und [13] zu finden.

Concurrent Version Systems (CVS) [14] ist keine Dokumentenmanagement-Anwendung im eigentlichen Sinne, sondern stammt aus dem Bereich des Software-Engineering und dient zur gemeinsamen Verwaltung von Source-Code für größere Software-Projekte. Die Funktionsweise von CVS gestattet es jedoch, dieses Werkzeug auch zur Entwicklung und Wartung web-basierter Anwendungen und multimedialer Präsentationen einzusetzen. Die besondere Leistungsfähigkeit von CVS liegt in der Verwaltung von Versionen und Versionszweigen für aus vielen einzelnen Dateien und Verzeichnissen bestehenden Projekten in einem sogenannten Repository. Verzeichnisbäume und Dateien können ganz oder teilweise auf lokale Datenträger repliziert werden. CVS visualisiert Änderungen, die lokal erfolgt sind bzw. durch andere Benutzer in das Repository eingespielt wurden und übernimmt die Synchronisation der Replikat mit dem Repository. Konflikte können sowohl durch Sperren vermieden als auch durch das „Mischen“ (merge) von zwei parallel entstanden Versionen einer Datei automatisch aufgelöst werden. Dies ist allerdings nur eingeschränkt möglich und kann manuelle Korrekturen erforderlich machen. Nachteilig ist die vergleichsweise komplexe Benutzerführung, die ein Verständnis der Arbeitsweise des CVS voraussetzt und für fachfremde Nutzer einige Hürden bereithält.

Basic Support for Cooperative Work (BSCW) [15] ist eine web-basierte Softwarelösung zur Unterstützung der kooperativen Arbeit auf einem gemeinsamen Ressourcenbestand. Auf einem Server können über ein Web-Interface oder eine Java-Anwendung Verzeichnisstrukturen eingerichtet und mit anderen Nutzern gemeinsam zur Ablage von Dateien genutzt werden. Durch Ereignisinformationen erhält der Nutzer ein Gefühl für die Aktivitäten anderer. Für die Bearbeitung eines Dokuments ist es notwendig, dieses explizit für andere zu sperren, es auf die lokale Festplatte herunterzuladen und nach Abschluss der Bearbeitung wieder in das BSCW-System einzustellen.

TOWER (Theater of Work Enabling Relationship) [16] gestattet die Visualisierung der Aktionen von BSCW-Nutzern in einer 3D-Welt durch symbolische Handlungen und durch Gesten von Avataren. Dadurch sollen Group-Awareness und die Möglichkeit zur informellen Begegnung zwischen verteilt arbeitenden Nutzern geschaffen werden.

NESSIE [17] stellt ebenfalls Awareness-Funktionen für Gruppenarbeit mittels einer 3D-Umgebung bereit, visualisiert jedoch zusätzlich Informationen, die durch Sensoren wie z.B. Lichtschranken und Geräuschpegelmessern in der Büroumgebung des Anwenders gewonnen werden. Der Aufwand für die notwendige technische Infrastruktur und datenschutzrechtliche Aspekte dürften einer breiten Anwendung jedoch im Wege stehen.

Aus dem Xerox Palo Alto Research Center stammen die Projekte *Placeless Documents* [18] und *Harland* [19], deren Ziel es ist, Dokumente anhand ihrer Eigenschaften anstatt ihres Speicherorts zu verwalten. Zur Beschreibung von Dokument-Eigenschaften werden Properties in Form von Name-Wert-Paaren verwendet. Harland unterstützt dabei die Typung von Dokumenten anhand eines Schemas. Berechtigungsmechanismen auf Property-Ebene werden allerdings nicht bereitgestellt. Auf der Grundlage von *Placeless Documents* wurden Ansätze zur Unterstützung von Workflows und dokumentenzentrierter Zusammenarbeit realisiert [20].

Über das betrachtete Anwendungsfeld hinaus ist das Projekt *Science-To-Science (S2S)* [21] von Interesse, das die Entwicklung und den Betrieb eines JXTA-Search konformen Peer-To-Peer-Netzwerks zur Suche und zum Austausch von wissenschaftlichen Dokumenten zum Ziel hat. Dieses Projekt ist insofern für die Unterstützung von verteilten Teams und virtuellen Gemeinschaften von Relevanz, als dass es die verteilte Publikation und den Austausch von Dokumenten im Verbund mit Hintergrundinformationen und Anmerkungen unterstützt und die notwendigen Suchmechanismen bereitstellen soll.

Obwohl existierende Lösungen die unter 2.3 genannten Anforderungen zum Teil erfüllen, gibt es kein System, welches allen genannten Kriterien gerecht werden kann. Im Folgenden wird daher ein Konzept vorgestellt, welches speziell unter dem Gesichtspunkt der informellen Zusammenarbeit in den Dokumentenbeständen virtueller Teams entworfen wurde.

