



Offene Integration von GEO- und Planungsdiensten in ein Internet-Contentportal

Diplomarbeit

Universität Hamburg Fachbereich Informatik

von

Niels Hoffmann

Tienrade 11 21031 Hamburg Matrikelnummer 4724342

eingereicht bei

Prof. Dr.-Ing. Karl Kaiser
(Erstbetreuer)
Arbeitsbereich technische Informatiksysteme (TIS)
Universität Hamburg, Fachbereich Informatik

und

Prof. Dr. Joachim W. Schmidt
(Zweitbetreuer)
Arbeitsbereich Softwaresysteme (STS)
Technische Universität Hamburg Harburg

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst und dazu keine anderen als die Angegebenen Quellen und Hilfsmittel verwendet habe.
Hamburg, den 29. Juni 2005
Niels Hoffmann

Inhaltsverzeichnis

Kapitel 1 I	Einleitung	9
1.1 Ziel	e der Arbeit.	10
1.2 Stru	ıktur der Arbeit	11
1.3 Dan	ıksagung	11
Kapitel 2 I	nternetportal "Metropolregion Hamburg"	13
2.1 Hist	torie des Portals	13
2.2 Inha	alte und Ziele des Portals	14
2.2.1	Aufgaben und Funktion des Portals	14
2.2.2	eGovernment	15
2.2.3	Redaktionelle Prozesse	16
2.2.4	Dienst-, Datensyndikation und -integration	16
Kapitel 3	Content Management System (CMS)	19
3.1 Auf	gaben und Einsatz von CMS	19
3.2 CM	S Funktionsumfang.	20
Kapitel 4	Konzepte zur Integration von Inhalten in Web-Portale	25
4.1 Inte	grationstopologien	26
4.1.1	Punkt zu Punkt	26
4.1.2	zentrale Verteiler	27
4.2 Inte	grationsstrategien	28
4.3 Date	enintegration in CMS	31
4.3.1	Alternativen der Datenintegration	32
4.3.2	Strukturelle Komponenten für die Datenintegration	40
4.3.3	Datensynchronisation	45
4.3.	3.1 Pessimistische Verfahren	46
4.3.	3.2 Optimistische Verfahren	48
4.3.4	Datenredundanzen vermeiden.	50
4.4 Die	nstintegration in CMS	52
4.4.1	Alternativen der Dienstintegration	52
112	Diensteunchronication und Dienstredundanz	58

4.4.3	Dienstintegration in CMS	59
Kapitel 5	Geografisches Informationssystem (GIS)	61
5.1 Umf	ang und Aufgaben eines GIS	62
5.2 Eins	atzszenarien	63
5.3 Erfa	ssung und Pflege von Geodaten und -Informationen	63
5.4 Beis	pielsysteme	66
5.5 Ope	n Geospatial Consortium (OGC TM)	67
5.5.1	Ziele des Konsortiums	67
5.5.2	Geography Markup Language (GML)	68
5.5.3	OGC TM konforme Web-Dienste	69
5.5.3	3.1 Web-Map-Service	69
5.5.3	3.2 Web-Feature-Service	71
Kapitel 6 In	ntegration von CMS- und GIS-Diensten	73
6.1 Löst	ungsansatz	74
6.2 Syst	emanforderungen	75
6.2.1	Standards und Kompatibilität	76
6.2.2	Systemarchitektur	77
6.2.3	Use Cases	80
6.2.4	Systemdesign	83
6.2.4	4.1 Schnittstelle zum GIS	90
6.2.4	4.2 Schnittstelle zum CMS	92
6.3 Anw	vendungsbeispiel "Gemeinde Seevetal"	93
6.3.1	Systemaufbau	95
6.3.2	Konfiguration und Anpassungen.	97
6.3.3	Einbindung in CoreMedia©	99
Kapitel 7 Z	usammenfassung	103
Kapitel 8 A	usblick	107
Literaturver	zeichnis	109

Abbildungsverzeichnis

Abbildung 1: Portal der MRH und Contentpartner	17
Abbildung 2: Syndikation von Diensten und Daten in das MRH Portal	18
Abbildung 3: Integrationstopologie – Punkt zu Punkt	27
Abbildung 4: Integratinstopologie - zentraler Verteiler	28
Abbildung 5: One-Shot Integrationsstrategie	29
Abbildung 6: balancierte binäre Integrationsstrategie	30
Abbildung 7: zwei Beispiele für binäre gewichtete Integrationsstrategien	30
Abbildung 8: Datenintegration über Replikation in ein CMS	33
Abbildung 9: Datenintegration über Schnittstellen in ein CMS	36
Abbildung 10: Datenintegration über Föderation in ein CMS	38
Abbildung 11: kapselnde Schnittstellen zwischen Quellsystemen und CMS	41
Abbildung 12: Zielkonflikte bei der Datensynchronisation aus [Thorsten Reincke 1999]	45
Abbildung 13: Unterbrechung der Verbindung bei Verweisen	51
Abbildung 14: Integration von Diensten als Kopie	53
Abbildung 15: Integration von Diensten als Einbindung	54
Abbildung 16: konkrete Alternativen bei der Dienstintegration	56
Abbildung 17: redundant ausgelegte Dienste mit loadbalancing	58
Abbildung 18: Gesamtarchitektur des GIS-Integrationssystems	78
Abbildung 19: Austauschprotokolle im Gesamtsystem des GIS-Integrationssystems	79
Abbildung 20: Use Case Diagramm Benutzer – Karteninteraktion	80
Abbildung 21: Use Case Diagramm Benutzer – Annotationen	81
Abbildung 22: Use Case Diagramm Redakteur – WebMaps	82
Abbildung 23: Use Case Diagramm Redakteur – Annotationen	83
Abbildung 24: Aktivitätsdiagramm Redakteur - Verwaltung der WebMaps im GIS-Integrator	84
Abbildung 25: Sequenzdiagramm WebMap definieren	85
Abbildung 26: Aktivitätsdiagramm Redakteur - Annotierung der WebMaps im GIS-Integrator	86
Abbildung 27: Sequenzdiagramm WebMap anzeigen	87
Abbildung 28: Sequenzdiagramm Annotation freigeben	88
Abbildung 29: Löschen einer Annotation	89
Abbildung 30: Überblick über die GIS Schnittstelle	90
Abbildung 31: Überblick über die CMS Schnittstelle	93
Abbildung 32: Struktur der GIS Integration in die Website der Gemeinde Seevetal	96
Abbildung 33: Erweiterung des CoreMedia Dokumentmodells	100

Kapitel 1

Einleitung

Die folgende Arbeit beschäftigt sich mit zwei Themen. Allgemein stehen Konzepte zur Integration unterschiedlicher Informationsquellen im Mittelpunkt der Arbeit mit dem Ziel, diese zentral erreichbar und verwaltbar zu machen. Speziell geht es dann um das Thema, geografische Daten über Webportale zugänglich zu machen und entsprechende Interaktionsmöglichkeiten bereitzustellen. Der Schwerpunkt der Arbeit liegt auf dem zweiten Themenbereich, der unter Berücksichtigung des ersten bearbeitet wird.

Attraktive Internetportale müssen für die Aktualität ihrer Inhalte sorgen. Dies wird überwiegend mit einer eigenen Redaktion gewährleistet, die dafür ein Content Management System einsetzt. Zusätzlich werden Inhalte von anderen Anbietern oder von anderen Internetportalen bezogen und in das Portal integriert. Diese Fremdinhalte müssen gesondert behandelt werden, um ihre Konsistenz und Aktualität zu sichern, ihre Lebenszeiten festzulegen und Redundanzen zu vermeiden. Es muss also ein komplexes Austauschkonzept für diese Inhalte definiert werden.

Dabei sind nicht nur Daten, beispielsweise über Veranstaltungen, auszutauschen und zu integrieren, sondern auch komplexe Dienste, etwa eine Hotelbelegungsauskunft mit Buchungsmöglichkeit, die über mehrere Portale hinweg realisiert ist. Dabei ist es sicher nicht zu empfehlen, vergleichbare Dienste über die selben Daten auf mehreren Portalen parallel zu implementieren und diese dann später miteinander zu synchronisieren. Vielmehr ist es vorzuziehen, derartige Dienste nur einmal zu realisieren und über entsprechende Schnittstellen anzubieten, damit unterschiedliche Portale mit den notwendigen Informationen versorgt werden können. Ebenso wie bei der Datenintegration treten auch bei der Dienstintegration Integritäts-, Aktualitäts- und Konsistenzprobleme auf.

Raum- und ortsbezogene Visualisierungen spielen eine zentrale Rolle bei der Vermittlung von Informationen. Für den Betrachter von strukturierten Plänen ist es einfacher, sich Strecken,

10 Kapitel 1 Einleitung

Abmessungen und Verhältnisse vorzustellen und sich so ein realitätsgetreueres Bild der tatsächlichen geografischen Gegebenheiten zu machen, als für einen Leser reiner Textinformation. Da Text Raum für unterschiedliche Interpretationen lässt, ist es sinnvoll, diesen gegebenenfalls mit raum- und ortsbezogenen Visualisierungen zu unterstützen und so für unterschiedliche Betrachter eine möglichst eindeutige Interpretation zu ermöglichen.

Dies gilt selbstverständlich nicht nur für gedruckte Informationen wie Stadtpläne, Lagepläne, Flurpläne usw., sondern auch für die Informationskommunikation im Intra- oder Internet. Da Raum- und Ortsdaten (Geodaten) bei verschiedensten Behörden, Institutionen und Firmen sowieso in digitaler Form zur Verfügung steht, ist es besonders wünschenswert, diese Daten teilweise über das Intra- oder Internet publizieren und mit Zusatzinformationen anreichern zu können.

Texte, Geodaten und Zusatzinformationen liegen meist verteilt über unterschiedliche Einrichtungen, Systeme und Strukturen vor. Aus diesem Grund ist es notwendig, sich Gedanken über sinnvolle Integrationsmaßnahmen zu machen, durch die eine einfache Verwaltung und eine gemeinsame Publikation dieser verteilten Inhalte auf verschiedenen Web-Plattformen ermöglicht wird.

1.1 Ziele der Arbeit

In dieser Arbeit wird ein Konzept erarbeitet und umgesetzt, welches es erlaubt, dezentral vorhandene geografische Daten mehreren Webportalen über eine redaktionelle Schnittstelle zur Verfügung zu stellen. Portalredakteuren soll es so möglich sein, Karten über eine komfortable Schnittstelle ortsbezogen mit weiteren redaktionellen Inhalten anzureichern und in ihre Webseiten einzubinden. Für die Benutzer von Websites soll zusätzlich die Möglichkeit geschaffen werden, mit Karten zu interagieren und bestimmte Positionen zu annotieren.

Kern des Interesses der vorliegenden Arbeit sind also geografische Daten, die über Webportale verfügbar gemacht werden sollen. Derartige Redaktionsvorgänge sollen einfach und ohne besondere Vorkenntnisse beim Umgang mit geografischen Daten durchgeführt werden können. Dazu müssen Redakteure die Geodaten als Karten über das eingesetzte Content Management System (CMS) auf Websites platzieren können. Zusätzlich müssen sie die Karten ortsbezogen mit weiteren redaktionellen Inhalten anreichern können, die dann auf der bearbeiteten Website sichtbar und abrufbar sind. Als letztes wird es für Benutzer der Websites möglich sein, ortsbezogen die bereitgestellten Karten zu annotieren.

Um all dies zu realisieren, ist es notwendig, Verfahren zu schaffen und zu nutzen, durch die Geodaten von einem CMS aus referenziert und als grafische Visualisierung (Karten) mit im CMS gepflegten Zusatzinformationen veröffentlicht werden können. Außerdem muss eine Schnittstelle geschaffen werden, die es erlaubt, CMS-Inhalte an bestimmte Bereiche der Karten zu binden, um diese später zu visualisieren und zugreifbar zu machen.

