

Dokumentenmanagement für Verwaltungsvorgänge im E-Government *

R. Kunis, G. Rünger, M. Schwind

Fakultät für Informatik

Technische Universität Chemnitz

Straße der Nationen 62, 09111 Chemnitz

EMail: {krap,ruenger,schi}@informatik.tu-chemnitz.de

Zusammenfassung

Moderne Verwaltungen in Deutschland stehen zunehmend vor der Herausforderung, trotz stetigem Abbau der finanziellen Mittel die Bearbeitungsdauer und -qualität ihrer Verwaltungsvorgänge sowie die Flexibilität bei der Bearbeitung zu steigern. Das Forschungsprojekt „Referenzarchitektur für E-Government“ (RAfEG) hat das Ziel diese Verwaltungsvorgänge in Behörden elektronisch zu unterstützen und damit die Herausforderung anzunehmen. Der Großteil der Aufgaben des E-Government, was in diesem Zusammenhang vor allem als Government-to-Government verstanden werden soll, befasst sich mit der Bearbeitung von Dokumenten. Es müssen Dokumente (z. B. Anträge und Formulare) erfasst/erstellt, bearbeitet, gespeichert, weitergegeben und archiviert werden. Für das so genannte papierlose Büro, muss dies alles mittels eines Computers möglich sein und alle Mitarbeiter müssen auf denselben Daten arbeiten können, wobei eine nachvollziehbare Änderungshistorie gesichert sein muss. Die Realisierung des Dokumentenmanagementsystem innerhalb des RAfEG-Softwaresystems, das eine rechnergestützte Abarbeitung von Verwaltungsvorgängen ermöglicht, soll in diesem Bericht ausführlich vorgestellt werden.

*Das Projekt RAfEG wurde mit Mitteln des Bundesministeriums für Bildung und Forschung (BMBF) im Rahmen der Forschungsoffensive „Software Engineering 2006“ unter dem Förderkennzeichen 01 IS C07 C gefördert. Das Projekt setzte sich zusammen aus dem Institut für Wirtschaftsinformatik (IWi) im Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI), der NHConsult GmbH und der Professur Praktische Informatik der Technischen Universität Chemnitz. Weitere Informationen sind auf der Projekthomepage <http://www.rafeg.de> zu finden.

1 Einleitung

Dokumentenmanagementsysteme (DMS) spielen eine zentrale Rolle bei der Abarbeitung von Verwaltungsvorgängen. Der Einsatz eines flexiblen und vor allem freien, auf Open-Source Software basierenden DMS ist dabei von großer Bedeutung. Im Rahmen des Forschungsprojektes RAfEG bestand eine Herausforderung darin, ein solches System in die Gesamtarchitektur zu integrieren und an die speziellen Bedürfnisse von Behörden anzupassen. Diese Bedürfnisse beinhalten die einfache Integration in die bestehende IT-Infrastruktur, die revisionssichere Archivierung der Dokumente und die leichte Anbindung an das zu verwendende System zur Abarbeitung der Verwaltungsvorgänge. Für die erstellte Lösung standen drei wesentliche Punkte im Vordergrund: das Konzept der Prozessmappen, die Einführung von Sicherheitsleveln für Dokumente und die Möglichkeit der Klassifikation von Dokumenten nach deren Änderbarkeitsstatus.

Dieser Bericht ist wie folgt strukturiert: In Abschnitt 2 werden die drei Kernkonzepte vorgestellt. Abschnitt 3 befasst sich mit Dokumentenmanagementsystemen, die für den Einsatz geprüft wurden. Die erstellte Lösung wird in Abschnitt 4 detailliert erläutert. Abschnitt 5 stellt kurz die Zusammenarbeit der erstellten Lösung für das Dokumentenmanagement mit den anderen Bausteinen des RAfEG-Softwaresystems dar. In Abschnitt 6 wird die erstellte Lösung mit anderen aktuellen Lösungen für Dokumentenmanagementsysteme im E-Government verglichen. Eine abschließende Zusammenfassung wird in Abschnitt 7 gegeben.

2 Konzepte für das Dokumentenmanagement

In diesem Abschnitt werden die grundlegenden Konzepte der implementierten Lösung für das Dokumentenmanagement vorgestellt. Diese Konzepte gehen über die standardmäßigen Möglichkeiten des Dokumentenmanagements hinaus und bilden eine notwendige Erweiterung für Verwaltungsvorgänge im E-Government.

Prozessmappen Dokumentenmanagement im Bereich des E-Government befasst sich mit der Bereitstellung und Bearbeitung von Dokumenten für einzelne Verfahren. Ein Verfahren ist dabei ein in sich abgeschlossener Verwaltungsvorgang, der mit einem Antrag beginnt und der fertigen Abarbeitung des Verfahrens endet. Die verwendeten Dokumente sind dabei zum einen vorliegende Formulare, die während der Abarbeitung ausgefüllt werden, Gesetzesvordrucke, die eingesehen werden können und andere verfahrensspezifische Dokumente, z. B. Kartenmaterial im Falle des Planfeststellungsverfahrens. Das Planfeststellungsverfahren, das im Weiteren als Beispielverfahren dient, wird für übergeordnete raumbedeutsame Fachplanungen von Bauvorhaben angewendet.

Das Konzept der Prozessmappen funktioniert nach dem Prinzip, alle Daten eines Vorgangs zu speichern und alle Änderungen festzuhalten, d.h ein Verwaltungsverfahren ist immer an eine Prozessmappe gebunden. Die Prozessmappe wird dabei zum Start eines Vorganges initial als Verzeichnis erzeugt und wächst mit der Abarbeitung des Vorgangs an. Für alle Dokumente, die als Vorlagen zu Verwaltungsvorgängen dienen, ist eine Originalversion in einem Ordner mit Vorlagen gespeichert. Die initial benötigten Dokumente eines Vorgangs werden beim Start automatisch in die noch leere Prozessmappe kopiert. Während der Abarbeitung des Verfahrens können diese Dokumente geändert werden. Genauso können auch neue Dokumente hinzugefügt werden, z. B. Einsprüche von Bürgern,

Zusatzformulare für bestimmte Fälle, die nicht immer zu beachten sind, usw. Für Verwaltungsvorgänge ist es dabei von großer Bedeutung, dass für geänderte Dokumente alle Versionen, wobei eine Version einen Zustand des Dokumentes zwischen zwei Änderungen beschreibt, in der Prozessmappe aufbewahrt werden. Der Zugriff auf ein Dokument erfolgt dabei standardmäßig auf die aktuellste Version. Ältere Versionen können unter Angabe der Versionsnummer abgefragt werden. Ist ein Rücksprung zu einer alten Version aufgrund eines Bearbeitungsfehlers notwendig, kann eine alte Version als aktuelle Version gesetzt werden. Dokumente bzw. Dokumentversionen nicht gelöscht werden, da für die Archivierung alle Dokumente vorhanden sein müssen. Auf die genaue technische Realisierung der Prozessmappen wird später in Abschnitt 4 detailliert eingegangen.

Da bei größeren Verwaltungsvorgängen oft eine Hierarchie von kleineren Teilvorgängen vorliegt, die auch als einzelne Vorgänge gesehen werden können, ist das Konzept der Prozessmappen auch in dieser Hinsicht geeignet. Betrachtet man einen Vorgang, der aus Teilvorgängen besteht, so ist leicht zu erkennen, dass bei der Ausführung des größeren Vorgangs die einzelnen kleineren Vorgänge ausgeführt werden. Da diese kleineren Vorgänge dabei als eigenständig angesehen werden, besitzen sie ihre eigene Prozessmappe. Um Dokumente des Gesamtvorgangs nutzen zu können, werden bei Start der Teilvorgänge nicht die Vorlagen von Formularen u. Ä. in die leere Prozessmappe kopiert, sondern die entsprechenden Dokumente des Gesamtvorgangs. Diese können dann im Teilvorgang genutzt werden, d. h. eingesehen und wenn notwendig bearbeitet. Nachdem ein Teilvorgang abgeschlossen ist, werden diese Dokumente dann als neue Version der Ursprungsdokumente in die Prozessmappe des Gesamtvorgangs zurückkopiert.