4. Das EIT-Dokumentenmanagementsystem

Die Kernidee des hier vertretenen Lösungsansatzes liegt in der Einführung einer Dokumenten-Metapher in Verbindung mit einem feingranularen Rechtesystem, einem Konzept zur Strukturierung des Dokumentenraums und einer Strategie zur Verteilung von Dokumenten zwischen mehreren Speicherorten. Zudem wird die Verfügbarkeit von Dokumenten bei nicht-permanenter Netzwerkanbindung gewährleistet. Auf dieser Grundlage werden Workflows auf spezielle Dokumente abgebildet, die durch die gleichen Mechanismen wie die eigentlichen Inhalte und Dokumente gespeichert, verwaltet und zwischen den einzelnen Mitarbeitern verteilt werden.

4.1 Dokumente und Properties

Ein Dokument ist ein Container für Informationen zur Speicherung und Beschreibung einer beliebigen digitalen Ressource und besteht aus einem Konglomerat von Daten und Metainformationen. Der Zugriff auf Daten und Metainformationen erfolgt über eine einheitliche Metapher, die sogenannten Properties. D.h. jedes Dokument besteht aus einer Anzahl einzelner Properties, die sich wiederum aus den Elementen Name, Typ, Wert, Besitzer und Rechte zusammensetzen (siehe Abb. 1).

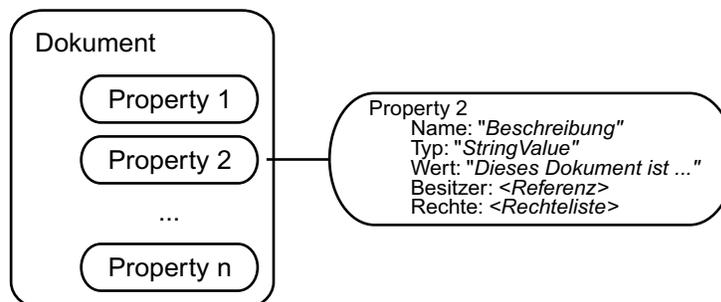


Abb. 1: Aufbau eines Dokuments

Ein Text-Dokument besteht beispielsweise aus den eigentlichen Nutzinformationen, d.h. einer Inhaltsobjekt-Property, in der z.B. eine Word- oder StarOffice-Datei abgelegt ist, sowie einer Reihe weiterer Properties, die beschreibenden Charakter besitzen oder zur Verwaltung des Dokuments im System notwendig sind. Hier sind das u.a. der Name des Textdokuments, der Typ des Binärobjekts, ein systemweit eindeutiger Identifikator, Informationen zu Speicherstandorten und eine Property, welche die Zugriffsrechte einzelner Nutzer beschreibt. Unter Einhaltung bestimmter Randbedingungen, die von der Dokumenttypdefinition (siehe Abschnitt 4.2) vorgegeben sind, können dem Dokument weitere Properties hinzugefügt werden. Daten und Metadaten werden auf einer gleichberechtigten Ebene untrennbar miteinander zu einem Dokument verknüpft.

Um eine flache, einheitliche Metapher der Repräsentation von Informationen zu realisieren, werden im EIT-System sämtliche – auch die systemspezifischen Informationen – durch Dokumente abgebildet. So erfolgt die Beschreibung von Benutzerkonten durch spezielle Dokumente, deren Properties, wie z.B. der aktuelle Online-Status des Benutzers, durch die Rahmenanwendung automatisch aktualisiert werden. Auch wird der Speicherort eines Dokuments durch eine Property beschrieben, wobei eine Änderung dieses Werts durch den Benutzer das System zur Verlagerung des Dokuments auf einen anderen Rechner veranlasst.

Die Tatsache, dass es Properties gibt, die im eigentlichen Sinne keine persistent zu speichernden Informationen darstellen, sondern dynamisch aus dem Kontext des Systems generiert werden können, wird durch die Middleware verborgen. Der Umgang mit diesen Systemproperties gestaltet sich aus der Sicht von Nutzern und Applikationsprogrammierern genau so wie mit anderen Properties.

Die dargestellte Dokumentenmetapher gestattet die elegante Modellierung von Zusammenarbeitsvorgängen durch die Abbildung von Workflow-Vorgängen und der mit ihnen verbundenen Metadaten auf die Properties von speziellen Workflow-Dokumenten. Auf diesen Punkt soll in Abschnitt 4.7 näher eingegangen werden.

4.2 Dokumenttypen

Der genaue Aufbau eines Dokuments wird in einer Dokumenttypdefinition für die verschiedenen Dokumenttypen verbindlich festgelegt. In dieser wird u.a. definiert, aus welchen Properties ein Dokument besteht, wie oft eine bestimmte Property in einem Dokument auftreten darf oder muss (Kardinalität), von welchem Typ diese ist und welche Berechtigungen standardmäßig für Dokumente und Properties zu vergeben sind.