1.2 Struktur der Arbeit

1.2 Struktur der Arbeit

Die Arbeit stellt in den ersten drei Kapiteln die Motivation des Projektes heraus und diskutiert Basiskomponenten, die zur Erreichung der Zielsetzung benötigt werden. Die Motivationsgrundlage und Arbeitsumgebung bildet das Webportal der Metropolregion Hamburg und der daran partizipierenden Gemeinden und Kommunen. Diese verfügen über Webportale, auf denen die bereitgestellten Geodaten visualisiert werden sollen. Die dafür notwendigen Kernkomponenten sind Content Management Systeme (CMS) und Geografische Informationssysteme (GIS). In den nachfolgenden Kapiteln wird detaillierter auf die Konzeption, Planung und Umsetzung des Projektes mit der Gemeinde Seevetal eingegangen. Die vorliegende Arbeit ist eingebunden in ein Pilotprojekt, in welchem die Möglichkeiten der Integration von geografischen GIS-Diensten mit multimedialen CMS-Diensten zum Zwecke der Realisierung leistungsfähiger Webportale untersucht werden.

1.3 Danksagung

Ich bedanke mich herzlich bei Herrn Prof. Karl Kaiser vom Fachbereich Informatik der Universität Hamburg, Arbeitsbereich Technische Informatiksysteme, für die Betreuung meiner Arbeit und für die Unterstützung. Herrn Prof. Dr. Joachim W. Schmidt, Arbeitsbereich Softwaresysteme der TU Hamburg-Harburg, danke ich sehr für die Zweitbetreuung und für den Einsatz und das große Interesse, mit dem er das gesamte Projekt und diese Arbeit unterstützt hat. Weiterhin möchte ich Herrn Rainer Marrone, Oberingenieur am Arbeitsbereich STS der TU Hamburg-Harburg, bedanken, der jederzeit für Fragen und technische Auskünfte zur Verfügung stand. Ein besonderer Dank gilt auch Herrn Dr. Kai- Uwe Krause, Arbeitsbereich Stadt-, Regional- und Umweltplanung der TU-Harburg, der mit Prof. Schmidt maßgeblich an der Initiierung des Projektes beteiligt war, für die Beantwortung von Fachfragen und die Bereitstellung der benötigten Hard- und Software. Danken möchte ich auch den Mitarbeitern des Rathauses Seevetal, hier besonders Herrn Partho Banerjea und Frau Katrin Matthies, die dieses Projekt unterstützt und begleitet haben. Meinem Kommilitonen Herrn Olaf Bauer gilt ein besonderer Dank für die wunderbare Zusammenarbeit und seine Hilfe während des Projektes und dieser Arbeit. Danken möchte ich auch meiner Familie und meinen Freunden, die mich bei dieser Arbeit aufopferungsvoll unterstützt haben.

Kapitel 2

Internetportal "Metropolregion Hamburg"

Die Metropolregion Hamburg (MRH) setzt sich momentan aus der Freien und Hansestadt Hamburg, den fünf schleswig-holsteinischen Landkreisen Herzogtum Lauenburg, Pinneberg, Segeberg, Steinburg und Stormarn, dem Wirtschaftsraum Brunsbüttel, sowie den acht niedersächsischen Landkreisen Cuxhaven, Harburg, Lüchow-Dannenberg, Lüneburg, Rotenburg (Wümme), Soltau-Fallingbostel, Stade und Uelzen zusammen. Es wird allerdings unter den Beteiligten darüber nachgedacht diese Ausdehnung nach Westen bis vor Bremen, nach Süden bis vor Hannover, nach Norden bis Kiel und nach Osten bis Schwerin zu erweitern. Die in diesen Bereichen vorhandenen Wirtschaftsfelder sollen mit einbezogen werden, um die MRH so in der entsprechenden Größe aufzustellen, und sie auch weltweit wirtschaftlich konkurrenzfähig zu machen.

Das Projekt "Internetportal für die Metropolregion Hamburg" ist eine Projektkooperation zwischen dem Arbeitsbereich Softwaresysteme und dem Arbeitsbereich Stadt-, Regional- und Umweltplanung der Technischen Universität Hamburg-Harburg aus dem Jahre 2002.

2.1 Historie des Portals

Das Internetportal der MRH ist Teil der Gesamtstrategie, die dazu dient die MRH in ihrem geografischen, wirtschaftlichen und verwaltungstechnischen Komplexität nach innen und aussen zu repräsentieren.

Das Internetportal wurde 2002 im Rahmen eines Teilprojekts, welches an der TUHH durchgeführt wurde, ins Leben gerufen und erstmalig umgesetzt. Die erste Veröffentlichung fand allerdings erst im Juni 2003 statt, nachdem einige inhaltliche und strukturelle Überarbeitungen vorgenommen wurden.

Im Jahr 2004 wurde das Internetportal nochmals überarbeitet und an das Layout des Webportals hamburg.de [Hamburg.de 2005] angepasst. Derzeit wird das Portal redaktionell über ein CMS gepflegt und soll stetig erweitert werden.

Das Portal soll dem Nutzer einen Mehrwert bieten. Dieser Mehrwert soll sich auf wirtschaftliche, verwaltungstechnische und persönliche Interessen der Benutzer innerhalb der MRH erstrecken (siehe dazu [MRH Machbarkeit 2003]).

2.2 Inhalte und Ziele des Portals

Die Kerninhalte des Internetportals umfassen die Bereiche "Politik und Verwaltung", "Wirtschaft und Arbeit" und "Leben und Erholen" bzw. aktuell "Tourismus und Kultur". (siehe dazu die Website [MRH 2005]).

Es sollen Informationen, Ansprechpartner und Dienste für die in der Metropolregion wohnende Bevölkerung, in- und ausländische Firmen und Investoren zur Verfügung gestellt werden. Das Portal soll die nationale und internationale politische und wirtschaftliche Kooperation unterstützen, das Regionalbewusstsein stärken, den Standort MRH wirtschaftlich stärken und ausweiten, sowie den Bereich eGovernment unterstützen ([REK 2000] und [Bauer 2005]).

Im Folgenden werden die zentralen Inhalte und Funktionsweisen des Portals dargestellt, von denen einige bereits umgesetzt sind, oder sich in Umsetzung beziehungsweise in Planung befinden.

2.2.1 Aufgaben und Funktion des Portals

Das Portal der MRH hat die Aufgabe Schnittstellen und Kommunikationsmöglichkeiten zwischen den Bürgern, der Verwaltung, regionalen, nationalen und internationalen Firmen, der Politik, Institutionen, Bildungseinrichtungen und Wirtschaftsförderern zu etablieren. Diese Schnittstellen und Kommunikationsmöglichkeiten ermöglichen eine erweiterte und effizientere Wertschöpfung der Wirtschaft in der MRH, welche die Transparenz von Führung und Verwaltung der MRH für alle Einwohner erhöht, die Kosten für Bürgerinformationen und -verwaltung senken, verwaltungstechnische Pflichten für Bürger und Firmen wesentlich vereinfachen und den Zugang zu Informationen und Diensten aus der MRH zentralisieren. Nähere Informationen finden sich in [MRH Machbarkeit 2003], unter [MRH 2003] und [Bauer 2005].

Um diese Ziele zu erreichen, soll das Portal der MRH auf die Integration von Daten und Diensten setzen, die durch andere Dienstleister, innerhalb und ausserhalb der MRH, zur Verfügung gestellt werden. Der Umfang der Redaktion und die Verwaltung für das Portal der MRH soll effizient und schlank gehalten werden, um die laufenden Kosten zu minimieren.

2.2.2 eGovernment

Eines der zentralen Themen des Portals der MRH ist die Erweiterung der eGovernment-Dienste für die Bürger.

Der Begriff eGovernment umfasst dabei mehrere Bereiche und Ziele, die in einigen Bereichen, aufgrund technischer oder infrastruktureller Umstände, nur teilweise umsetzbar sind¹:

Moderne Verwaltung

Die Verwaltung erscheint der Bürgerin und dem Bürger als ein moderner und kundenorientierter Dienstleister.

Informationen, Anträge, Formulare sind über das Internet verfügbar, können am Rechner ausgefüllt werden, an die entsprechende Behörde übermittelt und dort elektronisch weiterverarbeitet werden. Informationen, Auskünfte und Bescheide der Behörden können entweder direkt von der Website abgerufen oder der Bürgerin und dem Bürger elektronisch zugesandt werden.

Bürgerbeteiligung

Das politische Interesse und Engagement der Bürgerinnen und Bürger wird nachhaltig gefördert.

Sie informieren sich im Internet über die aktuelle Ratsarbeit, Projekte, Vorhaben und dergleichen, diskutieren über kommunalpolitische Themen und können über elektronische Abstimmungsmöglichkeiten Anteil am Entscheidungsprozeß nehmen.

Tourismus

Durch eGovernment werden Städte, Landkreise, Länder und Regionen als Reiseziele attraktiver

Aktuelle Informationen über Anreisemöglichkeiten, Sehenswürdigkeiten, Kultur- und Freizeitangebote erleichtern die Planung des Urlaubs. Durch Online-Buchung von Anreise, Hotelzimmern, Ferienhäusern, Konzerten, Führungen und sonstigen Veranstaltungen besteht die Möglichkeit den Urlaub von zu Hause aus zu organisieren und zu buchen.

Wirtschaft

Durch eGovernment beschleunigen sich Planungs- und Genehmigungsverfahren, werden Unternehmenskosten gesenkt, neue Aufträge generiert und weitere Kundenkreise erschlossen.

Durch Online-Abwicklung von Routineaufgaben, wie Kfz-Zulassung oder Meldeauskünfte, spart die Bürgerin und der Bürger Zeit und Geld, und durch elektronische Ausschreibungen und Vergaben wird es Unternehmen ermöglicht, leichter und schneller am Wettbewerb teilzunehmen oder auf diesen zu reagieren.

¹ Weiteres zum Begriff eGovernment findet sich in [Bauer 2005]

Der Bürger wird durch das Portal einen einfacheren und schnelleren Zugang zu Dokumenten und verwaltungstechnischen Prozessen erhalten. Weiterhin wird es ihm erleichtert werden, seinen Auskunftspflichten nachzukommen und sich aktiv an Entscheidungen zu beteiligen.

Viele Prozesse im Bereich eGovernment, gerade in Entscheidungsprozessen, stehen in Zusammenhang mit geografischen Informationen. Dazu gehören beispielsweise Gewerbegebietsbörsen, Planauslegungen, Bauvorhaben, bis hin zu interkommunaler Zusammenarbeit bei übergreifenden Projekten. Hier hilft es dem Bürger, aber auch den Projektbeteiligten, sich durch eine Visualisierung der Geodaten einen Überblick über die Vorhaben zu verschaffen. Zusätzlich soll die Möglichkeit geboten werden, sich anhand dieser Visualisierungen aktiv am Projektprozess zu beteiligen (siehe auch [Krause u. a. 2003] und [MKT 2005]).

2.2.3 Redaktionelle Prozesse

Die redaktionelle Pflege des gesamten Portalinhaltes der Metropolregion Hamburg soll im Aufwand so gering wie möglich gehalten werden. Die Redaktion soll sich nur um die Teile und Inhalte des Portals kümmern, die nicht durch andere Portale und Websites zur Verfügung gestellt werden können (Content Syndication¹). Dies betrifft zum Beispiel wichtige Meldungen und allgemeine Informationen zur Metropolregion. Der überwiegende Teil der Inhalte soll verteilt aus anderen Quellen gespeist werden (siehe Kapitel 2.2.4).

Es ist ein redaktioneller Prozess vorgesehen, in dem der oder die Redakteure die entsprechenden Inhalte verfassen und in der Website einordnen. Die Freigabe dieser Inhalte erfolgt dann durch eine weitere Person, welche die Inhalte und Positionierung noch einmal überprüft und dann für die Website freigibt, oder nochmals dem Redakteur vorlegt (Vier-Augen-Prinzip²). So wird sichergestellt, dass alle Inhalte fehlerfrei und abgestimmt auf der Website erscheinen.

2.2.4 Dienst-, Datensyndikation und -integration

Das Portal MRH soll Daten und Informationen von verschiedenen Anbietern zentral verfügbar machen. Dabei sollen die zu vermittelnden Inhalte überwiegend aus externen Quellen gespeist werden. Diese externen Quellen können neben unterschiedlichen und verwandten Daten auch ähnliche oder gar gleiche zur Verfügung stellen. Das Portal soll dabei vermeiden dem Benutzer redundante Informationen verschiedener Anbieter zu präsentieren.