Sicherheitslevel Das RAfEG-Softwaresystem bietet die Möglichkeit Verwaltungsvorgänge behördenübergreifend abzuarbeiten. Die meisten Vorgänge innerhalb einer Behörde beziehen andere Institutionen mit in die Abarbeitung eines solchen Vorgangs ein. Im Falle des Planfeststellungsverfahrens sind dies z. B. Träger öffentlicher Belange oder Bürger. Für die verteilte Prozessabarbeitung ist es notwendig, Informationen zwischen den beteiligten Stellen zu übertragen. Da in diese Übertragung mehrere Netze (vom sicheren internen Netz einer Behörde über den sächsischen InfoHighway zur Verbindung von Behörden, bis hin zum unsicheren Internet ist dabei alles vertreten) involviert sind, müssen spezielle Sicherheitsmechanismen für sensible Informationen umgesetzt werden. Um die Informationen, speziell die Dokumente, je nach genutztem Verbindungstyp und Inhalt schützen zu können, werden im Dokumentenmanagement des RAfEG-Softwaresystems Sicherheitslevel vergeben. Die Einordnung von Dokumenten in Sicherheitslevel ermöglicht die Sicherung von Informationen in Abhängigkeit vom Dokument und vom Übertragungsweg. Eingeteilt werden können die Dokumente dabei in ein dreistufiges Sicherheitskonzept. Weitere Informationen und eine detaillierte Beschreibung der Sicherheitsmechanismen ist in [JKR06] zu finden.

Klassifikation der Dokumente Die Dokumente innerhalb des Dokumentenmanagements lassen sich in statische und dynamische Dokumente unterteilen. Statische Dokumente dürfen nicht verändert werden, was zum einen initial gelten kann, also sobald sie in das Dokumentenmanagementsystem kopiert wurden oder zur Laufzeit festgelegt werden kann. Dynamische Dokumente werden während der Abarbeitung eines Vorgangs durch Bearbeiter geändert.

Statische Dokumente werden entweder durch einen Administrator im DMS-Repository abgelegt, so genannte Vorlagen, oder während der Abarbeitung geschlossen. Vorlagen

stellen die Grundlage für die dynamischen Dokumente dar. Beispiele für Vorlagen sind Vordrucke, unausgefüllte Formulare usw. Diese statischen Dokumente werden für die Verwaltungsvorgänge fest in deren Beschreibung als Eingaben vorgegeben. Das bedeutet, dass ein Verwaltungsvorgang, der abgearbeitet werden soll am Anfang seine Vorlagen in den Abarbeitungsordner (Prozessmappe) kopiert. Während der Abarbeitung des Vorgangs können diese kopierten Dokumente dann geändert werden und sind somit dynamisch. Die Vorlagen sind von den Änderungen jedoch nicht betroffen. Die Änderungen an den Dokumenten in der Prozessmappe, die als dynamische Dokumente bezeichnet werden, werden in einer Historie nachvollziehbar gespeichert. Dynamische Dokumente können während der Abarbeitung geschlossen werden, was sie für den Verwaltungsvorgang zu statischen Dokumenten macht, die nicht mehr geändert werden können. Das Schließen ist notwendig, um Dokumente dahingehend zu sichern, dass sie in einer Bearbeitung durch unsichere Quellen bzw. nachträgliche Änderung geschützt sind. Ist ein Prozess abgeschlossen, werden alle Dokumente archiviert. Archivierte Dokumente sind wieder statische Dokumente und können nicht mehr geändert werden.

3 Systeme für Dokumentenmanagement

Dokumentenmanagementsysteme unterstützen Softwaresysteme bei der Erstellung/Erfassung, Änderung, Speicherung, Weitergabe und Archivierung von elektronischen Dokumenten. Es existiert eine Vielzahl an Dokumentenmanagementsystemen, die sich in der Unterstützung des genannten Basisfunktionalitäten und bei Zusatzfunktionen unterscheiden.

Für das Forschungsprojekt RAfEG wurde ein System- und Kriterienkatalog für DMS [BHRV05] erstellt, der auf den Standards und Architekturen für E-Government-Anwendungen [Bun03] aufbaut. In diesem sind die wichtigsten Anforderungen an ein DMS-System ausführlich dargestellt: Erstellung/Erfassung, Änderung, Prüfung, Verteilung, Dokumentenverwendung und Archivierung.

3.1 Bedeutung von Open-Source

Bei Software im Bereich der öffentlich Verwaltung ist davon auszugehen, dass sie über viele Jahre hinweg im Einsatz ist und durch den Wandel an Anforderungen und sich ändernde Hardware- und Softwareplattformen ständig den sich ändernden Gegebenheiten angepasst werden muss. Bei der DMS-Komponente des RAfEG-Softwaresystems zur Abarbeitung von Verwaltungsvorgängen im E-Government kommt als weiterer wichtiger Punkt hinzu, dass gespeicherte Daten teilweise über mehrere Jahrzehnte verfügbar bleiben müssen. Setzt man für die DMS-Komponente Closed-Source Software von Drittanbietern ein, besteht die Gefahr, dass aufgrund von Akquisitionen, Insolvenz oder strategischen Entscheidungen der Drittanbieter das DMS-System nicht in den entsprechenden langen Zeiträumen gepflegt bzw. weiterentwickelt wird, wie es die Software in öffentlichen Verwaltungen benötigt.

Das Problem der Software-Wartung über viele Jahre lässt sich bei der Nutzung von Closed-Source Software durch den Einsatz offener Schnittstellen bzw. Standards umgehen. So kann bei der Einstellung der Wartung der Software durch den Drittanbieter auf ein anderes System, das die gleichen offenen Schnittstellen implementiert, umgestiegen werden.

Das Problem beim Einsatz von Closed-Source Software mit offenen Schnittstellen ist jedoch, dass auf ein neues System migriert werden muss, wenn das alte System nicht mehr gewartet wird oder von neuer Hardware/Software nicht mehr unterstützt wird. Die Migration der Daten des alten Systems auf das neue System kann dabei sehr aufwendig oder auch unmöglich sein, vor allem unter dem Gesichtspunkt sehr großer Datenbestände von Behörden die seit mehreren Jahrzehnten vorliegen. Für DMS-Systeme existieren die offenen Standards ODMA [DMW], DMA [Doc] und WebDAV/DeltaV [GWF⁺99], die alle aufgrund ihrer standardisierten Schnittstellen die Möglichkeit bieten, das DMS-System auszutauschen. ODMA und DMA haben allerdings den Nachteil, dass sie nur auf Windows-Plattformen zur Verfügung stehen und auf Technologien dieser Plattform aufsetzen, die selbst nicht offene Standards sind. Dadurch verschiebt sich das Problem der Verfügbarkeit, Wartbarkeit und Pflege von der Software auf das zugrunde liegende Betriebssystem.

Aus oben genannten Gründen basiert die DMS-Komponente des RAfEG-Softwaresystems auf freier Open-Source Software mit freien standardisierten Schnittstellen. Ein weiterer Vorteil von Open-Source Software ist dabei eine aktive Entwicklergemeinschaft, die über viele Jahre eine aktive Entwicklung/Anpassung von Software im eigenen Interesse durchführt. Außerdem bietet sie den Vorzug im Quellcode vorzuliegen und ermöglicht damit, Veränderungen bzw. Anpassungen selbst vorzunehmen. Weiterhin ist Open-Source Software meist lizenzkostenfrei verfügbar. Bei kommerzieller Closed-Source Software können für Anschaffung und für Wartungsverträge über mehrere Jahre erhebliche Kosten auf den Nutzer zukommen.

In den nächsten Abschnitten werden Protokolle und Systeme beschrieben, die sich für den Einsatz als Grundlage des DMS-System für das RAfEG-Softwaresystem eignen.