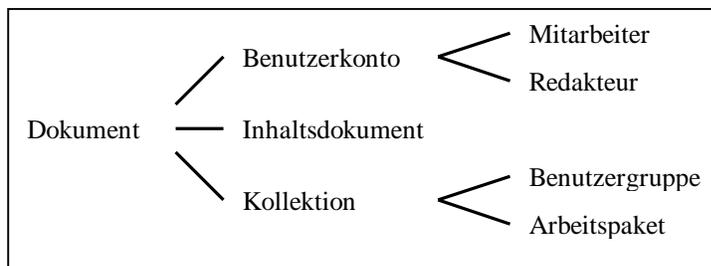


Abb. 2: Hierarchie von Dokumenttypen

Diese Definition der Dokumenttypen kann nach Bedarf durch Vererbungsmechanismen um neue Typen erweitert werden. Der abgeleitete Dokumenttyp übernimmt dabei alle Properties aus der Dokumenttypdefinition von der er abgeleitet wird. Es ist nicht zulässig, einmal definierte Properties, ähnlich wie in der objektorientierten Programmierung, zu überladen. Daher müssen Properties mit dem gleichen Namen immer vom selben Typ sein. Lediglich die Einschränkung der Kardinalität einer Property ist zulässig. Einen Überblick über die bisher im EITMMS verwendeten Dokumenttypen und ihre Hierarchie gibt Abbildung 2.

Name	Typ	Kardinalität	
		Min	Max
ID	DocumentIDValue	1	1
Name	StringValue	1	1
Typ	StringValue	1	1
Erstellung	DateValue	1	1
Letzte Änderung	DateValue	1	1
Besitzer	DocumentIDValue	1	1
Status	IntegerValue	1	1
Rechte	RightsValue	1	1
Hauptspeicherort	DocumentIDValue	1	1
Nebenspeicherort	DocumentIDValue	0	N
Beschreibung	StringValue	0	N
LockSystem	LongValue	1	1
LockBenutzer	DocumentIDValue	1	1

Abb. 3: Typdefinition für den Basistyp *Dokument*

Um Dokumente mit dem EIT-System verwalten zu können, ist es notwendig, dass sie aus einer bestimmten Grundmenge von Properties bestehen. Daher sind alle Dokumenttypen direkt oder indirekt vom Basistyp *Dokument* (siehe Abb.) abgeleitet.

Die Notation der Typdefinitionen erfolgt durch ein XML-Dokument, wobei Propertytypen auf korrespondierende Java-Klassen der Rahmenanwendung abgebildet werden.

4.3 Rechtekonzept

Zur Regelung des Zugriffs auf Dokumente und Properties existiert ein zweistufiges Berechtigungskonzept, welches einerseits Rechte über das gesamte Dokument definiert,

andererseits aber auch die feingranulare Vergabe von Berechtigungen auf der Ebene von einzelnen Properties gestattet. Auf diese Weise werden nutzerspezifische Sichten auf ein Dokument ermöglicht, und es können dem Dokument z.B. persönliche Anmerkungen und private Metainformationen hinzugefügt werden.

Auf der Ebene des Dokuments können für einen Benutzer vier einander beinhaltende Berechtigungsstufen festgelegt werden: Die Berechtigung *visible* bedeutet, dass ein Nutzer lesenden Zugriff auf das Dokument hat. Das Recht *add* erlaubt es dem Benutzer, dem Dokument weitere Properties hinzuzufügen, insofern dadurch keine Randbedingung der Dokumenttypdefinition verletzt wird. Das Recht *delete* gestattet, das Dokument als Ganzes zu löschen. Das Recht *change* erlaubt, die Berechtigungen auf Dokumentenebene zu ändern und anderen Benutzern auf dieser Ebene Rechte zu gewähren oder zu entziehen.

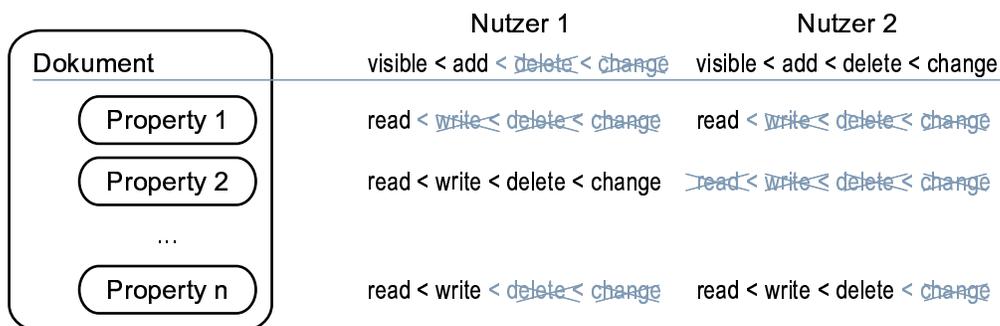


Abb. 4: Rechte auf Dokumenten- und Metadaten-Ebene

Auch auf der Ebene von einzelnen Properties können Berechtigungen für einzelne Nutzer vergeben werden, die einander ebenfalls beinhalten: Das Recht *read* erlaubt es einem Nutzer, den Wert einer Property zu lesen, *write* ermöglicht, diesen Wert zu verändern. Nutzer mit der Berechtigung *delete* dürfen die Property aus dem Dokument entfernen, sofern die in der Dokumenttypdefinition festgelegten Minimal-Kardinalitäten garantiert bleiben. Letztendlich gestattet das Recht *change*, auch auf dieser Ebene andere Nutzer zu berechtigen oder ihnen Rechte zu entziehen. In Abb. 4 sind beispielhaft die Berechtigungen für zwei verschiedenen Nutzer dargestellt.