Zu den externen Quellen gehören neben Datenanbietern auch Dienstanbieter. Nach Möglichkeit sollen alle Daten, die von externen Anbietern stammen, zur Vermeidung von Redundanz

¹ Unter Content-Syndication wird die Verbindung von Inhalten verschiedener Websites verstanden. Besonders interessant ist Content-Syndication für Websites, die ihr Angebot mit business- und branchenrelevanten Informationen aufwerten wollen. So zum Beispiel Börsenkurse, aktuelle Nachrichten, aber auch speziell ausgewählte Inhalte anderer Webseiten. [Wikipedia 2005]

² Das Vier-Augen-Prinzip besagt, dass wichtige Entscheidungen nicht von einer einzelnen Person getroffen werden sollen bzw. dürfen. Ziel ist es, das Risiko von Fehlern und Missbrauch zu reduzieren. [Wikipedia 2005]

auch nur dort vorgehalten werden. In Einzelfällen können die Daten aber auch in das Portal der Metropolregion repliziert werden.

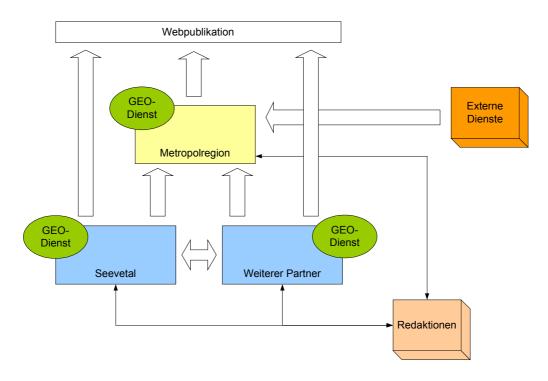


Abbildung 1: Portal der MRH und Contentpartner

Die Abbildung 1 zeigt das angestrebte Gesamtsystem mit den verschiedenen regionalen Portalen als Contentlieferanten. Wie zu erkennen ist, liefert die Website der Gemeinde Seevetal einen Teil ihres Contents an das Portal der MRH. Daneben können noch beliebig weitere Partner existieren, die ihrerseits Teile ihres Contents der MRH zur Verfügung stellen. Zusätzlich können die Partner untereinander Content austauschen. Externe Dienste, wie zum Beispiel Auskunftsysteme, werden in das Portal der MRH eingebunden. Der Austausch von Content muss zwischen allen Beteiligten grundsätzlich vertraglich geregelt werden. Die Websites der Partner und das Portal der MRH werden durch Content Management Systeme (CMS) verwaltet und mit einer eigenen Redaktion gepflegt. Diese Redaktionen entscheiden auch, welcher Content der Lieferanten wo auf ihrer Site eingestellt wird. Alle Websites sollen um GEO-Dienste erweitert werden können, um geografische Informationen grafisch (zum Beispiel in Form von Karten) darzustellen.

Über die Partner sollen diverse Dienste und Daten in das Portal der MRH integriert werden. Einen Ausschnitt der angedachten Dienst- und Datensyndikation bietet die folgende Abbildung.

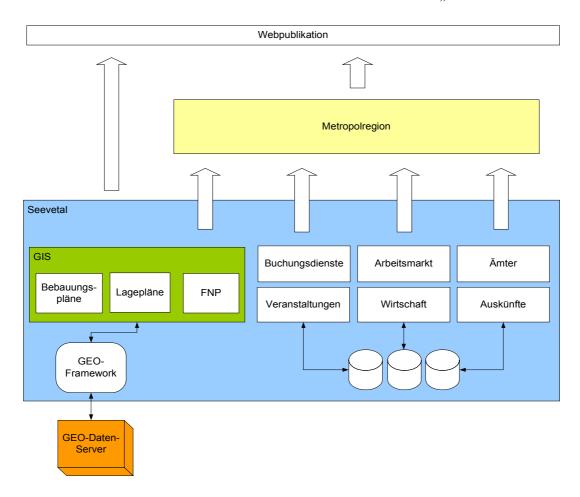


Abbildung 2: Syndikation von Diensten und Daten in das MRH Portal

Wie in Abbildung 2 zu sehen ist, werden regionale Inhalte und Dienste der Gemeinde Seevetal direkt über die eigene Website publiziert und zusätzlich über das Portal der Metropolregion den Bürgern zentral zur Verfügung gestellt. Zusätzlich stellt eine Geografisches Informationssystem (GIS) den Bürgern, auf Basis verteilter Geodaten, Pläne interaktiv dar. Auf die gleiche Weise integrieren sich auch alle anderen beteiligten Partner in die Gesamtstruktur der Internetpräsenz der Metropolregion Hamburg.

Kapitel 3

Content Management System (CMS)

Der Begriff Content Management System setzt sich aus drei einzelnen Begriffen zusammen.

Der Content ist der Inhalt und die Strukturierung von Informationen, die vermittelt werden sollen. Er kann dabei für verschiedene Publikationskanäle vorgesehen sein, wie Webseiten, Emails, Printmedien, usw.

Das Management bezieht sich auf die Verwaltung des Contents, die Organisation von definierten Arbeitsschritten und Prozessen und die Publikation der Inhalte durch bestimmte Personen (Redakteure).

Das System bezieht sich auf die eingesetzte Technik in Form von Hardware und Software, sowie auf die vorhandenen Schnittstellen und die Einsatzumgebung mit der Tätigkeiten auf dem Content durchgeführt werden.

Bei einem CMS werden einzelne Inhalte als Dokument bezeichnet. Dokumente sind dabei eine Aggregation von Datenfeldern, also Datensätze, die einen bestimmten Inhalt repräsentieren.

Zusammengefasst ist ein CMS also ein System, mit dem Personen (Redakteure) in einer bestimmten Umgebung mit definierten Werkzeugen und festgelegten Arbeitsschritten Inhalte (Dokumente) verwalten und publizieren können (siehe dazu auch [Baumann 2001], [Lüpsen], [Internet Lexikon 2005] und [Wikipedia 2005]).

3.1 Aufgaben und Einsatz von CMS

Ein CMS bietet den Redakteuren ein Werkzeug, welches es ihnen erleichtert Inhalte zu verwalten und zu pflegen. Durch vordefinierte Masken, Arbeitsschritte und Prozesse werden Fehler vermieden und sichergestellt, dass der Informationstransport immer auf die gleiche Weise von statten geht.

Aus diesem Grund existiert bei einem CMS eine Trennung von Darstellung und Inhalt, damit sich ein Redakteur nicht um die korrekte Präsentation des von ihm bearbeiteten Dokuments kümmern muss. Außerdem muss der Redakteur über keine Erfahrung mit der Art der späteren Darstellung des Inhalts verfügen. Er benötigt zum Beispiel keine HTML-Kenntnisse, wenn der Inhalt später auf einer Website dargestellt wird. Dadurch besteht in CMS die Möglichkeit, Darstellung und Inhalt unabhängig voneinander zu verändern und durch Schnittstellen über verschiedene Kanäle zu publizieren (siehe auch [CMS Evaluation]). Die Art der Präsentation des Inhalts kann dabei von Kanal zu Kanal variieren. Die Darstellung eines Inhalts auf dem Display eines Handys ist zum Beispiel wesentlich spartanischer, als der gleiche Inhalt auf einer aufwendig gestalteten Website, oder in einem gedruckten Katalog (vgl. [Holler 2003]).

3.2 CMS Funktionsumfang

Ein Content Management System sollte neben der Trennung von Darstellung und Inhalt noch über weitere Funktionen verfügen. Der folgende Überblick über den Funktionsumfang von CMS orientiert sich an [rothfuß/ried]. Weitere CMS-Anforderungen und -Eigenschaften finden sich in [Ziegler 2002] und [CMS Evaluation].

Zugangskontrolle

Der Zugang zum CMS wird nur einem fest definierten Kreis von Benutzern gewährt. Benutzer können zu Gruppen zusammengefasst werden und Rechte können sowohl für Gruppen, als auch speziell für einzelne Benutzer festgelegt werden. Rechte, die vergeben werden können, sind beispielsweise Lese-, Schreib-, Lösch-, Publikationsrechte und dergleichen. Je nach Ausprägung des CMS sind noch weitere Rechte individuell einstellbar. Die Rechtearchitektur sollte restriktiv gestaltet sein, also dem Benutzer alles verwehren, außer es ist ihm explizit erlaubt.

Logging / Protokollierung

Logging- und Protokollierungsdaten verfügen über einen Zeitstempel, um zeitliche Verläufe von Aktionen und Zuständen nachvollziehen zu können. Logging und Protokollierung wird unterschieden.

Das Logging von Aktionen und Zuständen sind Aufzeichnungen von diversen Systeminformationen. Dies können Statusmeldungen, Fehlermeldungen, Laufzeitmeldungen und
dergleichen sein. Loggingdaten werden häufig in verschiedene Relevanzebenen gegliedert.
Ein Beispiel für verschiedene Relevanzebenen wäre folgendes: Fehler (Priorität 1), Warnungen (Priorität 2), Informationen (Priorität 3). Fehler geben Auskunft darüber, dass zum
Beispiel ein Problem vorherrscht, welches die weitere Ausführung eines Prozesses
verhindert. Eine Warnung wäre zum Beispiel , dass ein nicht erwünschter Zustand des
CMS erreicht ist, der allerdings nicht den Weiterbetrieb gefährdet. Informationen geben
zum Beispiel einem Administrator Anhaltspunkte zum Status des CMS. Fehler müssen also

3.2 CMS Funktionsumfang 21

umgehend beseitigt werden, besitzen also die höchste Priorität, danach folgen die Warnungen und Informationen. Die Anzahl der Relevanzebenen ist je nach CMS unterschiedlich eingeteilt und kann wesentlich feiner sein. Loggingdaten werden überwiegend im Dateisystem geführt.

Die Protokollierungsdaten eines CMS umfassen Logindaten, Zugriffe, Operationen auf Dokumenten, Versionierungsinformationen und so weiter. Protokollierungsdaten besitzen normalerweise keine Relevanzordnung. Auch werden verschiedene Protokollierungsdaten nicht immer im gleichen Datenspeicher abgelegt. Beispielsweise macht es Sinn, dass Logindaten im Dateisystem abgelegt werden, Versionsinformationen sind aber in der Regel eher im gleichen Datenspeicher abgelegt wie die Datensätze der zu pflegenden Dokumente selber.

Der Detaillierungsgrad von Logging und Protokollierung kann einstellbar sein und sollte vom Administrator dem Systemumfang sowie den Anforderungen entsprechend angepasst sein.

Backup / Restore

Ein CMS sollte über die Möglichkeit verfügen, seine Einstellungen und die Datensätze in ein Zweitsystem zu sichern und diese auch wieder von dort zu übernehmen. Dieses Zweitsystem könnte zum Beispiel das Dateisystem sein, in dem die Datensätze in Dateien gesichert werden und auch aus diesen gesicherten Datensätzen das System wiederhergestellt werden kann. Dabei kann ein CMS-eigenes Backup-Restore-System eingesetzt werden, oder auch eines, dass für das Datenspeicherungssystem existiert, welches vom CMS verwendet wird.

Mehrplatzanwendung

Da CMS meist von mehr als einen Redakteur eingesetzt werden, ist es notwendig, dass auf ein CMS von mehreren Arbeitsplätzen gleichzeitig zugegriffen werden kann.

Check-In und Check-Out

Die vom CMS verwalteten Dokumente (Texte, Multimediainhalte, usw.) können von den Redakteuren zur Bearbeitung ausgeliehen werden. Dadurch wird sichergestellt, dass Daten nicht gleichzeitig von mehreren Redakteuren verändert werden können und einsichtig ist, wer welche Daten zur Zeit bearbeitet. Dies verhindert Inkonsistenzen und ermöglicht einen besseren Überblick über bearbeitete Daten und die Abstimmung darüber.

Metainformationen und Verwaltungsfunktionen

Dokumente eines CMS können mit Metainformationen angereichert werden. Dazu gehören Namensgebung, Beschreibungen, Versionsdokumentationen und weitere Informationen die mit den Daten beziehungsweise ihrer Verwendung im Zusammenhang stehen. Verwaltungsfunktionen sind zum Beispiel die automatische Überprüfung von verknüpften Dokumenten, Automatisierungshilfen, das Auffinden von Inkonsistenzen und Ähnliches.