3.2 Protokolle für Dokumentenmanagementsysteme

Im Bereich des Dokumentenmanagements haben sich die folgenden Protokolle etabliert: DMA, ODMA und WebDAV. DMA und ODMA sind Protokolle, die auf OLE bzw. DCOM aufbauen beides Objektmodelle für die Windows Plattform. Durch diese Einschränkung des nutzbaren Betriebssystems wird das WebDAV-Protokoll als offene Schnittstelle zwischen DMS-System und den weiteren RAfEG-Softwaresystemkomponenten eingesetzt. Ein weiterer Vorteil von WebDAV ist, dass es aufgrund seiner großen Verbreitung häufig in Standardsoftware integriert ist.

WebDAV ist eine Erweiterung des HTTP/1.1 Protokolls. Es erweitert das HTTP/1.1 Protokoll um neue Headerinformation und Methoden, die den Schreibzugriff auf verteilte Dokumente verbessern. Die genauen Erweiterungen zu Http/1.1 sind:

- Collections: Als Collections werden Verzeichnisse verstanden. Es wurden Methoden zur deren Erstellung zum HTTP-Standard hinzugefügt.
- Locking: Verzeichnisse und Dateien können durch einen Write-Lock vor überschreiben geschützt werden.
- Properties: Metadaten können an Verzeichnisse und Dokumente angehängt werden. Diese können von Entwicklern für erweiterte Funktionalitäten genutzt werden.
- Copy/Move Methoden: Es wurden Methoden zum Kopieren und Verschieben von Dateien und Verzeichnissen zum Standard hinzugefügt.

Während der ursprüngliche WebDAV-Standard die Versionierung von Dokumenten nicht beinhaltete, gibt es in der Erweiterung WebDAV/DeltaV [Net02] die Möglichkeit, Dokumente zu versionieren und unterschiedliche Bearbeitungszustände vorzuhalten, die nach der Bearbeitung zusammengefügt werden können (checkout/checkin). Eine weitere Erweiterung, die DeltaV bietet, ist die Autoversionierung von Dateien. Dabei wird für jede Veränderung an einer Datei oder an den Metadaten einer Datei eine neue Version des Dokuments erzeugt. Diese Funktionalität ist für das RAfEG-System ausreichend, um die Revisionsicherheit zu gewährleisten. Da WebDAV eine Erweiterung zum bestehenden HTTP/1.1 Standard ist, können die Autorisierungs- und Authentifizierungsmethoden von HTTP/1.1 genutzt werden. Weitere von HTTP/1.1 geerbte, wichtige Eigenschaften sind die Verschlüsselung der Übertragung, die Unterstützung von Proxies, sowie das Caching von Daten. Aufgrund der Popularität von HTTP bietet WebDAV eine ausgezeichnete Integration in die bestehende IT-Infrastruktur großer Verwaltungen. So stellen die internen Sicherheits- und Schutzmechanismen, z. B. der Einsatz von Firewalls, keine Hürde für die Verwendung von WebDAV als Protokoll für das DMS-System dar.

3.3 Open-Source DMS-Systeme

Im folgenden sollen zwei Open-Source DMS-Systeme vorgestellt und hinsichtlich ihrer Verwendungsmöglichkeit für das RAfEG-Softwaresystems miteinander verglichen werden. Es handelt sich dabei um Jakarta Slide und ein System bestehend aus Apache Webserver mit zugehörigem Subversion-Modul.

3.3.1 Jakarta Slide

Jakarta Slide ist ein Framework zur Entwicklung von Content Management Systemen, das unter der Apache Lizenz frei als Open-Source verfügbar ist. Es stellt kein fertiges Produkt dar, sondern kann als Basis für eigene DMS-Systeme verwendet werden. Jakarta Slide unterstützt WebDAV und seine Erweiterungen, z. B. DeltaV und DASL (DAV Searching and Locating) sehr gut. Jakarta Slide bietet eine Client-API an, die für eigene Projekte genutzt werden kann, und ist sehr flexibel bei der Abspeicherung seiner Daten. Neben einer Abspeicherung im lokalen Dateisystem können Datenbanken verschiedener Hersteller oder freier Projekte als Ablagesystem genutzt werden. Das WebDAV/DeltaV Interface zu Jakarta Slide ist als Servlet im Apache Tomcat-Servlet-Container realisiert, was die Anbindung einer LDAP-basierten Nutzerverwaltung ermöglicht. Jakarta Slide ist im Apache Jakarta Projekt integriert und aufgrund der Popularität dieses Projektes ist davon auszugehen, dass Jakarta Slide über viele Jahre aktiv weiterentwickelt werden wird.

Der große Nachteil von Jakarta Slide ist, dass es Versionen von Dateien immer als vollständige Kopien abspeichert, was die Datenmenge im Vergleich zu anderen Lösungen deutlich erhöht.

3.3.2 Subversion

Subversion ist ein Versions-Management System ähnlich CVS (Concurrent Versioning System). Es steht unter einer Open-Source Lizenz frei zur Verfügung. Subversion kann als Kommunikations-Protokoll WebDAV/DeltaV benutzen. Dazu stellt Subversion ein Modul für den Apache-Webserver zur Verfügung, die eine Einbindung von Subversion in den

Apache-Webserver erlaubt. Durch die Nutzung des Apache-Webserver kann zusätzlich sehr einfach eine Anbindung an eine auf LDAP-basierende Nutzerverwaltung realisiert werden. Die Abspeicherung der Daten kann, wie bei Jakarta Slide, sehr flexibel gestaltet werden. Neben einer Abspeicherung im lokalen Dateisystem kann auch eine Abspeicherung der Daten in Datenbanken erfolgen. Subversion implementiert sehr leistungsfähige Algorithmen, die bei Änderungen sowohl von Text, als auch von Binärdaten nur die Änderungen abspeichern. Die Popularität von Subversion ist in letzter Zeit sehr angestiegen und es ist davon auszugehen, dass es über einen sehr langen Zeitraum aktiv gepflegt werden wird.

3.4 Auswahl des Systems

Da bei der Abspeicherung von Daten in der öffentlichen Verwaltung sehr große Datenmengen, insbesondere durch PDF-Formulare, Bilder und GIS-Daten, zu erwarten sind, die oft in mehreren Versionen abgespeichert werden, wurde ein auf Subversion basierendes System als DMS-Komponente ausgewählt. Nichtsdestotrotz kann das DMS-System aufgrund der Verwendung von offenen Schnittstellen ausgetauscht und so in Zukunft auch ein anderes System, z. B. Jakarta Slide, genutzt werden sofern sich entsprechende Anforderungen ergeben.

4 Die DMS-Komponente des RAfEG-Softwaresystems

Das RAfEG-Softwaresystem unterstützt Behörden bei der Abarbeitung von Verwaltungsvorgängen in Form von Workflows. Ein Workflow ist dabei ein Verwaltungsvorgang, der in der Form modelliert ist, dass er von einem Rechnersystem abgearbeitet werden kann. Ein in der Abarbeitung befindlicher Workflow wird dabei im Folgenden als Prozess bezeichnet. Die zentrale Komponente des RAfEG-Softwaresystems nutzt zur Abarbeitung der modellierten Workflows ein Workflow Managementsystem, das die Workflowbeschreibungen interpretiert und den Nutzern die anstehenden Aufgaben, mit allen zugehörigen Dokumenten anzeigt, um eine Bearbeitung zu ermöglichen. Die logische Sicht auf das RAfEG-Softwaresystem mit angebundenem DMS ist in Abbildung 1 dargestellt. Ein An-