4.4 Strukturierung des Dokumentenraums durch Kollektionen

Um den Dokumentenraum, der die Gesamtmenge aller für einen Nutzer sichtbaren Dokumente umfasst, in kleinere Teilbereiche zu strukturieren, wird das Konzept der Kollektionen verwendet. Diese gestatten es, in ähnlicher Weise wie Verzeichnisse in

Dateisystemen eine Teilmenge von Dokumenten zu selektieren, besitzen jedoch gegenüber statischen Verzeichnisstrukturen den Vorteil, dass die Selektion dynamisch durch eine Anfrage über Metadaten vorgenommen werden kann.

erweitert: <i>Dokument</i>			
Name	Typ	Kardinalität	
		Min	Max
Anfrage	StringValue	1	1
Einschlussliste	IDListValue	1	1
Ausschlussliste	IDListValue	1	1

Abb. 5: Typdefinition für den Dokumenttyp *Kollektion*

Kollektionen werden im EITMMS durch spezielle Dokumente beschrieben und sind direkt vom Basistyp *Dokument* abgeleitet, verfügen jedoch zusätzlich über die Properties *Anfrage*, *Einschlussliste* und *Ausschlussliste* (siehe Abb. 5). Durch die *Anfrage* werden aus der Gesamtmenge aller Dokumente genau diejenigen selektiert, deren Properties den angegebenen Parametern genügen. Diese Ergebnismenge wird um die in der *Einschlussliste* referenzierten Dokumente ergänzt. Die in der *Ausschlussliste* aufgezählten Dokumente werden aus der Ergebnismenge entfernt.

Auf diese Weise können einerseits statische Verzeichnishierarchien nachgebildet werden, was der Verständlichkeit und Benutzbarkeit des Systems zugute kommt. Andererseits wird es durch Kollektionen möglich, auf sehr einfache Weise Funktionen zur Verfügung zu stellen, die einem Anwender z.B. neu hinzu gekommene Dokumente in einer gesonderten Kollektion anzeigen. Auch das Einordnen eines Dokuments in mehrere Kollektionen ist möglich, wodurch starre Ordnungsmechanismen vermieden werden.

4.5 Verteilung und Konfliktvermeidung

Dokumente werden auf miteinander vernetzten Rechnern verteilt abgelegt. Die Kommunikation zwischen den einzelnen lokalen Instanzen des EITMMS erfolgt auf Peer-to-Peer-Basis und wird unter Nutzung des Frameworks JXTA [22] implementiert.

Die Verteilung von Dokumenten zwischen unterschiedlichen Rechnern wird über die Properties Hauptspeicherort (HSO) und Nebenspeicherort (NSO) gesteuert. Diese Properties referenzieren Rechnersysteme, auf denen eine lokale Instanz des EITMMS ausgeführt wird.

Jedes Dokument besitzt genau einen HSO, der als Synchronisationspunkt für die Vermeidung von Konflikten durch die zeitgleiche Bearbeitung von Dokumenten verwendet wird. Werden einem Dokument ein oder mehrere NSO zugewiesen, erstellt das System ein Replikat auf dem angegebenen Rechner und sorgt für dessen Aktualisierung.

Schreibender Zugriff auf ein Dokument ist nur dann möglich, wenn das lokale System auf dem Hauptspeicherstandort das Setzen einer Sperre auf das betreffende Dokument veranlassen kann, welche durch die Properties *LockSystem* und *LockBenutzer* beschrieben wird. Diesen Properties kann entnommen werden, durch welchen Benutzer auf welchem lokalen System das Dokument gerade bearbeitet wird.

Um nicht auf eine permanente Netzwerkverbindung angewiesen zu sein, kann ein Nutzer eine temporäre Sperre auf dem Dokument errichten (Check-Out). Das Dokument bleibt so lange gesperrt, bis es vom Benutzer explizit wieder zur Bearbeitung durch andere freigegeben wird (Check-In). In der Zwischenzeit kann das Dokument ausschließlich durch den sperrenden Nutzer und auch nur auf der lokalen Instanz des EITMMS bearbeitet werden, von der aus die Sperre errichtet wurde.

4.6 Inhaltsdokumente, externe Anwendungen und interne Editoren

Die eigentliche Bearbeitung von Daten und Inhalten erfolgt i.d.R. durch externe Anwendungen, wie z.B. Grafikprogramme oder Texteditoren. Diese sind darauf angewiesen, dass die zu bearbeitenden Daten in Form einer oder mehrerer Dateien übergeben werden. Ausnahmen bilden Anwendungen, die z.B. ODMA (Open Document Managemen API) unterstützen. Allerdings gibt es nur eine sehr begrenzte Zahl von Anwendungen, bei denen das der Fall ist. Daher kann nicht davon ausgegangen werden, dass alle notwendigen externen Werkzeuge diese Unterstützung anbieten.