Metainformationen und Verwaltungsfunktionen sind dabei von CMS zu CMS verschieden und an Einsatzgebiet und -umfang entsprechend angepasst.

Anfragefunktionen

Hiermit werden zum einen Suchfunktionen angeboten, die es ermöglichen in umfangreichen Datenmengen eines CMS bestimmte Dokumente aufzufinden. Die Suche kann dabei sowohl über die Metainformationen, als auch über die Dokumentinhalte angelegt sein. Zum anderen sind auch Anfragen nach besonderen Eigenschaften der Dokumente und des CMS denkbar. Dies sind zum Beispiel Suchmöglichkeiten nach unverlinkten beziehungsweise unverknüpften oder verwaisten Dokumenten und nach Redakteuren.

Massenoperationen

Massenoperationen sind Operationen, die gleichzeitig auf einer großen Menge von Daten ausgeführt werden. Dazu gehören unter anderem Operationen wie Suchen und Ersetzen, Publikationsoperationen, das Verschieben von Dokumenten, Umbenennungen, Archivierungen und Versionierungen. Die Typen von Massenoperationen sind grundsätzlich vom Einsatzgebiet des CMS abhängig und ermöglichen Ablauf- und Zeitvorteile.

Bearbeitung und Verifikation

Redakteuren wird über ein oder mehrere Frontends die Verwaltung und Pflege der Dokumente erleichtert. Als Frontend wird häufig eine webbasierende Schnittstelle angeboten, die ohne großen Installations- und Administrationsaufwand die dezentrale Pflege der Dokumente ermöglicht. Häufig werden auch proprietäre Schnittstellen angeboten, wie zum Beispiel CMS-eigene Editoren. Diese ermöglichen eine Auslagerung von Funktionalität auf das Clientsystem und somit eine performantere Ausführung von Operationen und eine flexiblere Anpassung auf die Bedürfnisse von Redakteur und CMS. Dies muss allerdings mit einem höheren Implementations- und Administrationsaufwand erkauft werden, da sowohl die CMS Editoren für verschiedene Betriebssysteme umgesetzt werden müssen, als auch eventuelle Systemanpassungen und -vorraussetzungen dafür notwendig sind.

Mehrsprachfähigkeit

Mehrsprachigkeit sollte ein CMS auf zwei Arten unterstützen. Zum einen sollte das CMS in mehreren Sprachen verfügbar sein. Dazu gehört zum Beispiel, dass der erwähnte Editor in seiner Menüführung, seinen Optionen, Beschreibungen und dergleichen in mehreren Sprachen zur Verfügung steht. Zum anderen sollten die zu verwalteten Dokumente in mehreren Sprachen pflegbar sein und gleiche Dokumente in unterschiedlichen Sprachen miteinander in Verbindung gebracht werden können.

Workflow

Workflows definieren die Arbeitsschritte, welche die Dokumente innerhalb eines CMS durchlaufen. Eine mögliche Folge wäre zum Beispiel: erzeugen – bearbeiten – speichern – freigeben – überprüfen – publizieren. Die in CMS eingesetzten Workflows können sowohl in der Anzahl, als auch in der Ausprägung variieren. Neben vordefinierten Workflows soll-

3.2 CMS Funktionsumfang 23

ten in einem CMS eigene Workflows definierbar sein. Workflows definieren im CMS jeweils zu einem Zeitpunkt einen eindeutigen Zustand für jedes einzelne Dokument.

Trennung von Content und Layout

Im CMS werden die Datensätze zu den Dokumenten unabhängig von Ihrer Darstellung gespeichert. Dies bedeutet, das Layoutinformationen nicht zusammen mit den Datensätzen gespeichert werden, und nicht vom Redakteur verwaltet werden müssen. Dadurch ist es möglich, dass die Dokumente über unterschiedliche Ausgabekanäle geleitet werden können und jeweils in ein spezielles Layout eingebunden werden. Möglich ist so zum Beispiel die Ausgabe als HTML-, XML-, oder PDF-Format. Das Layout für die speziellen Ausgabekanäle wird durch Templates realisiert. Die Dokumente und die Templates werden mit Hilfe eines Generators in eine Enddarstellung transformiert und an den entsprechenden Ausgabekanal übergeben.

Im- und Export

Ein CMS sollte über Schnittstellen verfügen, die es ermöglichen die Dokumente sowohl aus dem System zu exportieren, um sie anderweitig verwerten zu können, als auch externe Datensätze in das System zu importieren. Dies erleichtert Migrationsverfahren von einem CMS zum anderen, oder Datentransfer von System zu System, um Datensätze zu replizieren.

Dabei sollten verschiedene Ausprägungen der Im- und Exporte möglich sein. Der Bestand an Dokumenten des CMS sollte sowohl komplett, als auch nur teilweise im- und exportiert werden können. Beim Im- und Export sollen bestimmte Eigenarten des CMS beachtet und verwendet werden. Beispielsweise muss eine korrekte Versionsbehandlung der Dokumente beim Im- und Export eingehalten werden.

Link-Stabilität und -Konsistenz

Ein CMS sollte die Dokumente so verwalten, dass ein einzelnes Dokument über die Zeit immer wieder über den gleichen Zugriff zu erreichen ist. Dies bedeutet, dass es immer über die gleiche URL, API Abfrage oder dergleichen abrufbar ist. Verwendet man zu verschiedenen Zeiten den gleichen Zugriff auf ein Dokument, dann liefert das CMS grundsätzlich immer das gleiche Dokument aus.

Zusätzlich dazu muss auch die Konsistenz erhalten bleiben. Dies bedeutet, dass zum Beispiel beim Löschen eines Dokuments, Verknüpfungen und Verweise innerhalb des CMS auf dieses Dokument ebenfalls gelöscht werden. So stellt das CMS intern sicher, dass es keine toten Verbindungen zwischen Dokumenten gibt.

Multimedia

Ein CMS verwaltet eine große Bandbreite an Inhalts- und Dateiformaten. Dazu gehören zum Beispiel Texte, Grafiken, Audiodateien, Filmdateien und Anwendungen. Diese einzelnen Elemente sollten miteinander verknüpft werden können, um sie gemeinsam darzustellen.

Als Content Management Systeme werden häufig nur Web-CMS aufgefasst, die speziell für die Verwaltung und Publikation von Informationen im Internet oder Intranet ausgelegt sind. Diese bilden aber nur einen Teil der Einsatzmöglichkeiten von CMS. Die zu publizierende Gesamtinformation setzt sich in einem CMS aus mehreren einzelnen Bestandteilen zusammen, die unterschiedlichen Typs (Texte, Bilder, Audio, Video, ...) und Herkunft sein können. Dadurch ist es sehr einfach möglich Inhalte redaktionell zu pflegen, über unterschiedliche Plattformen zu publizieren und eine große Bandbreite von Konsumenten anzusprechen.

Kapitel 4

Konzepte zur Integration von Inhalten in Web-Portale

Ziel einer Integration ist es, Inhalte aus verteilten Systemen in ein Zielsystem zu überführen und in diesem darzustellen, beziehungsweise darüber verfügbar zu machen. Diese Inhalte können Texte, Bilder, Musikstücke und Filme sein. Es sollten aber auch komplexere Inhalte und Funktionen in ein Zielsystem integrierbar sein. Beispiele hierfür sind Auskunftssysteme, Verkaufsplattformen und dergleichen, die neben Informationen auch komplexe Funktionen zur Verfügung stellen und der Benutzer mit dem Web-Portal interagieren kann.

Sinn macht eine Integration immer dann, wenn die darzustellenden Inhalte schon in anderen Systemen verfügbar sind. Dies bedeutet, dass es vermieden werden kann, die Informationen in zwei verschiedenen Systemen verwalten und pflegen zu müssen. Somit ist eine höhere Effizienz bei der Inhaltspflege möglich. Dazu bedarf es einem Abgleichmechanismus zwischen den verschiedenen Systemen, die die gleichen Informationen vorhalten und transportieren sollen. Dieser Mechanismus wird Integration genannt.

• Ein Vorteil bei der Integration ist es, dass kein ausgezeichnetes Systeme für die Verwaltung aller Inhalte verwendet werden muss, sondern die Möglichkeiten genutzt werden kann, Inhalte aus verschiedenen Systemen in anderen Systemen abzubilden. So können für die jeweiligen, oft unterschiedlichen Anwendungen und Umgebungen immer die optimalen Systeme eingesetzt werden, trotzdem die Inhalte mit einem anderem System an anderer Stelle verwendet werden. Detaillierte Betrachtung zur Informationsbereitstellung, durch Integration von Inhalten in verschiedene Systeme, finden sich in [Ullman 1997] (vgl. auch [Mantel u.a. 2000]).

Diese Arbeit betrachtet vorrangig Integrationsprozesse und Anwendungsfälle, die sich für den Abgleich von Inhalten für Web-Portale eignen. Es existieren darüber hinaus diverse Möglichkeiten und Anforderungen, Inhalte über verschiedene Systeme hinweg verfügbar zu machen.

Zu unterscheiden sind bei der Integration die Art und Weise, wie die Inhalte im Systemverbund verteilt werden sollen. Dazu existieren Integrationstopologien, die sich je nach Anwendungsfall spezialisieren lassen. Des weiteren lässt sich die Art der Inhalte entscheiden, die integriert werden sollen. Wie oben erwähnt, wird zwischen einfachen Inhalten (Datensätzen beziehungsweise die dazu aggregierten Daten) und komplexeren Inhalten (Diensten) unterschieden. Bei der Integration werden die Daten betrachtet, da sie die Grundlage für alle Inhalte bilden und somit in den Systemen des Systemverbunds verfügbar gemacht werden müssen. Dementsprechend gilt es zwischen einer Daten- und einer Dienstintegration zu unterscheiden (vgl. [Becker u.a. 1999]).

4.1 Integrationstopologien

Es gibt verschiedene Möglichkeiten, um verteilte Inhalte durch einen Integrationsprozess über ein Webportal zugänglich zu machen. Daten und Dienste lassen sich gleichermaßen über solche Topologien in andere Systeme integrieren.

Die folgenden Abschnitte bieten einen Überblick über Integrationstopologien samt ihrer Vorund Nachteile. Die Topologien sind in [stefan conrad 1997] für Datenbanken dargestellt und hier für Daten und Dienste entsprechend angepasst. Daten und Dienste werden als Inhalte betrachtet, die integriert werden sollen.

4.1.1 Punkt zu Punkt

Bei der Punkt-zu-Punkt-Topologie steht ein System direkt mit einem anderen System in Kontakt und kann dort die Inhalte abholen. Jedes System kann mit anderen beliebigen Systemen verbunden werden. Die Integration kann bei dieser Topologie sowohl unidirektional als auch bidirektional erfolgen. Mit dieser Topologie lassen sich Inhalte auch transitiv integrieren, also Inhalte über Systeme hinweg transferieren.

Die Vorteile dieser Topologie sind:

- Sie ist übersichtlich bei einer geringen Anzahl beteiligter Systeme und wenig Verbindungen.
- Bei einer geringen Menge an Systemverbindungen ist die Topologie schnell und kostengünstig umzusetzen.
- Durch die Direktverbindung ist die Topologie flexibel und schnell an punktuelle Veränderungen der Systeme oder der Verbindungen anpassbar.
- Die Verbindungsgeschwindigkeiten zwischen den verbundenen Systemen sind schnell.
- Es sind keine weiteren Hardware-, Software- und Entwicklungskosten für zusätzliche Drittsysteme notwendig.

4.1 Integrationstopologien 27

Die Topologie hat folgende Nachteile:

• Sie ist unübersichtlich beziehungsweise unpraktikabel bei einer großen Anzahl beteiligter Systeme und Verbindungen.

- Sie hat hohe Folgekosten bei umfassenden Erweiterungen und Veränderungen.
- Die Topologie ist sehr unflexibel bei Einbindung neuer oder gleicher beziehungsweise ähnlicher Systeme.
- Veränderungen an einzelnen Systemen ziehen eventuell Veränderungen an allen anderen verbundenen Systemen nach sich.
- Die Administration und Überwachung ist durch die dezentrale Struktur sehr aufwendig
- Es ist aufwendig die Verwaltung von Zugriffsrechten festzulegen.