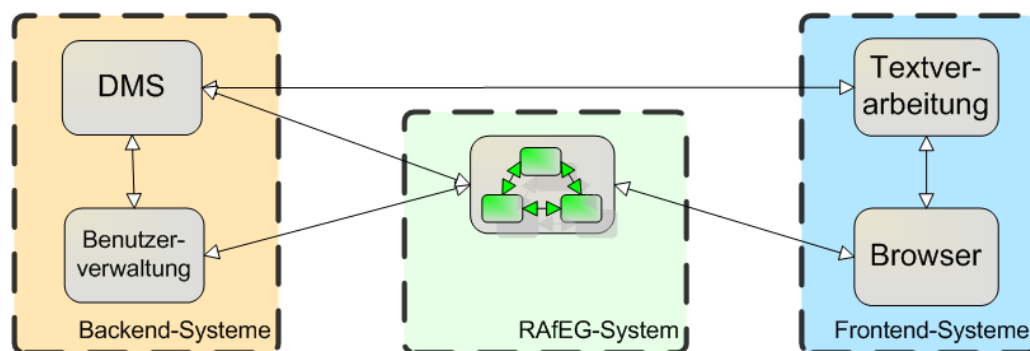


Abbildung 1: Logische Sicht auf die Nutzung des DMS

wender des Systems kommuniziert mit dem RAfEG-Softwaresystem über einen Browser. Das RAfEG-Softwaresystem ist dabei so konzipiert, dass die Interaktion mit dem Nutzer auch über andere Medien geschehen kann. Durch die Nutzung eines Browsers ist jedoch die Unabhängigkeit von Betriebssystem und speziellen Systemumgebungen gegeben. Der

Nutzer ruft über den Browser seine Aufgabenliste ab. Diese besteht aus einer Beschreibung aller anstehenden Aufgaben mit Ihren Dringlichkeiten und dem aktuellen Bearbeitungszustand. Entscheidet sich der Nutzer für eine Aufgabe, erhält er Details bezüglich der Abarbeitung und eine Übersicht über die Dokumente, die mit dieser Aufgabe verknüpft sind. Diese Dokumente werden vom RAfEG-Softwaresystem in einem zugehörigen Dokumentenmanagementsystem abgelegt, auf das das System mit den Nutzerdaten des eingeloggten Nutzers zugreifen kann. Dazu überprüft es über die Benutzerverwaltung, ob der Nutzer auf das Dokumentenmanagementsystem zugreifen darf und ob er das Dokument lesen bzw. später dann auch schreiben darf. Darf er es lesen, wird ein Link auf das Dokument in seinem Browser angezeigt und dieser kann je nach Dateityp mit einer entsprechenden Textverarbeitung geöffnet und bearbeitet werden. Die Art des Dokumentes ist für das RAfEG-Softwaresystem dabei nicht von Bedeutung. Nach dem Ändern eines Dokumentes kann dieses als neue Version in das DMS zurückgeschrieben werden. Dies geschieht über einen HTTP-Uploadmechanismus.

4.1 Interner Aufbau des DMS

Als DMS wird, wie in Abschnitt 3.3 dargelegt, eine Kombination aus Apache Webserver und Subversion genutzt. Der Aufbau ist in Abbildung 2 dargestellt.

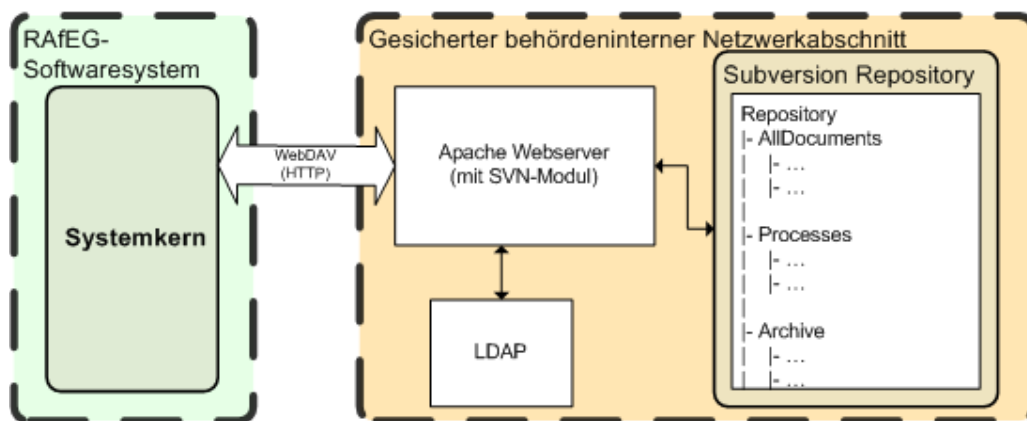


Abbildung 2: Struktur des Dokumentenmanagementsystems

Der Zugriff auf das realisierte DMS geschieht über HTTP/1.1. Der genutzte Apache Webserver wird mit SVN-Modul eingesetzt, damit er mit einem Subversion Repository kommunizieren kann. Greift ein Nutzer lesend auf das DMS zu, so werden als erste seine Rechte über die Rechteverwaltung des Apache Webservers geprüft. Dieser kann dazu auf das innerhalb des RAfEG-Softwaresystems eingesetzte LDAP-Directory zugreifen. Wird dem Nutzer der Zugriff gewährt, kann er das angeforderte Dokument lesen. Beim Schreiben erfolgt die Autorisierung analog zum lesenden Zugriff. Hier wird dann für jede Änderung an einem Dokument eine neue Version dieses Dokuments mit Informationen über den ändernden Nutzer abgelegt.

4.2 Anbindung des DMS an die RAfEG-Software

Das DMS für die Unterstützung der elektronischen Verwaltungsvorgänge ist als externe Komponente des RAfEG-Softwaresystems angebunden. Diese Herangehensweise bietet

die Möglichkeit, Software, die nicht zum Kern des RAfEG-Softwaresystems gehört, leicht durch eigene Implementierungen ersetzen zu können. Abbildung 3 zeigt einen Gesamtüberblick des RAfEG-Softwaresystem. Das RAfEG-Softwaresystem ist als Baukastensystem

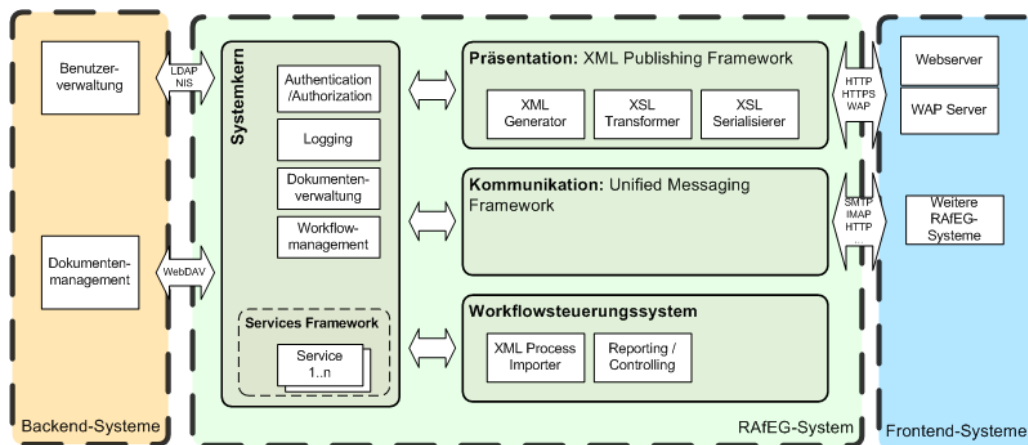


Abbildung 3: Gesamtansicht auf das RAfEG-Softwaresystem unter dem Gesichtspunkt der DMS-Anbindung

tem konzipiert und besteht aus vier Hauptkomponenten. Das sind der RAfEG-Kern, die Präsentationskomponente, die Kommunikationskomponente und die Workflowsystemkomponente. Eine detaillierte Darstellung der RAfEG-Softwarearchitektur und des zugrundeliegenden Baukastensystems ist in [BHK⁺06] zu finden. Der RAfEG-Kern verbindet das RAfEG-Softwaresystem mit benötigten externen Anwendungen, die vorimplementiert (z. B. das DMS) bzw. durch Drittanbieter (z. B. die auf LDAP basierende Nutzerverwaltung) realisiert sind, bei der Einführung aber auch durch vorhandene Lösungen durch Nutzung offener Schnittstellen angebunden werden können. Die Präsentationskomponente dient der Anbindung der Nutzerschnittstelle an das RAfEG-Softwaresystem und die Kommunikationskomponente der Übermittlung von Daten zwischen mehreren RAfEG-Softwaresystemen. In der Workflowsystemkomponente befindet sich das eingesetzte Workflow Managementsystem und die implementierten Zugriffsfunktionen.