Im EITMMS werden sogenannte Inhaltsdokumente verwendet, um mit Daten umzugehen, die für die Bearbeitung durch externe Anwendungen bestimmt sind. Diese kapseln in einer Inhaltsobjekt-Property ein beliebiges Binärobjekt und enthalten

Informationen zum Typ dieses Objektes. Das EITMMS assoziiert diesen Typ mit Anwendungen, die auf dem lokalen System installiert sind. Entscheidet sich der Nutzer, ein Inhaltsdokument zu bearbeiten, wird ihm eine Auswahl der verfügbaren Anwendungen angeboten. Das System überträgt den Inhalt des Inhaltsobjektes in eine temporäre Datei und übergibt diese der ausgewählten Anwendung. Anschließend wird die erzeugte Datei vom System überwacht, um Veränderungen ihres Inhaltes zu registrieren und diese gegebenenfalls in das Inhaltsobjekt zu übernehmen.

Diese Vorgehensweise erlaubt prinzipiell die Integration beliebiger externer Werkzeuge. Problematisch allerdings ist die Benutzung solcher Anwendungen, die innerhalb der Binärdaten Referenzen auf andere Datenobjekte halten. Um die Integrität dieser Dokumente zu wahren, ist es notwendig, dass das verwendete Format dokumentiert ist und die Referenzen durch Import- bzw. Exportfilter des EITMMS ersetzt und auf Inhaltsdokumente abgebildet werden können.

Für eine Reihe von Dokumenten, wie z.B. Kollektionen, Benutzerkonten und Arbeitspakete, ist es nicht sinnvoll, auf externe Editoren zurückzugreifen. Für diese Dokumente kommen interne Editoren zum Einsatz. Über eine Modulschnittstelle kann das EITMMS um interne Editoren für neue Dokumenttypen erweitert werden. Diese Editoren sind nicht auf die Übergabe von Dateien angewiesen sind, sondern können über ein API direkt auf Dokumente und deren Properties zugreifen.

4.7 Workflows und Kommunikationsmittel

Die Unterstützung von Workflows erfolgt durch Abbildung aller relevanten Metadaten, durch die der Arbeitsablauf beschrieben wird, auf die Properties eines Workflow-Dokuments. Für dieses Dokument existieren interne Editoren, durch die auf der Benutzeroberfläche die Abarbeitung des Workflows gesteuert wird.

Beispielhaft wurde ein Auftrags-Review-Zyklus umgesetzt, bei dem ein Auftraggeber einen Arbeitsauftrag für einen Auftragnehmer definiert. Das zur Modellierung verwendete Dokument *Arbeitspaket* (siehe Abb. 66) ist von einer Kollektion abgeleitet. Dadurch können weitere Dokumente mit diesem Workflow verknüpft werden, die z.B. Halbzeuge oder Vorlagen enthalten.

erweitert: <i>Kollektion</i>			
Name	Typ	Kardinalität	
		Min	Max
Auftragsbeschreibung	StringValue	1	1
Auftragsstatus	IntegerValue	1	1
Bearbeitungsstatus	IntegerValue	1	1
Bearbeiter	DocumentIDValue	1	1
Verlauf	StringValue	1	1

Abb. 6: Typdefinition für den Dokumenttyp *Arbeitspaket*

Der Auftrag wird durch den Auftraggeber ausgelöst, indem der Arbeitspaket-Editor (siehe Abb. 8) die Werte verschiedener Properties und die Zugriffsrechte für Auftragnehmer und Auftraggeber verändert. Nach dem Auslösen des Auftrags kann sich der Auftragnehmer entscheiden, ob er den Auftrag akzeptiert oder ablehnt. Wird der Auftrag abgelehnt, kann der Auftraggeber den Arbeitsauftrag modifizieren und ggf. einen anderen Auftragnehmer angeben. Wird der Auftrag akzeptiert, erhält der Auftragnehmer Zugriff auf die im Workflow-Dokument enthaltenen bzw. referenzierten Dokumente und kann eigene Dokumente hinzufügen. Ist die Bearbeitung abgeschlossen, kann sich der Auftraggeber entscheiden, ob er die Ergebnisse akzeptiert oder ablehnt.

Alle Aktionen werden dabei in der Property *Verlauf* dokumentiert, wodurch ein Nachvollziehen des Bearbeitungsweges möglich wird. Der Bearbeitungszustand kann den Properties *Auftragsstatus* und *Bearbeitungsstatus* entnommen werden und wird durch die Verwendung unterschiedlicher Icons visualisiert. Der Transport von Arbeitspaketen und der zugehörigen Dokumente zwischen den Rechnern von Auftraggeber und Auftragnehmer erfolgt transparent durch die Verteilungsmechanismen des EITMMS.