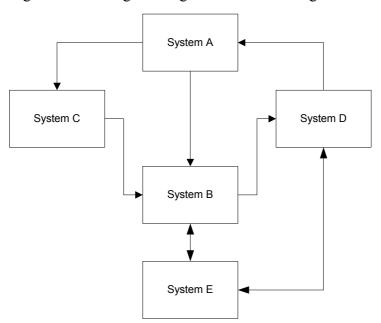


Abbildung 3: Integrationstopologie – Punkt zu Punkt

4.1.2 zentrale Verteiler

Bei dieser Topologie existieren ein oder mehrere Verteilersysteme, die Inhalte anderer Systeme weiterleiten. Die Verteilersysteme müssen dazu sowohl die Verfügbarkeit von Inhalten aus anderen Systeme registrieren, als auch diese Informationen an Systeme weitergeben. Die Verteilersysteme dienen als Vermittlungsstationen zwischen den verschiedenen Systemen. Die Verteiler müssen dabei also über eine eigene Logik verfügen, die den Inhaltstransport gewährleistet.

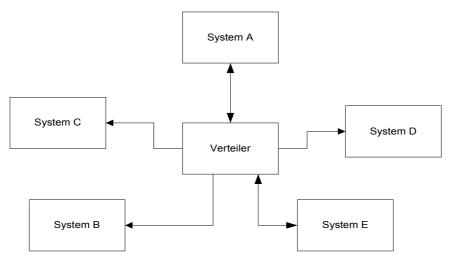


Abbildung 4: Integratinstopologie - zentraler Verteiler

Die Vorteile dieser Topologie sind:

- Durch die dedizierten Verteilersysteme ist die Topologie auch für eine große Anzahl an beteiligten Systemen und Verbindungen einsetzbar.
- Die Verwaltung und Steuerung der Verbindungen ist zentral implementiert.
- Die Festlegung und Verwaltung von Zugriffsrechten geschieht auf den Verteilersystemen.
- Die Folgekosten bei umfassenden Erweiterungen und Veränderungen sind niedrig.
- Administration und Überwachung der Verteilersysteme ist übersichtlich.
- Neue Systeme lassen sich leicht einbinden.

Die Topologie hat folgende Nachteile:

- Die Kosten der ersten Umsetzung sind hoch.
- Die Verteilersysteme erzeugen zusätzliche Hard-, Software- und Entwicklungskosten.
- Die Verbindungszeiten sind abhängig von den Verteilersystemen.

4.2 Integrationsstrategien

Die dargestellten Integrationstopologien lassen sich mit verschiedenen Integrationsstrategien umsetzen. Die in den Integrationstopologien dargestellten Systeme können aus heterogenen Komponenten (Daten- und Dienstquellen) bestehen. Diese Komponenten müssen nach außen hin eine homogene Darstellung erhalten um sie in andere Systeme integrieren zu könnend. Dazu wird je nach System eine passende Integrationsstrategie angewandt.

Die folgenden unterschiedlichen Integrationsstrategien sind in [stefan conrad 1997] für Datenbanken zu finden und für Daten und Dienste abgewandelt.

Integration lässt sich bei entsprechender Übersichtlichkeit in einem Schritt realisieren, oder sukzessive in mehreren aufeinander folgenden Integrationsschritten, die in binäre und n-äre Integrationsstrategien unterschieden werden. Bei binären Strategien werden für jeden Integrationsschritt genau zwei Datenquellen verwendet. Bei n-ären Strategien werden für jeden Integrationsschritt genau zwei Datenquellen verwendet.

4.2 Integrationsstrategien 29

tegrationsschritt mindestens zwei Datenquellen verwendet. Je mehr Integrationszwischenschritte die einzelnen Daten- oder Dienstformate bei der Integration durchlaufen müssen, umso mehr werden sie im Integrationsprozeß in ihrer Struktur verändert. Dabei ist es durchaus möglich, dass einige Formate sich kaum verändert im Endprodukt der Integration wiederfinden, andere sich hingegen sehr stark verändern (siehe dazu binäre und n-äre Integrationsstrategien in [stefan conrad 1997]).

Der Einsatz einer binären oder n-ären Strategie ist vor allem dann sinnvoll, wenn die Daten und Dienste aus den verschiedenen Systemen nach der Integration einen Bezug zueinander haben sollen. In den einzelnen Integrationszwischenschritten werden dann mehr und mehr Bezüge hergestellt, in den Zwischenformaten abgelegt, und am Ende in das Zielsystem integriert.

Bei einer einfachen Integration kann diese in einem Schritt vollzogen werden. Bei diesem Ansatz spricht man von der One-Shot Strategie. Diese ist in der Abbildung 5 dargestellt.

Wie zu sehen ist, werden die Daten und Dienste ohne Zwischenschritt direkt in das Zielsystem integriert und stehen dort zur Verfügung.

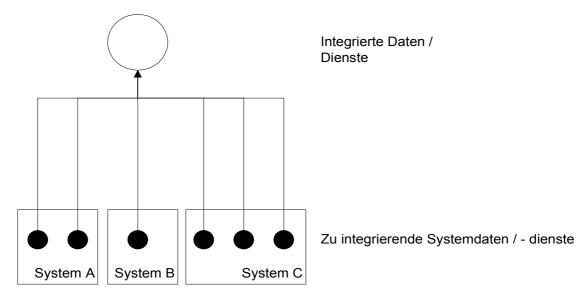


Abbildung 5: One-Shot Integrationsstrategie

Bei der binären Integrationsstrategie werden jeweils pro Integrationsschritt zwei Daten- oder Dienstformate in ein Zwischenformat überführt, die dann ihrerseits wiederum paarweise in weitere Zwischenformate überführt werden. Die Reihenfolge, in der die Integrationsschritte durchgeführt werden, und welche Daten- oder Dienstformate beziehungsweise Zwischenformate daran beteiligt sind, bestimmt das Endresultat. Daher wird die binäre Integrationsstrategie in eine balancierte und eine gewichtete Strategie unterteilt.

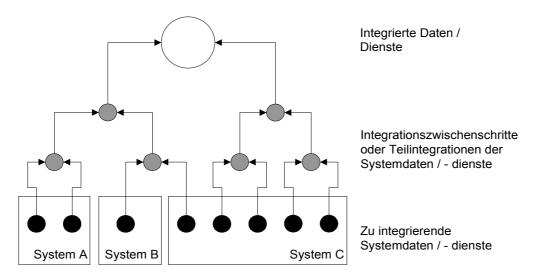


Abbildung 6: balancierte binäre Integrationsstrategie

Bei der balancierten binären Integrationsstrategie durchlaufen alle Daten- oder Dienstformate der beteiligten Systeme die gleiche Anzahl an Integrationsschritten. Es werden in einer Integrationsstufe immer paarweise zwei Formate in ein Zwischenformat überführt.

Ist die Integrationsstrategie nicht balanciert, wird von einer gewichteten Strategie gesprochen. Einige Daten beziehungsweise Dienste werden bei dieser Strategie bevorzugt. Bevorzugte Daten oder Dienste unterliegen dadurch weniger Veränderungen als nicht bevorzugte, da sie nicht so viele Integrationsschritte durchlaufen müssen. Sind zum Beispiel zwei Datensätze identisch oder haben den gleichen Inhalt, kann der bevorzugte Datensatz behalten und der zweite verworfen werden. Die folgende Abbildung zeigt zwei mögliche gewichtete binäre Integrationsstrategien.

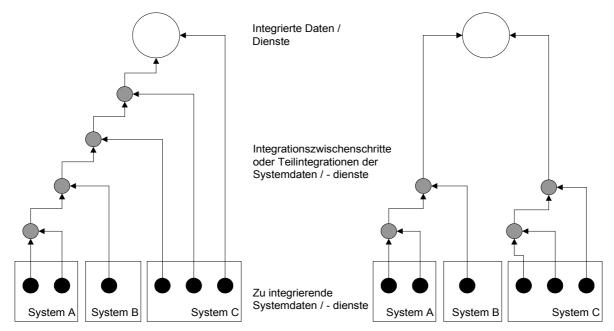


Abbildung 7: zwei Beispiele für binäre gewichtete Integrationsstrategien

4.2 Integrationsstrategien 31

Die Inhalte des Systems A durchlaufen in der links abgebildeten Strategie mehr Integrationsschritte, als in der rechts abgebildeten Strategie. Der Einfluss der Inhalte von System B und C auf die Inhalte von System A ist also bei der Integration in der linken Strategie größer, als in der rechten. Zu Integrationssstrategien siehe auch [Westerfeld 2000].

4.3 Datenintegration in CMS

Um sich diesem Thema zu nähern muss im ersten Schritt der Begriff "Daten" festgelegt werden. Im [Internet Lexikon 2005] findet sich folgende Definition:

Daten ist der eingedeutschte Plural von lat. pl. data, sg. datum = Gegebenes

Die Informatik und Datenverarbeitung (EDV) benutzen Daten als (maschinen)-lesbare und -bearbeitbare Repräsentation von Information. Die Information wird dazu in Zeichen(ketten) kodiert, deren Aufbau strengen Regeln folgt, der so genannten Syntax.

Die Semiotik definiert Daten als potenzielle Information. In der Semiotik werden Daten heute in die Sigmatik-Ebene eingeordnet.

Man unterscheidet:

- strukturierte Daten (zum Beispiel Datenbanken, XML)
- unstrukturierte Daten (beispielsweise Dokumente)

Während sich strukturierte Daten relativ einfach maschinell weiterverarbeiten lassen, ist dies bei unstrukturierten Daten nur schwer beziehungsweise ungenau möglich.

In [Meyers Lexikon 1990] lautet die informatische Definition zum Begriff "Daten" wie folgt:

In der Informations- und Datenverarbeitung versteht man unter Daten anfallende Informationen, die sich durch Zeichenfolgen eindeutig beschreiben bzw. in einer Form darstellen lassen, in der sie maschinell verarbeitet werden können.

Laut Definition sind Daten also die maschinenlesbare Repräsentationen von Informationen.

Daten liegen in Unternehmen und Institutionen in verschiedenen Formaten und Systemen vor. Diese Daten haben teilweise einen Bezug zueinander, dessen Herstellung oftmals eine Forderung in den betreffenden Unternehmen oder Institutionen ist. Dieser Bezug lässt sich nicht ohne Weiteres herstellen, da oftmals keine Schnittstellen zwischen den Systemen bestehen und diese häufig räumlich getrennt sind. Im Anwendungsfall dieser Arbeit, der Metropolregion Hamburg (MRH), sind Daten, die miteinander in Beziehung gebracht und gemeinsam über die Webportale publiziert werden sollen, sogar über verschiedenen Unternehmen und Institutionen verteilt.

Da die Portale der MRH über CMS gepflegt werden, ist es notwendig, die verteilt vorliegenden Daten in die CMS der Portale zu integrieren.

Da die Daten verteilt bei unterschiedlichen Beteiligten und in unterschiedlichen Systemen vorliegen, treten zum einen rechtliche Probleme auf, die durch entsprechende Verträge der Betei-

ligten geregelt werden müssen. Zum anderen treten informatische Probleme, wie Integritäts-, Konsistenz-, Redundanz- und Synchronisatiosprobleme auf.

Im Folgenden wird auf die Möglichkeiten eingegangen, bestehende externe Daten in CMS zu integrieren und auf Lösungsmöglichkeiten der zuvor beschriebenen informatischen Probleme aufgezeigt.

4.3.1 Alternativen der Datenintegration

Wie bereits erläutert, liegen die zu integrierenden Daten in unterschiedlichen Formaten und Systemen vor. Diese Daten sollen in ein einzelnes System integriert werden. Dabei ist zu beachten, dass Daten teilweise redundant vorliegen, miteinander in Beziehung stehen oder gebracht werden müssen, bestimmten Veränderungen unterliegen und gewisse Lebenszyklen haben.

Um Daten aus unterschiedlichen Quellen in ein CMS zu integrieren, existieren verschiedene Methoden, die mit den zuvor beschriebenen Integrationsstrategien umgesetzt werden können. Zu diesen Methoden zählen:

- Replikation der Daten,
- Verwendung / Bereitstellung von Schnittstellen und
- Föderation

Im Folgenden wird detaillierter auf diese drei Methoden eingegangen, sowie deren Vor- und Nachteile und Einsatzmöglichkeiten dargestellt. Die Methoden sind aus [DavidS.Linthicum2003] entnommen und für die CMS-Integration entsprechend erweitert beziehungsweise spezialisiert worden.