Eine zentrale Rolle bei der Abarbeitung von Verwaltungsvorgängen durch das RAfEG-Softwaresystem spielen die im DMS abgelegten Prozessmappen. Eine Prozessmappe bezeichnet einen Container, in dem alle Daten eines Verwaltungsvorgangs, der modelliert als Workflow durch das WfMS abgearbeitet wird, abgelegt werden. Dieser Container wird im RAfEG-Softwaresystem als Ordner im DMS abgelegt. Die Unterteilung der Prozessmappe erfolgt dabei in Dokumente und Aktennotizen. Dokumente beziehen sich dabei auf alle zum Verwaltungsvorgang zugehörigen und im Verlauf der Abarbeitung durch einen Nutzer benötigten Dokumente. Aktennotizen sind eine Möglichkeit innerhalb des RAfEG-Softwaresystems weiteren Bearbeitern des Verwaltungsvorgangs Notizen über Abweichungen bzw. wichtige Änderungen verfügbar zu machen. Der genaue Aufbau des DMS wird im folgenden Abschnitt detailliert vorgestellt. Der Begriff des Repositories bezeichnet den Hauptordner im DMS.

4.3 Der Aufbau des Repositories

In Abschnitt 4.1 wurde beschrieben, dass als DMS ein Apache Webserver mit WebDAV-Modul genutzt wird und als Datencontainer ein Subversion Repository zum Einsatz

kommt. Die Verzeichnisstruktur innerhalb des DMS besteht aus drei primären Ordnern für die Dokumente. Im ersten Ordner (Vorlagenordner) werden die Vorlagen für Dokumente, Grafiken und Webformulare der geladenen Workflowdefinitionen, die als Workflows modellierten Verwaltungsvorgänge, abgelegt. Der zweite Ordner (Prozessmappenordner) untergliedert sich in die für alle gestarteten Prozesse angelegten Prozessmappen. Im dritten Ordner (Archivordner) werden die Teile der Prozessmappen von abgeschlossenen Prozessen abgelegt, die für einen späteren Zugriff gesichert werden müssen (z. B. die ausgefüllten Formulare und die Aktennotizen). Die Verzeichnisstruktur der Daten im DMS ist in Abbildung 4 dargestellt und wird im Folgenden näher beschrieben.

Der Vorlagenordner Im Vorlagenordner, intern unter dem Namen „AllDocuments“ angelegt, befindet sich mindestens das Verzeichnis „docs“, in dem sich allgemeine Formulare und Dokumente befinden, die nicht an spezielle Verwaltungsvorgänge gebunden sind. Diese Dokumente müssen manuell, durch einen Administrator, im Repository abgelegt werden. Zusätzlich wird für jedes geladene Prozesspaket ein Ordner mit den für die darin definierten Workflows relevanten Dokumenten angelegt. Prozesspakete beinhalten eine oder mehrere Workflowdefinitionen, die logisch zusammenhängen. Ein Workflow, der intern aus einer Menge an Unterworkflows besteht kann z. B. mit diesen in einem Paket modelliert werden. In der Abbildung ist der Ordner „pfv“ ein solches Paketverzeichnis für das Planfeststellungsverfahrenpaket. Die Ordner der Prozesspakete sind:

Ordner „docs“ Der Ordner enthält die vordefinierten Formulare aller Workflows des Pakets, z. B. Anträge, Gesetzestexte. Die Dokumente können sich dabei mit denen im allgemeinen „docs“-Ordner überschneiden. Für die später gestarteten Prozesse wird bei mehreren Versionen später eine durch den Administrator festgelegt.

Ordner „svg“ Dieser Ordner beinhaltet die graphischen Darstellungen der modellierten Verwaltungsvorgänge des Pakets. Diese werden genutzt, um den Fortschritt der Abarbeitung zu veranschaulichen.

Ordner „webforms“ Der letzte Ordner enthält die Webformulare, für die Anzeige der einzelnen Aufgaben in der grafischen Oberfläche. Ein Webformular ist eine Beschreibung der modellierten Variablen der einzelnen Aufgaben eines Workflows und deren Bearbeitungsmöglichkeiten, z. B. nur lesen, lesen und schreiben.

Der Inhalt der Ordner „svg“ und „webforms“ kann durch das in [BDR06] beschriebene Tool zur Umwandlung der Workflowbeschreibungen aus AML erstellt werden. Die Vorlagendokumente müssen manuell hinzugefügt werden, da diese nicht in die Prozessbeschreibung integriert sind bzw. aus ihr

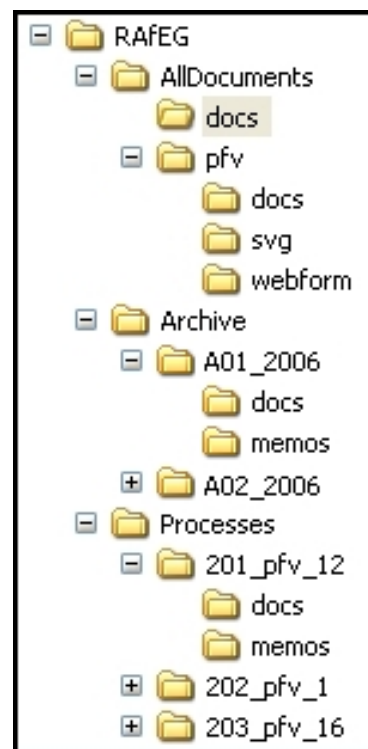


Abbildung 4: Verzeichnisstruktur im DMS

erzeugt werden können. Neben den Dokumenten werden im Paketvorlagenordner also zusätzlich eine graphische Darstellung der Workflowdefinitionen für eine spätere Veranschaulichung (Ordner „svg“) und die Webformulare zur Anzeige (Ordner „webform“) abgelegt.

Beim Einlesen/Ändern eines Prozesspaketes wird eine Mappingdatei angelegt. Diese bildet die in der Workflowdefinition unter einem generischen Namen angegebenen Dokumente auf real existierende Dokumente des Vorlagenordners ab. Ein Beispiel einer solchen Mappingdatei für das Planfeststellungsverfahren ist in Abbildung 5 zu finden. Das Erstellen geschieht dabei nicht automatisiert, sondern wird vom Administrator, der das Paket ändert/neu einliest angegeben. Die Nutzung generischer Namen in der Workflowdefinition und die Nutzung eines Mappings auf existierende Dokumente ist von großen Nutzen, da sich im E-Government Formulare, Antragsmuster usw. ändern können und die Größe der Workflows eine manuelle Änderung der Dokumente schwierig macht.

Der Prozessmappenordner Im Ordner für die Prozessmappen „Processes“ wird für jeden gestarteten Verwaltungsvorgang ein eigenes Unterverzeichnis mit dem Namen der beim Prozessstart erzeugten, eindeutigen Workflowidentifikation angelegt. In Abbildung 4 sind das z. B. „201_pfv_12“ und „202_pfv_1“. Die Namen setzen sich aus einer aufsteigenden fortlaufenden Nummer (201 und 202), dem Namen des Pakets, in dem der Prozess definiert ist (pfv) und dem Prozessidentifikator innerhalb des Pakets zusammen (12 bzw. 1). Die Unterverzeichnisse des Prozessmappenordners beinhalten einen Unterordner für alle Dokumente, die während der Abarbeitung gelesen oder geschrieben werden müssen (Ordner „docs“) und einen Unterordner für die Aktennotizen (Ordner „memos“).