Neben der Kommunikation an Arbeitsartefakten ist die Bereitstellung expliziter Kommunikationsmittel vorgesehen. Dies wird dadurch vereinfacht, dass

Benutzerkonten ebenfalls durch Dokumente beschrieben werden und innerhalb des EITMMS alle notwendigen Metadaten über den Benutzer, wie z.B. sein Online-Status, bekannt sind. An der Einbindung bzw. Implementation externer und interner Kommunikationswerkzeuge wird zurzeit gearbeitet.

5. Umsetzung der Konzepte im Prototypen

Die beschriebenen Konzepte wurden zu wesentlichen Teilen in Form eines Prototypen als Java-Anwendung umgesetzt. Dieser umfasst u.a. einen lokalen Persistenzmechanismus, ein gegenüber der Beschreibung leicht vereinfachtes Typungs- und Berechtigungskonzept, eine Schnittstelle zu Standardanwendungen und einen Grundbestand an Dokumenttypen und zugehörigen internen Editoren. Es wurden dabei primär solche Dokumenttypen umgesetzt, die für die Demonstration der Konzepte unentbehrlich sind: z.B. Benutzerkonten, Kollektionen und Inhaltsdokumente.

Die Benutzersicht besteht aus einem dreigeteilten Hauptfenster (siehe Abb. 7). Im linken Teil wird ein benutzerspezifischer Ausschnitt aus dem Dokumentenraum in Form eines ausklappbaren Baumes angezeigt, dessen Wurzelement die sogenannte Root-Kollektion des Benutzers ist. Die einzelnen Knoten des Baumes stellen jeweils die im übergeordneten Knoten enthaltenen Kollektionen dar, die Blätter visualisieren die enthaltenen Dokumente. Die Root-Kollektion stellt für jeden Benutzer einen personalisierten Ausgangspunkt zum Aufbau einer eigenen Ordnungshierarchie über der gesamten Dokumentenmenge dar.

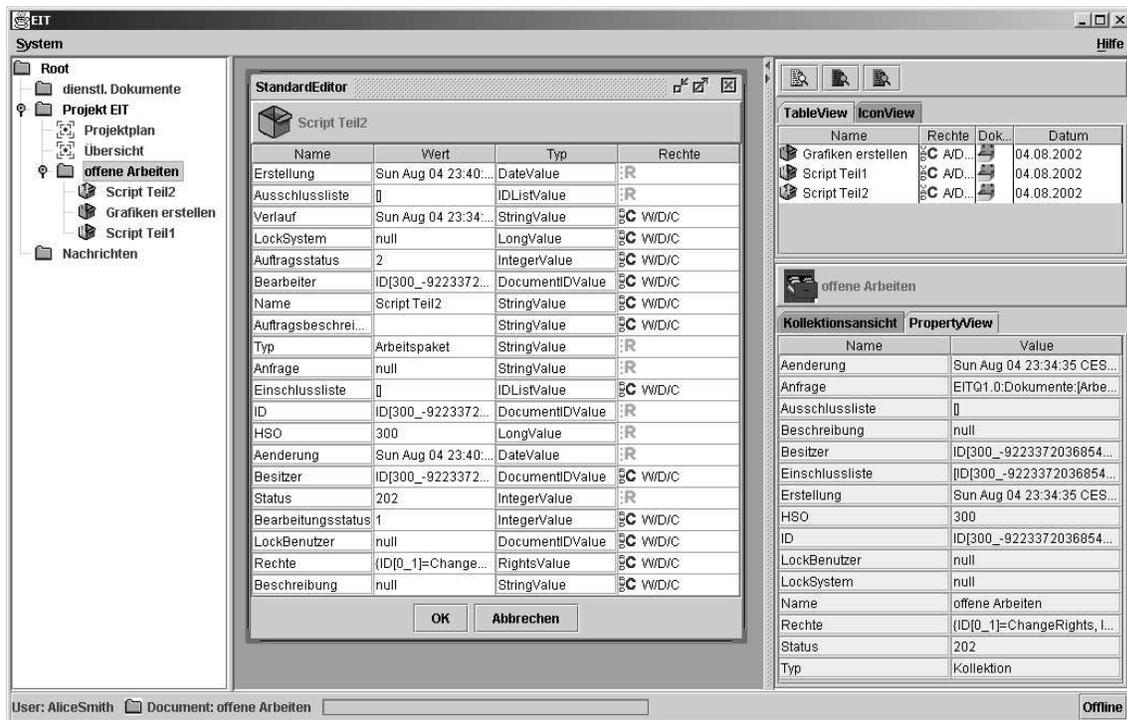


Abb. 7: Benutzeroberfläche des EITMMS-Prototypen

Im mittleren Teil ist Raum für die Anzeige von internen Editoren. In der Abbildung ist der Standard-Editor zu sehen, mit dessen Hilfe die Properties beliebiger Dokumente bearbeitet werden können. Auch die Berechtigungen für Dokumente und Properties können mit dem Standard-Editor geändert werden. Der in Abb. 88 dargestellte Editor für Arbeitspakete wird ebenfalls im mittleren Bereich angezeigt.