Replikation der Daten

Bei der Replikation¹ der Daten werden diese aus den externen Quell-Systemen, in das Ziel-CMS kopiert. Dort können sie dann mit den Zugriffs- und Verwaltungsmethoden des Ziel-CMS bearbeitet und publiziert werden.

Replizierte Daten stellen immer eine genaue Kopie der Originaldaten dar. Diese Kopie kann zwar im CMS geändert und verarbeitet werden, eine Abbildung auf die Originaldaten ist allerdings nicht mehr möglich. Demzufolge ist die Replikation ein gerichteter Vorgang. Es besteht nur die Möglichkeit die Daten der verschiedenen Quell-Systeme über das Ziel-CMS zu präsentieren, aber keine Möglichkeit über das Ziel-CMS in den Quell-Systemen Operationen

¹ Eine Replikation ist eine Verdopplung bzw. Vervielfältigung von Daten. Der Datenbestand der replizierten Daten ist in der Regel mit dem Original identisch. Die vervielfältigten Daten (Replikate) werden für gewöhnlich an unterschiedlichen Standorten gehalten. Dies dient einerseits zur Datensicherung; andererseits ermöglicht es die Betrachtung (und in bestimmten Fällen sogar die Bearbeitung) von großen Datenmengen an voneinander getrennten Orten, ohne Netzlast zu verursachen. [Wikipedia 2005]

auf den Daten durchzuführen. Diese Festlegung wird zuweilen aufgeweicht, indem die replizierten Daten oder bestimmte Teile dieser nach der Bearbeitung im Ziel-CMS wieder auf das Quellsystem zurück übertragen werden. Abbildung 8 zeigt die Replikation von Datensätzen aus unterschiedlichen Systemen in ein CMS.

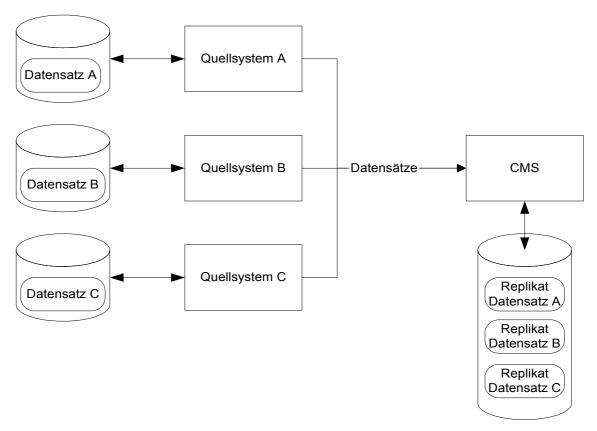


Abbildung 8: Datenintegration über Replikation in ein CMS

Um eine Replikation durchzuführen, muss zumeist eine Transformation der Daten vorgenommen werden. Die Daten werden also aus den unterschiedlichen Formaten der Quellsysteme in ein Datenformat überführt, welches kompatibel zum Ziel-CMS ist.

Zusätzlich treten bei der Replikation einige Fragen und Probleme auf.

Es muss festgelegt werden, wann welche Daten aus den Quell-Systemen in das Ziel-CMS repliziert werden. Um diese Frage zu beantworten, ist zu definieren, welche Daten aus welchen Systemen über das Ziel-CMS zur Verfügung gestellt werden sollen. Die Einschränkungen dazu sollte in den Quellsystemen festgelegt werden, um die Replikation geheimer beziehungsweise nicht öffentlicher Daten zu verhindern. Es wird ein Transformationsschema benötigt, durch welches sichergestellt wird, dass die zu replizierenden Daten der Quellsysteme kompatibel zu dem benötigten Format im Ziel-CMS sind. Zusätzlich definiert das Transformationsschema, wie die Daten im Ziel-CMS strukturiert sein sollen.

Um die Aktualität der replizierten Daten im Ziel-CMS sicher zu stellen, werden die Daten in bestimmten Zyklen repliziert. Für die Replikationszyklen lassen sich Prioritätsstufen festlegen.

Diese Prioritätsstufen sind abhängig von der Art der Daten und wie zeitnah diese Daten aufgrund ihres Informationsgehalts repliziert werden müssen. Unterscheiden muss man hier zwischen manueller und automatischer Replikation. Bei manueller Replikation wird der Abgleich der Daten durch eine Person manuell angestoßen. Hier kann es notwendig sein, dass bei jedem Abgleich festgelegt wird, welche Daten repliziert werden sollen. Bei manueller Replikation bestimmt die Person, zu welchem Zeitpunkt die Daten zu replizieren sind. Bei automatischer Replikation bestimmt ein Programm, wann die Replikation durchgeführt wird. Sehr wichtige Daten können zum Beispiel sofort repliziert werden, sobald sie eine Änderung erfahren. Die Replikation kann auch zyklisch (getaktet) durchgeführt werden, wie zum Beispiel jede Stunde oder einmal am Tag um ein Uhr. Es ist auch möglich, spezielle Zeitpunkte für die Replikation festzulegen, um zum Beispiel den Abgleich zu Zeitpunkten sicher zu stellen, an denen Ereignisse eintreten werden, die eine Replikation der Daten erfordern. Ausführliche Betrachtungen der Datenreplikation und der Problemstellungen finden sich unter anderem in [Lenz 97]. [Sybase 2000] bietet einen Überblick über die in der Praxis angewandten Replikations- und Synchronisationsverfahren.

Die Initiierung der Replikation kann sowohl von den einzelnen Quellsystemen vorgenommen, als auch durch das Ziel-CMS angestoßen werden. Die Replikation lässt sich also sowohl im push-, als auch im pull-Betrieb realisieren. Der push-Betrieb ermöglicht, dass Daten sofort an das Ziel-CMS übermittelt werden können, sobald sie in einem Quellsystem zur Verfügung stehen. Außerdem ist es möglich, dass zu bestimmten Zeitpunkten das Ziel-CMS die entsprechenden aktuellen Daten aus den Quellsystemen über den pull-Betrieb abholt. Somit stehen dem Ziel-CMS mit der Replikation alle zeitkritischen Daten zum richtigen Zeitpunkt zur Verfügung.

Zusätzlich muss durch eine Strategie festgelegt sein, wie mit Daten verfahren wird, die bereits im Ziel-CMS vorhanden sind und durch eine erneute Replikation überschrieben werden würden. Dies gilt vor allem, wenn replizierte Daten im Ziel-CMS durch Redakteure geändert oder ergänzt wurden. Es kann grundsätzlich festgelegt sein, dass die Datenquelle die ausschlaggebenden Daten (die Master-Daten) hält, also Datensätze im Ziel-CMS grundsätzlich bei der Replikation überschrieben werden. Wird ein Datensatz nicht wieder repliziert kann er entweder im Ziel-CMS erhalten bleiben oder beim Replikationsvorgang gelöscht werden. Sollen die bereits in das Ziel-CMS replizierten älteren Daten erhalten bleiben, können gleiche immer wieder replizierte Datensätze mittels einer Versionierung innerhalb des Ziel-CMS festgehalten werden. Es kann auch durch eine Person überprüft werden, welche Datensätze in das Ziel-CMS repliziert werden sollen und wie sich das Ziel-CMS bei der Replikation der Datensätze jeweils verhalten soll. Diese Strategie lässt sich natürlich nur bei einer überschaubaren Datenmenge einsetzen, da der redaktionelle Aufwand sonst zu hoch wäre. Die Strategien lassen sich durchaus gemeinsam nebeneinander einsetzen. Je nach Ziel-CMS und dessen Einsatz ist für unterschiedliche Datensätze eine andere Strategie notwendig.

Außerdem sind die Lebenszeiten, die replizierte Datensätze im Ziel-CMS haben, festzulegen. Die Lebenszeiten können von der zuvor beschriebenen eingesetzten Strategie beeinflusst

4.3 Datenintegration in CMS 35

werden beziehungsweise abhängig sein. Ist die Datenquelle ausschlaggebend, und werden somit alte Datensätze grundsätzlich gelöscht, dann entspricht die Lebenszeit der Datensätze genau dem Replikationszyklus. Werden ältere Datensätze nicht durch den Replikationszyklus entfernt, dann besteht die Möglichkeit, dass sie durch einen Redakteur gelöscht werden müssen, nach einer abgelaufenen Zeitspanne im Ziel-CMS automatisch entfernt oder zu einem festgelegten Zeitpunkt aus dem Ziel-CMS genommen werden. Auch hier sind Mischformen anwendbar und abhängig vom CMS und dessen Einsatzzweck.

Bei vielen Daten, die zu replizieren sind, ist es sinnvoll und teilweise notwendig, dass die Quelldaten strukturiert in das Ziel-CMS repliziert werden. Durch Anreicherung der Quelldaten in den Quellsystemen mit Metadaten können die Datensätze sinnvoll in bereits vorhandene, oder neu zu erzeugende Strukturen des Ziel-CMS repliziert werden. So ist eine Verwaltung der Daten im Ziel-CMS zum Beispiel nach Themenbereichen möglich und eine einfache und zügige Weiterverarbeitung durch die Redakteure gegeben.

Vorteile der Integrationsvariante:

- Die Replikationsstruktur ist gut überschaubar.
- Die Variante ist schnell, einfach und kostengünstig zu implementieren.
- CMS bieten oft Schnittstellen, die für die Replikation verwendet, oder angepasst werden können.
- Es können beliebig viele Quellsysteme angebunden werden.
- Die Quellsystemen können zwischen Daten, die repliziert und welchen die nicht repliziert werden sollen, sauber trennen.
- Die Variante lässt sich als push- und pull-Mechanismus implementieren.
- Eine Anreicherung mit Metadaten ermöglicht eine Replikation in vorhandene Strukturen des Ziel-CMS mit semantischen Bezug.

Nachteile der Integrationsvariante

- Die Variante ist nicht rückkanalfähig, womit keine Änderungen der Quelldaten vom CMS aus möglich sind.
- Sollen in kurzen Zeiträumen große Datenmengen abgeglichen werden, dann ist diese Variante ineffizient.
- Bestehende Operationen auf den Daten innerhalb der Quellsysteme werden nicht mit repliziert und müssen im CMS nachgebildet werden.

Verwendung / Bereitstellung von Schnittstellen

Bei dieser Variante der Integration stellen die Quellsysteme ihrerseits Schnittstellen zur Verfügung, die dem Ziel-CMS den Zugriff auf ihre Daten erlauben. Das Ziel-CMS muss dazu an die entsprechenden Schnittstellen der Quellsysteme angepasst werden, und die gelieferten Datensätze in die eigene Repräsentation transformieren. Es besteht, ähnlich wie bei der Replikation, die Möglichkeit die Daten zwischenzuspeichern, woraus dann allerdings die zuvor beschriebenen Probleme der Redundanz und Konsistenz resultieren. Weiterhin können die benö-

tigten Daten auch immer zu den Zeitpunkten von den Quellsystemen abgerufen werden, zu denen sie benötigt werden. Dabei muss aber sichergestellt sein, dass die Quellsysteme zu diesen Zeitpunkten auch zur Verfügung stehen.

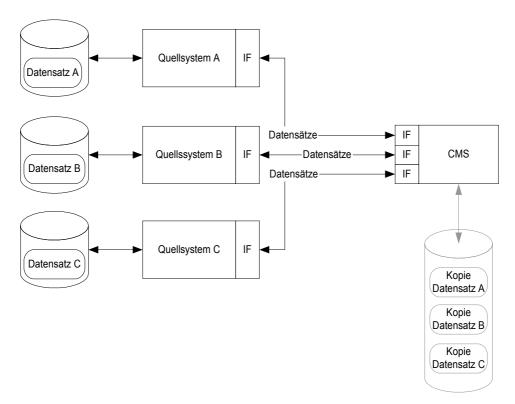


Abbildung 9: Datenintegration über Schnittstellen in ein CMS

Wie in Abbildung 9 zu sehen ist, verfügt jedes Quellsystem über eine eigene Schnittstelle, an die das Ziel-CMS angebunden ist. Das Ziel-CMS seinerseits muss an jede dieser Schnittstellen angepasst werden. Es benötigt also für jedes Quellsystem eine spezielle seperate Schnittstelle (es sei denn die Schnittstellen zweier Quellsysteme sind identisch).