Beim Start des Prozesses werden im Allgemeinen die benötigten Dokumente aus dem Vorlagenordner in den Ordner „docs“ der Prozessmappe kopiert. Dieser Vorgang geschieht über das Mapping, das im Vorlagenordner für die Prozesse des Paketes definiert wurde. Ein Unterschied besteht hier bei Unterprozessen. Im Falle des Starts eines Unterprozesses ist das Kopieren von Dokumenten aus dem Vorlagenordner nicht sinnvoll, denn es ist möglich, dass im Unterprozess ein Dokument weiterverarbeitet wird, das im übergeordneten Prozess bereits geändert wurde. Hier wird ausgehend vom aktuellen Prozess zuerst das Dokumentenverzeichnis als Teil der Prozessmappe des Elternprozesses, sofern vorhanden, nach genutzten Dokumenten durchsucht und in dem Falle, dass ein Dokument in diesem vorliegt, wird dieses in die Prozessmappe des aktuellen Prozesses kopiert. Ist der übergeordnete Prozess wieder ein Unterprozess, wird in der sich ergebenden Hierarchie solange nach einem Dokument gesucht, bis es entweder gefunden wurde oder ein Prozess keinen übergeordneten Prozess besitzt. Wurde ein Dokument in der Prozesshierarchie nicht gefunden, wird es aus dem Vorlagenverzeichnis kopiert. Bei der Beendigung eines Prozesses geschieht dieser Mechanismus wieder in der umgekehrten Richtung. Für alle Dokumente des beendeten Prozesses wird beim über-

```
<?xml version="1.0" encoding="UTF-8" ?>
<mapping>
  <files>
    <map realname="Antrag1.pdf"
         refname ="antrag_anhoehrung" />
    <map realname="Antrag2.txt"
         refname ="antrag_rueckleitung" />
    <map realname="Formular29.pdf"
         refname ="formular_betroffene" />
    <map realname="Muster20.pdf"
         refname ="muster_bekanntmachung" />
  </files>
  <directories/>
</mapping>
```

Abbildung 5: Mappingdatei zur Abbildung von generischen Dokumentennamen, die in der Workflowdefinition für die Aufgaben angegeben sind (refname), auf real existierende Dokumente (realname).

geordneten Prozess nachgeschaut, ob dieser das Dokument besitzt. Ist dies der Fall, wird es in diesen als neue Version zurückkopiert.

Der Archivordner Im Archivordner werden die Daten aller abgeschlossenen Verwaltungsvorgänge für eine eventuelle spätere Einsichtnahme abgelegt. Das Archivverzeichnis ist in Abbildung 4 dargestellt.

Statt unter dem eindeutigen Identifikator, der innerhalb der Prozessmappe genutzt wurde, wird ein abgeschlossenes Verfahren unter seinem Aktenzeichen (in der Abbildung „AZ_2006_01 “ bzw. „AZ_2006_02 “) gespeichert. Dazu wird der Ordner „docs“ kopiert und für den Ordner mit den Aktennotizen werden zum einen die beiden Dateien, die zum Anzeigen der Notizen einer Aufgabe benötigt werden, kopiert und eine neue Datei angelegt, in der die kompletten Memos des Prozesses und aller seiner Unterprozesse verfügbar sind. Die archivierten Dokumente werden im E-Government benötigt, da für alle Verwaltungsvorgänge eine Einsichtnahme über einen längeren Zeitraum möglich sein muss.

4.4 Implementierung der DMS-Komponente

Das RAfEG-Softwaresystem stützt sich auf die Enterprise Java Beans Architektur. Für eine nahtlose Integration der DMS-Komponente in das RAfEG-Softwaresystem wurde diese ebenfalls mit Hilfe von Enterprise Java Beans implementiert. Der Kern des DMSs innerhalb des RAfEG-Softwaresystems ist eine Stateful Session Bean namens „FileStoreBean “. Diese kommuniziert mit dem Subversion-Modul eines Apache Webservers. Dieses Modul führt die Befehle der FileStoreBean aus, verwaltet die ihm anvertrauten Daten und gibt entsprechende Statusinformationen zurück.

Sowohl die FileStoreBean als auch das Subversion-Modul sind über LDAP an die Rechteverwaltung des RAfEG-Softwaresystems gekoppelt, durch das sie Informationen zur Nutzerauthentifizierung beziehen.

4.4.1 Klassenhierarchie

Bei der Implementierung der DMS-Komponente wurde größter Wert auf deren Flexibilität gelegt, um zukünftige Erweiterung zu realisieren, bzw. den Austausch ganzer Komponenten zu ermöglichen. Eine Änderung ist z. B. notwendig, wenn ein anderes Protokoll als WebDAV unterstützt werden soll. Die Komponente des DMS, die IO-Operationen mit einem Abspeicherungssystem anbietet, ist die so genannte FileStoreBean, für die die Klassenhierarchie in Abbildung 6 dargestellt ist. Diese Stateful Session Bean bietet die folgenden Methoden zur Arbeit mit dem Apache Webserver an:

- Kopieren, Verschieben, Löschen von Dateien,
- Setzen bzw. Löschen Schreiblocks,
- Abfragen und Setzen des Security-Levels und
- Auslesen von Versionsinformationen.

Die FileStoreBean besitzt intern ein Objekt der Klasse WebDAVFileStore, an das sie die Anfragen weiterleitet. Diese Klasse implementiert das Interface FileStore. Durch weitere Implementierungen des Interfaces ist es möglich neben WebDAV weitere Methoden

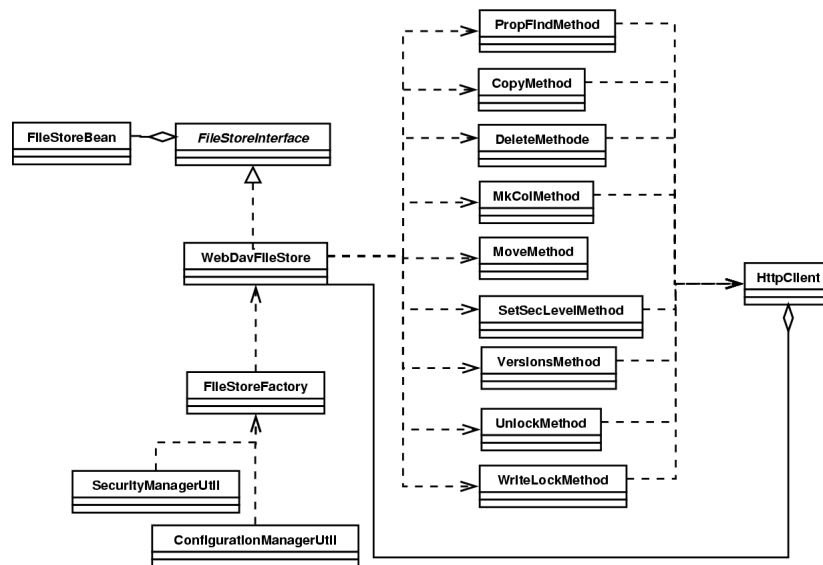


Abbildung 6: Ausschnitt aus der Klassenhierarchie zur Implementierung der DMS-Komponente über WebDAV

zur Datenablage zu unterstützen. Die Klasse FileStoreFactory erzeugt Objekte, die das Interface FileStore benutzen und trägt somit zur Flexibilität bei. Dabei benutzen Objekte der Klasse FileStoreFactory Informationen aus der Bean ConfigurationManagerUtil zum Auslesen von Konfigurationsinformationen, z. B. der URL des Repositories. Weiterhin benutzen Objekte der Klasse FileStoreFactory die Bean SecurityUtils zur Abfrage von Authentifizierungsdaten, die zum Verbindungsaufbau mit dem Apacheserver benötigt werden.

Zur Durchführung der IO-Operationen besitzt die Klasse WebDAVFileStore ein Objekt der Klasse HttpCilent, das vom Jakarta-Slide Projekt [Pro] zur Verfügung gestellt wird. Dieses Objekt hält die Verbindung zum Apache Server. Mit diesem Objekt arbeiten verschiedene Klassen, wie PropFindMethod, CopyMethod (siehe Klassen mit Endung „Method“ in Abbildung 6). Diese Klassen implementieren das Protokollhandling einzelner WebDAV-Anfragen.

4.4.2 Versionierung

Im Interface FileStore sind keine Methoden für das erstellen neuer Versionen einzelner Dateien zu finden. Das Anlegen einer Version erfolgt über die Autoversionierung. Dies bedeutet, dass bei jeder Änderung einer Datei, bzw. der zugehörigen Attribute, eine neue Version erstellt wird. Entsprechende Versionsstände lassen sich über Abfragen erhalten.