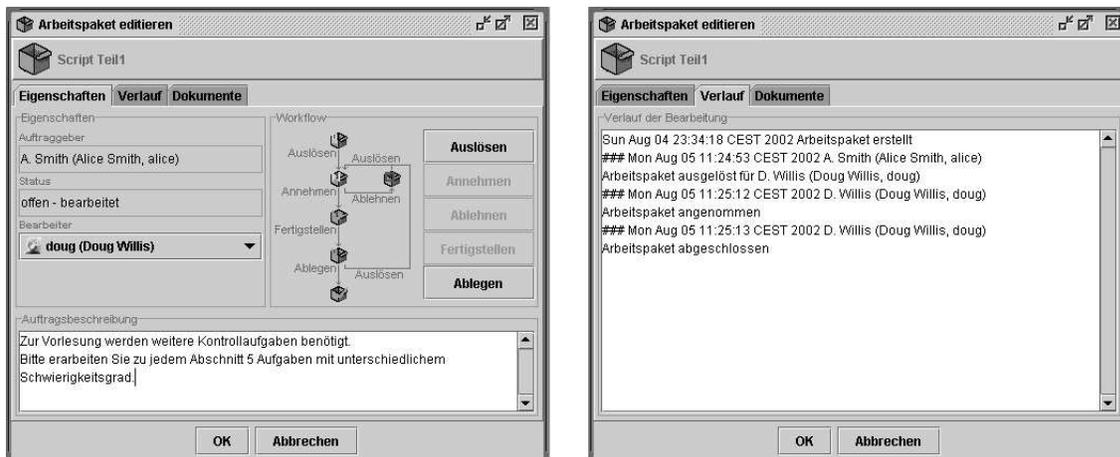


Abb. 8: Editor für Arbeitspakete

Das rechte Drittel des Fensters ist nochmals unterteilt in einen unteren Bereich, in dem alle für ein Dokument verfügbaren Anzeigemodule eingebettet sind und einen oberen Bereich zur detaillierten Anzeige des Inhaltes von Kollektion.

Über eine Modulschnittstelle kann das EITMMS um weitere Dokumenttypen, Editoren und Anzeigemodule erweitert werden. Die zugehörigen Menüeinträge, z.B. für den Aufruf von Editoren, die Erstellung neuer Dokumente oder zum Ausführen von dokumentenspezifischen Funktionen, werden automatisch in die Menüleiste und die Kontextmenüs der Benutzersicht eingebunden. Auch der Editor für Arbeitspakete, dessen Funktionalität weiter oben beschrieben wurde, ist über die Modulschnittstelle eingebunden.

6. Zusammenfassung und Ausblick

Auch wenn noch nicht alle vorgesehenen Funktionen des EITMMS implementiert und erprobt werden konnten, lässt sich aus den bisher mit dem Prototypen gesammelten Erfahrungen eine insgesamt positive Bilanz ziehen. Die vorgestellten Konzepte für Dokumente und Kollektionen sowie die Paradigmen der Loslösung der Dokumente von ihren Speicherorten und der Organisation des Dokumentenbestands über die Eigenschaften der einzelnen Dokumente bieten zur Verwaltung digitaler Ressourcen eine ausreichende Funktionalität. Sie sind aus der Benutzersicht ähnlich gut handhabbar wie Dateien und Verzeichnisse.

Nach dem Abschluss der noch ausstehenden Arbeiten an den Verteilungsmechanismen steht mit dem EITMMS ein Werkzeug zur Verfügung, mit dessen Hilfe verteilt arbeitende Nutzergruppen und virtuelle Gemeinschaften auf einem gemeinsamen Dokumentenbestand arbeiten können, ohne dass die einzelnen Nutzer mit dem Transport von Dokumenten auf das lokale System oder mit der Vermeidung von Konflikten belastet werden. Da ein Nutzer keine permanente Netzwerkanbindung benötigt, wird insbesondere den Bedürfnissen solcher Gruppen Rechnung getragen, die auf den Einsatz mobiler Rechentechnik angewiesen sind oder einen substantiellen Anteil von Mitgliedern ohne dauerhaften Netzwerkzugang besitzen.

Die Umsetzung des Workflow-Szenarios konnte auf Grundlage der Dokumenten-Metapher unter Nutzung der bereitgestellten APIs mit vergleichsweise geringem Aufwand realisiert werden, was für die Praktikabilität des Ansatzes spricht. Auch der Einarbeitungsaufwand für die zu verwendenden Schnittstellen ist relativ gering.

Problematisch ist der Umgang mit Inhaltsdokumenten, in deren Binärobjekten Referenzen auf Dateien gespeichert sind. Die damit verbundenen Schwierigkeiten könnten entweder durch spezielle Im- und Exportfilter für alle zum Einsatz kommenden Dateiformate oder durch erhebliche Eingriffe in die Dateisystem-Schnittstelle des Betriebssystems, z.B. durch die Entwicklung eines virtuellen Filesystem-Treibers umgangen werden. Diese Lösungen setzten allerdings nicht an der Ursache an, die darin besteht, das sich noch keine speicherortunabhängigen Mechanismen zum Zugriff auf Datenobjekte durchgesetzt haben. Solche Mechanismen müssten auf Betriebssystemebene etabliert und durch eine kritische Masse von Applikationen unterstützt werden.