Außerdem wird in der Abbildung deutlich, dass die Datensätze im Ziel-CMS zwischengespeichert werden können, um so unabhängig von den Quellsystemen verarbeitet werden zu können, oder eine höhere Performanz beim Datenzugriff zu erzielen. Es kann also als ein Caching der Datensätze eingesetzt werden.

Die Aktualität der Daten ist durch diese Integrationsvariante gewährleistet und es kann auch durch die Quellsysteme gesteuert werden, welche Daten zur Verfügung gestellt werden sollen.

Je nach Ausgestaltung der Schnittstellen ist es möglich, den Datenfluss als push-, oder pull-Mechanismus zu implementieren. Da allerdings die Quellsysteme oftmals bereits über entsprechende Schnittstellen verfügen, die es normalerweise nicht ermöglichen, ihre Daten aktiv an andere Systeme zu transferieren, muss der push-Betrieb entsprechend der Anforderungen des Gesamtsystems auf die Schnittstellen der Quellsysteme aufgesetzt werden.

Die Integration über Schnittstellen besitzt noch einen Vorteil gegenüber der reinen Datenreplikation. Es ist möglich, neben dem Datenaustausch auch Operationen in den Quellsystemen auszuführen. Dies ermöglicht es, durch die Schnittstellen kontrollierte Manipulationen auf den Daten und spezielle Systemoperationen auszuführen. Näheres dazu findet sich in Kapitel 4.4. Zusätzlich können Datensätze aus dem CMS an die Quellsysteme übertragen werden, um dort etwaige neue Daten und Änderungen an Datensätzen, die im CMS vorgenommen wurden, zur Verfügung zu stellen.

Vorteile der Integrationsvariante:

- Es treten keine Redundanz- und Konsistenzprobleme auf.
- Die Variante ist effizient, wenn große Datenmengen zur Verfügung gestellt werden sollen.
- Es kann ein Caching der Daten eingesetzt werden.
- Die Datensätze können in die Quellsysteme zurückgeschrieben werden, die Variante ist also rückkanalfähig.
- Operationen können in den Quellsystemen ausgeführt werden.

Nachteile der Integrationsvariante:

- Ein push-Mechanismus ist nur mit großen Aufwand implementierbar.
- Es ist eine spezielle Schnittstelle für jedes Quellsystem notwendig.
- Die Variante ist Abhängig von der Verfügbarkeit der Quellsysteme.
- Da das CMS die Datensätze von den Quellsystemen abruft, müssen diese Kapazitäten für die Auslieferung der Datensätze bereitstellen.
- Durch push- und pull-Techniken, Rückkanalfähigkeiten und mögliche Operationen in den Quellsystemen kann das Gesamtsystem komplex und unübersichtlich werden.

Föderation

Die Föderation¹ unterscheidet sich grundlegend von der Replikation. Während bei der Replikation die Daten in das CMS übernommen werden, sind sie bei der Föderation nur in den Quellsystemen vorhanden. Um den Zugriff auf die Daten zu ermöglichen, wird zwischen das CMS und die Quellsysteme eine abstrakte föderierte Darstellungsschicht eingezogen. Das CMS kann über diese Darstellungsschicht auf die Daten in den Quellsystemen zugreifen. Die Darstellungsschicht stellt dabei dem CMS eine einheitliche Schnittstelle zur Verfügung, greift ihrerseits mittels spezieller Schnittstellen auf die Quellsysteme zu und transformiert die Daten entsprechend für das CMS.

Das CMS verfügt selber über keine Kopie der einzelnen Datensätze. Alle Zugriffe auf die Datensätze erfolgt über die föderierte Darstellungsschicht - sie kann als eine Art virtuelles System angesehen werden, welches die Quellsysteme kapselt und eine einheitliche Schnittstelle für das

¹ Föderalismus bezeichnet eine Staatsform, in der Gliedstaaten (Länder) einen Bundesstaat bilden. [Wikipedia 2005] Übertragend auf die Informatik werden also mehrere Systeme in einem zusammengefasst.

Quellsystem A Datensatz A Föderierte CMS Darstellungsschicht Quellssystem B Datensätze Datensatz B (Virtuelles System) Quellsystem C Datensatz C Kopie Datensatz A Kopie Datensatz B Datensatz C

CMS zur Verfügung stellt. Das Virtuelle System kann so gestaltet werden, dass es eine Zwi-

Abbildung 10: Datenintegration über Föderation in ein CMS

schenspeicherung der Datensätze vornimmt. So können Ausfälle von Quellsystemen kompensiert und schnellere Zugriffe auf die Datensätze ermöglicht werden.

Auch bei der Föderation stellen sich ähnliche Fragen wie bei der Replikation.

Welche Datensätze aus den einzelnen Quellsystemen zu Verfügung gestellt werden, sollte durch die Quellsysteme selber festgelegt werden, um den Abruf nicht öfffentlicher Daten zu verhindern. Es ist zusätzlich möglich, dass das zwischengeschaltete virtuelle System einen Filter realisiert, um nur bestimmte Datensätze an das CMS oder nur bestimmte Anfragen an die Quellsysteme weiterzuleiten. Dies ist dann sinnvoll, wenn unterschiedliche CMS auf das virtuelle System zugreifen, oder verschiedene Benutzer des CMS einen gefilterten Zugriff auf die Quelldatensätze erhalten sollen. Es kann also mehreren Mandanten¹ der Zugriff auf die Daten ermöglicht und dies gezielt gesteuert werden. Um den Datenaustausch zu gewährleisten muss eine Schnittstelle zwischen den Mandanten und dem virtuellen System definiert werden, als auch je eine Schnittstelle zwischen dem virtuellen System und jedem Quellsystem. Dazu müssen jeweils entsprechende Datenformate definiert werden, welche durch das virtuelle System ineinander gewandelt werden.

Abgleichzyklen, wie sie bei der Replikation möglich sind, existieren bei der Föderation in dem Sinne nicht. Da die Datensätze direkt von den Quellsystemen abgerufen werden, sind diese immer aktuell. Durch die Quellsysteme kann die Auslieferung der aktuellen Datensätze unterbunden und nur etwas ältere Datensätze ausgeliefert werden (zum Beispiel zeitversetzte Bör-

¹ Der Mandant ist die oberste Ordnungsinstanz in dem IT-System, stellt eine datentechnisch und organisatorisch abgeschlossene Einheit im System dar und wird über Parameter gesteuert bzw. definiert. [Wikipedia 2005]]

4.3 Datenintegration in CMS

senkurse). Dem CMS stehen also immer die Daten zur Verfügung, die gerade in den Quellsystemen aktuell sind und ausgeliefert werden sollen. Verfügt das virtuelle System über einen Datenspeicher, ist es möglich, dass zwischen den Quellsystemen und dem virtuellen System eine Replikation eingesetzt oder ein Chaching der Daten implementiert wird. Dadurch werden die Quellsysteme von externem Zugriff entlastet.

Die Kommunikation von CMS und virtuellem System kann nur im pull-Betrieb erfolgen. Das CMS holt sich die Daten zu dem Zeitpunkt, an dem sie benötigt werden (zum Beispiel bei der Generierung einer Webseite). Die Daten müssen dazu nicht im CMS zwischengespeichert werden, es sind also quasi "live"-Daten aus den Quellsystemen. Verfügt das virtuelle System über einen eigenen Datenspeicher, dann können zwischen den Quellsystemen und ihm die Daten auch in push-Betrieb abgeglichen werden. Dies entspräche dann der Replikation.

Aufgrund dessen, dass das CMS über das virtuelle System einen indirekten Zugriff auf die Quellsysteme hat, ist es, neben der reinen Abfrage der Datensätze, auch möglich Operationen auf diesen Datensätzen in den Quellsystemen auszuführen. Dies würde die Datenintegration allerdings übersteigen und wird in Kapitel 4.4 zur Dienstintegration behandelt werden.

Konsistenzprobleme treten bei der Föderation normalerweise nicht auf. Da keine Datensätze im CMS zwischengespeichert werden, müssen sich nur die Quellsysteme um die Konsistenz ihrer eigenen Daten kümmern. Das virtuelle System muss sich nur dann mit der Datenkonsistenz befassen, wenn es über einen eigenen Datenspeicher verfügt und die Quelldaten cached.

Sollen mehrere CMS an das virtuelle System angebunden werden, ist es notwendig, dass darin eine entsprechende Zugriffssteuerung und Verwaltung implementiert ist. Diese ist notwendig, damit jedes angebundene CMS nur auf Daten zugreifen kann, die für das jeweilige CMS zugänglich sein sollen.

Vorteile der Variante

- Es treten keine Konsistenzprobleme auf, da keine redundanten Daten vorliegen.
- Es können beliebig viele Quellsysteme verwendet werden.
- Die Quellsysteme bieten oftmals eine fertige API um die Schnittstelle zum virtuellen System leichter implementieren zu können.
- Die Struktur ist einfach zu überschauen, wenn nur Daten abgerufen werden.
- Neben dem Abruf der Daten können auch Operationen auf den Quelldaten vom CMS ausgeführt werden.
- Es ist nur eine definierte Schnittstelle zum CMS vorhanden. Über diese Schnittstelle können weitere Systeme angebunden werden.
- Die Schnittstellen für den Datenaustausch zwischen Quellsystemen, virtuellem System und CMS sind einfach und schnell zu implementieren.
- Es treten keine Bandbreitenprobleme auf, wenn die Quellsysteme große Datenmengen zur Verfügung stellen, da keine Replikate der Daten im CMS vorhanden sind.

Nachteile der Variante

- Abhängigkeit des CMS von der ständigen Verfügbarkeit der Quellsysteme. Steht ein Quellsystem nicht zur Verfügung, oder ist die Leitung gestört, dann stehen auch dessen Daten nicht zur Verfügung.
- Bei großer Auslastung des CMS müssen die Quellsysteme entsprechend ausgelegt, oder ein Caching implementiert sein.
- Das virtuelle System kann je nach Anforderung komplex werden. Je mehr und je unterschiedlichere Quellsysteme und CMS angebunden werden sollen, umso höher ist der Aufwand zur Planung und Umsetzung.
- Da die Daten von den Quellsystemen abgerufen werden, müssen die Kosten für die Verbindungen und den Datentransfer beachtet werden, die bei großen Datenmengen schnell ansteigen können.

4.3.2 Strukturelle Komponenten für die Datenintegration

Prinzipiell eignet sich jedes CMS dazu Daten zu integrieren. Für diese Integration muss das CMS einige Voraussetzungen erfüllen. Diese Voraussetzungen sind im Folgenden dargestellt.

• Freie Definition von Dokumenttypen:

Das CMS muss darauf eingestellt werden können, dass die zu integrierenden Quelldaten unterschiedliche Eigenschaften besitzen und in unterschiedlichen Strukturen vorliegen können.

Dies bedeutet, dass die Datenfelder der Quell-Datensätze in den Dokumenttypen des CMS abgebildet werden können. Es muss also möglich sein, Dokumenttypen im CMS zu definieren, die die Textfelder, numerische Felder, Binärdatenfelder und ähnliches der Quelldatensätze aufnehmen können. Eventuell muss es das CMS ermöglichen, dass der Datentyp einiger Datenfelder der Quelldatensätze gewandelt werden kann, damit sie in das CMS integriert werden können.

Um eventuelle Abhängigkeiten zwischen Datensätzen beziehungsweise Verknüpfungen von Datensätzen untereinander abzubilden, muss das CMS die Möglichkeit bereitstellen, verschiedene integrierte Datensätze in Verbindung zueinander zu stellen.

• Importschnittstellen:

Die Importschnittstellen realisieren die Abbildung beziehungsweise Transformation der eingehenden Quelldaten in die Dokumente des CMS. Zu einer Importschnittstelle muss immer ein Pondon, die Exportschnittstelle, des Quellsystems der Daten existieren. Die Schnittstellen müssen dabei die Festlegungen für die Datenfelder der Quellsysteme und die Dokumenttypen des CMS beachten und die Transformation der Datenfelder realisieren.