4.4.3 Sicherheitslevel

Sicherheitslevel werden über Attribute implementiert. Attribute sind Metadaten die zusätzlich zum Inhalt einer Datei abgespeichert und danach wieder abgefragt werden können.

5 Zusammenarbeit mit anderen RAfEG-Softwaresystemkomponenten für die verteilte Abarbeitung von Verwaltungsvorgängen

Die verteilte Abarbeitung von Verwaltungsvorgängen befasst sich vor allem mit der Herausforderung, Teile eines Vorgangs mit allen zugehörigen Dokumenten automatisiert an eine externe Institution (z. B. eine zweite Behörde oder ein Träger öffentlicher Belange) zu versenden und die Rückgabe automatisiert in den sendenden Vorgang zu integrieren. Die Bearbeitung der externen Teilaufgabe soll dabei völlig transparent für Nutzer des Systems geschehen. Für die Abarbeitung eines Verwaltungsvorgangs werden die Workflowkomponente (zum Abarbeiten der modellierten Vorgänge und für das Anstoßen der externen Teile), der RAfEG-Kern (als Schnittstelle zum DMS und zur Benutzerverwaltung) und die Kommunikationskomponente (für das Versenden von Daten) benötigt. Diese wurden in Abschnitt 4.2 kurz vorgestellt.

Beim Start eines externen Teils eines Verwaltungsvorgangs werden alle Dokumente, die in der externen Institution für die Abarbeitung notwendig sind, versendet. Hier ist die Angabe von einzelnen Dokumenten, wie auch die Angabe ganzer Dokumentenverzeichnisse möglich. Zusätzlich zu den Dokumenten muss der Teil der Mappingdatei aus dem Vorlagenordner kopiert werden, in dem die generischen Namen der versendeten Dokumente auf real existierende abgebildet werden. Dabei ist für eine ausschließlich automatisierte Abarbeitung ein einheitliches Namensschema bei der Definition der Workflows notwendig. Das heißt, die generischen Namen der Dokumente in der Definition des externen Teils müssen mit denen im lokalen Teil übereinstimmen. Wenn dies nicht der Fall ist, muss manuell in die Abarbeitung eingegriffen und die Mappingdatei vom Administrator vor der Bearbeitung der ersten Aufgabe angepasst werden. Die Definition der zu übermittelnden Dokumente geschieht in der Aktivität, die automatisiert abgearbeitet wird und die Komponente ausführt, die den externen Teil startet. Die Nutzung von erweiterten Attributen ermöglicht dabei eine einheitliche Modellierung, die mit der für lokale Aufgaben übereinstimmt.

In Tabelle 1 sind die einzelnen Schritte, mitsamt den ausführenden Komponenten des RAfEG-Softwaresystems, dargestellt, die beim Start und bei der Beendigung des externen Teiles eines Verwaltungsvorgangs ausgeführt werden.

6 Vergleich mit anderen Arbeiten auf diesem Gebiet

Dieser Abschnitt dient der Einordnung der erstellten Lösung in andere aktuelle Arbeiten auf dem Gebiet des Dokumentenmanagements. Dazu werden das DOMEA Konzept, SAGA und MoReq kurz vorgestellt und mit der Lösung verglichen.

Das DOMEA-Konzept [Koo05] bildet die Grundlage für die weitere Vorgehensweise des Bundes zum papierlosen Büro. Hier werden vor allem Konzepte und Kriterien eingeführt, die zu einer papierlosen Verwaltung führen sollen. Das Domea-Konzept baut auf eine dreistufige Hierarchie, die sich in Dokument, Vorgang und Akte gliedert. Im Gegensatz zu der in RAfEG implementierten Lösung und den erstellten Konzepten wird nicht auf die hierarchische Prozessabarbeitung und Sicherheitskriterien für die verteilte Abarbeitung von Vorgängen eingegangen.

Tabelle 1: Notwendige Schritte bei Start und Beendigung eines externen Teilvorgangs

Start des externen Teils eines Verwaltungsvorgangs	
Starten des Anwendungsagenten, der die externe Abarbeitung anstößt	Workflowkomponente
Auslesen der im Prozess definierten externen Dokumente	DMS-Komponente im RAfEG-Kern
Senden der Dokumente über OSCITransport	Kommunikationskomponente
Schreiben der Dokumente in das DMS	DMS-Komponente im RAfEG-Kern
Ende des externen Teils eines Verwaltungsvorgangs	
Information der Instanz an Beobachter (Beobachter wartet auf Dokumente)	Workflowkomponente
Auslesen der Dokumente	DMS-Komponente im RAfEG-Kern
Senden der Dokumente an Beobachter	Kommunikationskomponente
Speichern der geänderten Dokumente	DMS-Komponente im RAfEG-Kern
Beendigung der Aktivität des Unterprozesses	Workflowkomponente

SAGA [Bun03] (Standard und Architekturen für E-Government Anwendungen) ist eine Vorschriftensammlung des Ministeriums des Inneren, die „Standards, Verfahren, Methoden und auch Produkte für die moderne IT-Entwicklung für E-Government“ vorstellt. Dabei werden Standards in verschiedene Kategorien (empfohlen, obligatorisch, in Beobachtung und abgewiesen) eingeteilt. SAGA listet zusätzlich Applikationen, wie den „Government Site Builder“, auf. Diese Software ist ein Content Management System dessen DMS-Komponente Versionierung bei Veränderung, Write-Locks und die Nutzung von Metadaten für Dokumente anbietet, was eine ähnliche Funktionalität wie die DMS-Komponente des RAfEG-Softwaresystems bedeutet. Der „Government Site Builder“ wurde jedoch nicht für die behördenübergreifende Zusammenarbeit entwickelt.

Auf EU-Ebene bildet „MoReq“ (Model Requirements for the Management of Electronic Records) [IDA02] die Grundlage für Anforderungen an Dokumentenmanagement. MoReq ist dabei eine sehr detaillierte Liste mit Anforderungen, die auf dem ISO 15489 Standard basiert. Neben den Anforderungen an DMS werden in MoReq weiterhin auch Anforderungen an dokumentenbezogene Funktionen, z. B. Workflow, E-Mail und elektronische Signaturen gestellt. Die Spezifikation legt dabei lediglich Anforderungen fest und gibt keine implementierte Lösung, wie die in diesem Bericht beschriebene, vor.

7 Zusammenfassung

Dokumentenmanagement spielt bei der Abarbeitung von Verwaltungsvorgängen im E-Government eine zentrale Rolle. Das DMS innerhalb des RAfEG-Softwaresystems basiert auf offenen Schnittstellen und freier Open-Source Software, was einen einfachen Austausch und eine leichte Anpassung an sich ändernde Bedingungen ermöglicht. Die Verwaltung der Dokumente innerhalb des DMS ist mittels des eingeführten Systems der Prozessmappen auf die Bearbeitung von Verwaltungsvorgängen zugeschnitten. Innerhalb einer Prozessmappe werden alle Dokumente des zugehörigen Verwaltungsvorgangs gespeichert und für jede Änderung wird eine neue Version des Dokumentes erzeugt, was eine nachvollziehbare

Änderungshistorie schafft. Dadurch ist das erstellte Konzept vergleichbar mit einer sich ändernden Datei aller Informationen des Prozesses, mit dem Zusatz jeden Änderungszeitpunkt wiederherstellen zu können. Die Möglichkeit der Archivierung aller Dokumente eines Verwaltungsvorgangs ermöglicht die spätere Einsichtnahme und wird im E-Government zwingend benötigt. Durch die Vergabe von Sicherheitsleveln für die Dokumente im DMS ist es zudem möglich, speziell für die Abarbeitung von behördenübergreifenden Verwaltungsvorgängen, die Sicherheitsmechanismen an die Bedürfnisse der Verwaltungen anzupassen.

8 Danksagung

Wir danken Daniel Beer und Steffen Höhne für die Hilfe und Unterstützung bei den Konzeptions- und Implementierungsarbeiten. Besonders möchten wir auch unseren Konsortiumpartnern innerhalb des BMBF Projektes, dem Institut für Wirtschaftsinformatik (IWi) im Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI), der NHConsult GmbH und dem Regierungspräsidium Leipzig für die gute Zusammenarbeit danken.