Für die Dokumenttypdefinition ist eine starke Anlehnung an Standardtechnologien, insbesondere XML-Schema anzustreben, um z.B. Abhängigkeiten zwischen Properties und die Definition von Randbedingungen für die Rechtevergabe modellieren zu können, ohne auf proprietäre Lösungen zurückgreifen zu müssen. Um den notwendigen Implementierungsaufwand in Grenzen zu halten, sind allerdings die Fortschritte bei der Implementierung von DOM Level 3 [23] abzuwarten, da dort definierte Mechanismen benötigt werden, die zurzeit noch nicht realisiert sind.

Der nächste Schritt zur Evaluierung des EITMMS ist der Einsatz beim Projektpartner im Rahmen eines Testszenarios, an dem sich externe und interne Mitarbeiter des Partners beteiligen. Dabei soll insbesondere auch geprüft werden, inwieweit synchrone und asynchrone Kommunikationswerkzeuge integriert werden sollten.

7. Literatur

- [1] Hartwig, R.; Herczeg, M.; Kritzenberger, H.: Aufgaben- und benutzerzentrierte Entwicklungsprozesse für web-basierte Lernumgebungen. In: *i-com : Zeitschrift für interaktive und kooperative Medien* 1 (2002), S. 18-24
- [2] IBM Lotus Domino.Doc Produkt-Homepage. URL <http://www.lotus.com/products/domdoc.nsf/content/domdochohomepage>
- [3] Novell GroupWise 6, Detailed View. URL <http://www.novell.com/products/groupwise/details.html>
- [4] Vignette V6 Content Suite. Hersteller-Homepage. URL <http://www.vignette.com/>

-
- [5] Gauss Interprise. Gauss VIP Content Manager. URL http://www.gaussvip.com/Products/VIP_ContentManager.htm
- [6] Infopark. NPS - Network Productivity System. Produkt-Homepage. URL <http://www.infopark.de/de/products/productinfo/nps/>
- [7] Groove Networks, Inc. Homepage. URL <http://www.groove.net>.
- [8] Zope Community. Projekt-Homepage. URL <http://www.zope.org/>
- [9] OpenCMS Project. OpenCMS. Projekt-Homepage URL <http://www.opencms.com>
- [10] Midgard - Open Source Application Server. URL <http://www.midgard-project.org/>
- [11] Feig & Partner GmbH: contentmanager.de – die deutsche content management site. URL <http://www.contentmanager.de>
- [12] Fiala, Z.: Content Management Systeme. *Diplomarbeit an der TU Budapest und der TU Dresden* (2001)
- [13] Würkert, T.: Dokumentenmanagement in einer teamorientierten Arbeitsumgebung. *Diplomarbeit an der TU Dresden* (2002)
- [14] Concurrent Versions System (CVS) Homepage. URL <http://www.cvshome.org>. Aktualisierung: 04/2002
- [15] BSCW (Basic Support for Cooperative Work) Homepage. *FIT und OrbiTeam Software GmbH*. URL <http://bscw.gmd.de>. Aktualisierung: 07/2002
- [16] TOWER Projekt-Homepage. Fraunhofer Institut für Angewandte Informationstechnik. URL http://www.fit.fraunhofer.de/projekte/tower/index_en.xml
- [17] Prinz, W.; Pankoke-Babatz, U.; Broll, W.: NESSIE: eine Infrastruktur zur Wahrnehmung für virtuelle Teams. In: *GMD-Spiegel* 1/2 (1999). URL http://www.gmd.de/de/GMD-Spiegel/GMD-Spiegel-1_2_99-html/NESSIE.html.
- [18] Xerox Palo Alto Research Center: Placeless Documents Project-Archive. URL <http://www.parc.xerox.com/csl/projects/harland>

- [19] Xerox Palo Alto Research Center: Harland - fast attribute storage. Projekt-Homepage. URL <http://www.parc.xerox.com/csl/projects/harland>
- [20] LaMarca, A.; Keith, E. W.; Dourish P.; et al: Taking the Work out of Workflow: Mechanisms for Document-Centered Collaboration. In: *Proceedings of the Sixth European Conference on Computer-Supported Cooperative Work* (1999)
- [21] Science-To-Science (S2S) Projekt-Homepage. *neofonie - Technologieentwicklung und Informationssysteme GmbH Berlin*. URL http://www.neofonie.de/profil/forschung_und_entwicklung/s2s.jsp
- [22] JXTA Projekt-Homepage. URL <http://www.jxta.org/>
- [23] Chang, B. (Hrsg.); Litani, E. (Hrsg.); et al: Document Object Model (DOM) Level 3 Abstract Schemas and Load and Save Specification. Version 1.0. W3C Working Draft. URL <http://www.w3.org/TR/2002/WD-DOM-Level-3-ASLS-20020409>. Aktualisierung: 09.04.2002