Literatur

- [BDR06] Daniel Beer, Jörg Dümmler, and Gudula Rünger. *Automatische Erzeugung von Workflowbeschreibungen im XPDL-Format aus ereignisgesteuerten Prozessketten*. Chemnitzer Informatik-Berichte CSR-06-11, Technische Universität Chemnitz, Chemnitz, 2006.
- [BHK⁺06] Daniel Beer, Steffen Höhne, Raphael Kunis, Gudula Rünger, and Michael Voigt. *RAfEG - Eine Open Source basierte Architektur für die Abarbeitung von Verwaltungsprozessen im E-Government*. Chemnitzer Informatik Berichte CSR-06-03, Technische Universität, Chemnitz, 2006.
- [BHRV05] Daniel Beer, Steffen Höhne, Gudula Rünger, and Michael Voigt. *Software- und Kriterienkatalog zu RAfEG - Referenzarchitektur für E-Government*. Chemnitzer Informatik-Berichte CSR-05-01, Technische Universität, Chemnitz, 2005.
- [Bun03] SAGA: Standards und Architekturen für E-Government-Anwendungen, 2003.
- [DMW] DMWare. Open Document Management API (ODMA).
- [Doc] Document Management Alliance. DMA 1.0-7 Specification.
- [GWF⁺99] Y. Goland, E. Whitehead, A. Faizi, S. Carter, and D. Jensen. RFC 2518: HTTP Extensions for Distributed Authoring – WEBDAV, 1999. Status: Proposed Standard.
- [IDA02] IDA Programme of the European Commission. Model Requirements For The Management Of Electronic Records - Specification, 2002.

- [JKR06] Sebastian Junge, Raphael Kunis, and Gudula Rünger. *In Vorbereitung: Ein Sicherheitskonzept für die verteilte Abarbeitung von Verwaltungsvorgängen im E-Government*. Chemnitzer informatikberichte, Technische Universität Chemnitz, Chemnitz, 2006.
- [Koo05] Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung. DOMEA - Dokumentenmanagement und elektronische Archivierung im IT-gestützten Geschäftsgang, Organisationskonzept, 2005.
- [Net02] Network Working Group. RFC 3253: Versioning Extensions to WebDAV, 2002. Status: Standards Track.
- [Pro] Apache Jakarta Project. Jakarta slide.

Chemnitzer Informatik-Berichte

In der Reihe der Chemnitzer Informatik-Berichte sind folgende Berichte erschienen:

- CSR-02-01** Andrea Sieber, Werner Dilger, Theorie und Praxis des Software Engineering, Oktober 2001, Chemnitz
- CSR-02-02** Tomasz Jurdzinski, Mirosław Kutylowski, Jan Zatoptionski, Energy-Efficient Size Approximation of Radio Networks with no Collision Detection, Februar 2002, Chemnitz
- CSR-02-03** Tomasz Jurdzinski, Mirosław Kutylowski, Jan Zatoptionski, Efficient Algorithms for Leader Election in Radio Networks, Februar 2002, Chemnitz
- CSR-02-04** Tomasz Jurdzinski, Mirosław Kutylowski, Jan Zatoptionski, Weak Communication in Radio Networks, Februar 2002, Chemnitz
- CSR-03-01** Amin Coja-Oghlan, Andreas Goerdt, André Lanka, Frank Schädlich, Certifying Unsatisfiability of Random $2k$ -SAT Formulas using Approximation Techniques, Februar 2003, Chemnitz
- CSR-03-02** M. Randrianarivony, G. Brunnett, Well behaved mesh generation for parameterized surfaces from IGES files, März 2003, Chemnitz
- CSR-03-03** Optimizing MPI Collective Communication by Orthogonal Structures, Matthias Kühnemann, Thomas Rauber, Gudula Rünger, September 2003, Chemnitz
- CSR-03-04** Daniel Balkanski, Mario Trams, Wolfgang Rehm, Heterogeneous Computing With MPICH/Madeleine and PACX MPI: a Critical Comparison, Dezember 2003, Chemnitz
- CSR-03-05** Frank Mietke, Rene Grabner, Torsten Mehlan, Optimization of Message Passing Libraries - Two Examples, Dezember 2003, Chemnitz
- CSR-04-01** Karsten Hilbert, Guido Brunnett, A Hybrid LOD Based Rendering Approach for Dynamic Scenes, Januar 2004, Chemnitz
- CSR-04-02** Petr Kroha, Ricardo Baeza-Yates, Classification of Stock Exchange News, November 2004, Chemnitz
- CSR-04-03** Torsten Hoefler, Torsten Mehlan, Frank Mietke, Wolfgang Rehm, A Survey of Barrier Algorithms for Coarse Grained Supercomputers, Dezember 2004, Chemnitz
- CSR-04-04** Torsten Hoefler, Wolfgang Rehm, A Meta Analysis of Gigabit Ethernet over Copper Solutions for Cluster-Networking, Dezember 2004, Chemnitz
- CSR-04-05** Christian Siebert, Wolfgang Rehm, One-sided Mutual Exclusion A new Approach to Mutual Exclusion Primitives, Dezember 2004, Chemnitz

Chemnitzer Informatik-Berichte

- CSR-05-01** Daniel Beer, Steffen Höhne, Gudula Rünger, Michael Voigt, Software- und Kriterienkatalog zu RAfEG - Referenzarchitektur für E-Government, Januar 2005, Chemnitz
- CSR-05-02** David Brunner, Guido Brunnett, An Extended Concept of Voxel Neighborhoods for Correct Thinning in Mesh Segmentation, März 2005, Chemnitz
- CSR-05-03** Wolfgang Rehm (Ed.), Kommunikation in Clusterrechnern und Clusterverbundsystemen, Tagungsband zum 1. Workshop, Dezember 2005, Chemnitz
- CSR-05-04** Andreas Goerdt, Higher type recursive program schemes and the nested pushdown automaton, Dezember 2005, Chemnitz
- CSR-05-05** Amin Coja-Oghlan, Andreas Goerdt, André Lanka, Spectral Partitioning of Random Graphs with Given Expected Degrees, Dezember 2005, Chemnitz
- CSR-06-01** Wassil Dimitrow, Mathias Sporer, Wolfram Hardt, UML basierte Zeitmodellierung für eingebettete Echtzeitsysteme, Februar 2006, Chemnitz
- CSR-06-02** Mario Lorenz, Guido Brunnett, Optimized Visualization for Tiled Displays, März 2006, Chemnitz
- CSR-06-03** D. Beer, S. Höhne, R. Kunis, G. Rünger, M. Voigt, RAfEG - Eine Open Source basierte Architektur für die Abarbeitung von Verwaltungsprozessen im E-Government, April 2006, Chemnitz
- CSR-06-04** Michael Kämpf, Probleme der Tourenbildung, Mai 2006, Chemnitz
- CSR-06-06** Torsten Hoefler, Mirko Reinhardt, Torsten Mehlan, Frank Mietke, Wolfgang Rehm, Low Overhead Ethernet Communication for Open MPI on Linux Clusters, Juli 2006, Chemnitz
- CSR-06-07** Karsten Hilbert, Guido Brunnett, A Texture-Based Appearance Preserving Level of Detail Algorithm for Real-time Rendering of High Quality Images, August 2006, Chemnitz
- CSR-06-08** David Brunner, Guido Brunnett, Robin Strand, A High-Performance Parallel Thinning Approach Using a Non-Cubic Grid Structure, September 2006, Chemnitz
- CSR-06-09** El-Ashry, Peter Köchel, Sebastian Schüler, On Models and Solutions for the Allocation of Transportation Resources in Hub-and-Spoke Systems, September 2006, Chemnitz
- CSR-06-10** Raphael Kunis, Gudula Rünger, Michael Schwind, Dokumentenmanagement für Verwaltungsvorgänge im E-Government, Oktober 2006, Chemnitz