Die im CMS verfügbaren Importschnittstellen legen fest, welche Formate die Datensätze haben können, die integriert werden. Zumeist werden die Datensätze für den Transport in ein Zwischenformat transformiert, welches von beiden Schnittstellen unterstützt wird. Schnittstellen können dafür verwendet werden, um externe Daten in den Datenspeicher des CMS zu importieren, oder sie zu der Zeit, zu der sie benötigt werden, über entsprechende Kanäle bei den Quellsystemen abzufragen und zur Verfügung zu stellen. Dies entspricht einer redundanten beziehungsweise einer nicht redundanten Datenhaltung.

Die Schnittstellen müssen zusätzlich noch eine zentrale Funktion erfüllen: Sie müssen Sicherheitsmechanismen realisieren, die erstens sicherstellen, dass nur bekannten Systemen oder Benutzern der Zugriff gewährt wird. Zweitens dürfen beim Quellsystem nur freigegebene Daten zur Verfügung gestellt werden, die möglicherweise auch nur bestimmten CMS oder Benutzern zugänglich sein sollen. Drittens muss bei einer Rückkanalfähigkeit des Quellsystems sichergestellt sein, dass sensible Daten nicht durch ein CMS geändert werden können und nur bestimmte Möglichkeiten diesen Systemen oder deren Benutzern gewährt werden.

Das CMS seinerseits muss über Schnittstellen verfügen, um alle notwendigen Quellsysteme anbinden zu können. Dabei ist es sinnvoll, wie im Kapitel 4.1 "Integrationstopologien" ebenfalls dargestellt, dass diese Schnittstellen gegenüber dem CMS gekapselt werden und dem CMS nur diese kapselnde Schnittstelle zur Verfügung gestellt wird. Die Kapselung fungiert dann als ein zentraler Verteiler, der seinerseits auf die Quellsysteme zugreift, der Zugriff aber nur vom CMS aus erfolgt.

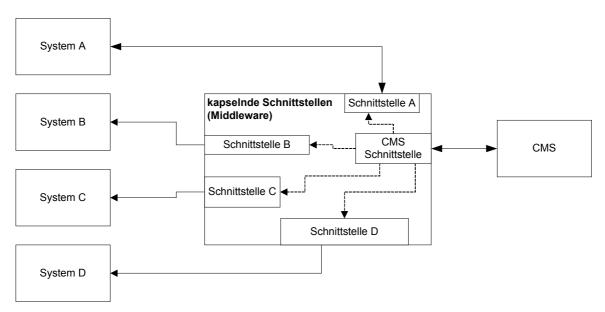


Abbildung 11: kapselnde Schnittstellen zwischen Quellsystemen und CMS

Wie in Abbildung 11 zu erkennen, wird die Komplexität und Verwaltung der Schnittstellen vor dem CMS gekapselt und komplett in der Middleware¹ realisiert. Somit muss

¹ Middleware bezeichnet in der Informatik anwendungsunabhängige Technologien, die Dienstleistungen zur Vermittlung zwischen Anwendungen anbieten, so dass die Komplexität der zugrundeliegenden Applikationen und Infrastruktur verborgen wird. [Wikipedia 2005]

sowohl das CMS, als auch jedes beteiligte Quellsystem an jeweils nur eine Schnittstelle angebunden werden, was die Verwaltung vereinfacht und Übersichtlichkeit auch bei komplexen Strukturen wahrt. Die in Abbildung 11 dargestellte Middleware ist ein eigenständiges System, in welches die notwendigen Systemintegrationen ausgelagert werden. Die Middleware führt alle anzubindenen Syteme zusammen und stellt eine einheitliche Zugriffsschnittstelle für unterschiedliche Endsysteme, in diesem Fall ein CMS, zur Verfügung. Da Middlewaresysteme über umfangreiche Möglichkeiten der Daten- und Dienstintegration verfügen, werden diese näher im Kapitel 4.4 "Dienstintegration" betrachtet. In [Wikipedia 2005] wird Aufbau und Struktur von Middleware präzise erläutert und es werden diverse Beispielsysteme vorgestellt.

Die Zugriffsschnittstelle, in diesem Fall die CMS Schnittstelle, kann, je nach Anforderungen und Einsatz, verschiedene Ebenen der Abstraktion implementieren und dem CMS eine mehr oder minder abstrakte Sicht auf die verfügbaren Quellsysteme ermöglichen.

• Die Quellsysteme sind dem CMS verborgen.

Art und Anzahl der Quellsysteme, die Daten zur Verfügung stellen, sind im CMS nicht bekannt und es kann auch nicht direkt auf eines der Quellsysteme zugegriffen werden. Alle Quellsysteme werden in der Middleware abgebildet und über die CMS-Schnittstelle gekapselt. Es können beliebig weitere Quellsysteme und deren Daten angebunden werden, ohne im CMS zusätzlich Anpassungen vornehmen zu müssen. Dem CMS werden die Daten über Zugriffsmethoden zur Verfügung gestellt, welche die Daten immer in der gleichen Struktur liefern beziehungsweise entgegennehmen, egal aus welchem Quellsystem diese Daten stammen oder in welches Zielsystem sie transferiert werden sollen. Da dem CMS nur die vorgegebenen Interaktionsmöglichkeiten mit der CMS-Schnittstelle zur Verfügung stehen, beschränkt sich der Aufwand der Integration in das CMS auf diese Schnittstelle. Weitere CMS können ebenfalls indirekt über die Schnittstelle mit den Quellsystemen interagieren. Die Kapselung stellt neben den Interaktionsmöglichkeiten auch sicher, dass alle Operationen korrekt auf den richtigen Quellsystemen ausgeführt werden und keine Inkonsistenzen auftreten.

• Die Quellsysteme sind für das CMS sichtbar.

Im CMS ist bekannt, welche Quellsysteme angebunden sind, über welche Daten diese verfügen und es kann auf jedes getrennt zugreifen. Auf diese Weise kann das CMS intern steuern, welche Möglichkeiten es seinen Nutzern für welche Daten der Quellsysteme anbietet. Daten von rückkanalfähigen Systemen können zum Beispiel von den Nutzern bearbeitet werden und gegebenenfalls zurückgespielt werden. Durch die vielfältigeren Möglichkeiten fällt die Implementation im CMS komplexer aus, da unter Umständen für jedes Quellsystem andere Möglichkeiten zur Verfügung stehen, die

4.3 Datenintegration in CMS

das CMS bedienen muss. Der Aufwand im CMS ist also höher und muss für jedes CMS und weitere Quellsysteme, welche angebunden werden sollen, speziell implementiert oder angepasst werden. Das CMS muss in diesem Fall die Steuerung der möglichen Operationen selbst leisten und die Vermeidung von Inkonsistenzen und dergleichen sicherstellen.

Die in Abbildung 11 dargestellt Middleware verfügt über unterschiedliche Schnittstellen zu mehreren Systemen. Da die CMS Schnittstelle indirekt auf die Repositories¹ aller angebundenen Systeme zugreift, lässt sich der Zugriff auf den Inhalt der Repositories vereinheitlichen. In [Spilker 2004] wird dazu eine Lösung angedacht. Der Zugriff auf die Inhalte aller Systeme wird über eine Programmierschnittstelle vereinheitlicht. Die Repositories aller Systeme werden mit den gleichen Methoden transparent vom CMS angesprochen. Alle Repositories können dann bei der Implementation gleich behandelt werden. Darunter wird in [Spilker 2004] "Content Unification" verstanden. [Spilker 2004] stellt dar, wie mit Unterstützung des Java Content Repository [JSR 170] unterschiedliche Repositories verschiedener Systeme in ein CMS integriert werden können. Dies wird am Beispiel des CMS CoreMedia SCI dargestellt. Als Referenzimplementierung ist bereits [Jakarta Slide] der Apache Group umgesetzt, welches die beschriebene einheitliche Schnittstelle zu unterschiedlichen Repositories zur Verfügung stellt. Die in Abbildung 11 dargestellte Middleware ließe sich auf diese Weise standardisieren und die angebundenen Respositories über die gleiche Schnittstelle in beliebige CMS integrieren.

Möglichkeiten zur Strukturierung der integrierten Daten:

Damit die Redakteure des CMS mit den, von den Quellsystemen gelieferten, Daten umgehen können, ist es notwendig, dass die Daten strukturiert im CMS abgelegt und den Redakteuren strukturiert als Dokumente zur Verfügung gestellt werden².

Für die Verwaltung der Dokumente durch Redakteure ist es demnach sinnvoll, die Dokumente in ordnerähnlichen Strukturen abzulegen und darzustellen. Dies ermöglicht einen hierarchisch strukturierten Zugang zu den Dokumenten. Die auf den Quelldatensätzen basierenden Dokumente müssen diesen Strukturen automatisch zugeordnet werden. Wie diese Zuordnung realisiert wird, ist abhängig von den Quellsystemen, deren Datensätzen und der Menge der Datensätze. Um Dokumente sinnvoll zuzuordnen, ist es notwendig die zugrundeliegenden Datensätze durch mehrere Filter laufen zu lassen. Die Filter überprüfen die Eigenschaften der Datensätze, anhand dieser jeder Datensatz an einen definierten Ausgabepunkt gelangt. Dieser Ausgabepunkt ist ein Knoten oder eine Position in der Datenstruktur des CMS, an welcher der Datensatz wiederzufinden ist. Diese Position ist sowohl von der Datenstruktur des CMS, als auch vom Datensatz abhängig.

¹ Ein Repository (deutsch: Lager, Depot), auch Repositorium genannt, ist eine Verzeichnisstruktur oder Datenbank, die Dateien inklusive Änderungsinformationen enthält. [Wikipedia 2005]

² Daten werden zusammengefasst als Datensätze, welche die inhaltliche Grundlage der Dokumente in einem CMS bilden.

Die Datensätze lassen sich nach unterschiedlichen Kriterien filtern, wodurch der Ausgabepunkt in der Datenstruktur des CMS automatisch ermittelt wird. Beispiele für mögliche Filterkriterien sind:

- · das Quellsystem,
- spezielle Wertebelegungen von Datensatz Eigenschaften,
- Metadaten, um die Datensätze der Quellsysteme angereichert sind oder von dem CMS angereichert werden,
- Indizierung der in den Datensätzen enthaltenen Texte, sowie
- Typen der Datensätze (Text, Bild, Audio, Video, ...).

Durch diese und weitere mögliche Filterkriterien können die integrierten Datensätze logisch in die vorhandene Struktur des CMS eingebunden werden. Dadurch wird eine redaktionelle Weiterverarbeitung erleichtert.

Wird eine große Menge von Datensätzen integriert, müssen die Datensätze über Suchfunktionen auffindbar sein. Sind Datensätze für die redaktionelle Weiterverabreitung aufgrund von Filterkriterien nicht relevant, werden solche Datensätze nicht in das CMS integriert. Einige Beispiele von notwendigen Suchfunktionen und -optionen in den integrierten Datensätzen sind:

- Stichwortsuche in Metadaten,
- · Sichtwortsuche in Texten und
- Datum und Uhrzeit der Integration.

Je nach Anwendung und Anforderungen des redaktionellen Workflows müssen hier zusätzliche Prozesse realisiert sein, wie zum Beispiel eine automatische Benachrichtigung über neue oder geänderte Datensätze.

Verwaltung der integrierten Daten:

Integrierte Datensätze sollten sich aus Sicht der Redakteure im CMS möglichst nicht von CMS-eigenen Datensätzen unterscheiden. Es sollten die gleichen Methoden und Funktionen für die Redakteure zur Verfügung stehen. Das CMS muss dafür sorgen, dass die integrierten Datensätze entsprechend gesondert behandelt werden. Dies ist zum Beispiel notwendig, wenn eine Rückkanalfähigkeit unterstützt werden soll und die geänderten Datensätze nach der Bearbeitung im CMS den Quellsystemen wieder verfügbar gemacht werden sollen.

Diese Verhaltensweise des CMS hat den Vorteil, dass sich sämtliche Integrationen für die Redakteure transparent darstellen. Ein Redakteur muss nicht auf spezielle Vorgehensweisen oder Notwendigkeiten bei integrierten Dokumenten achten, sondern bearbeitet und verwendet sie genau wie alle anderen Dokumente. Um diese Transparenz für den Redakteur herzustellen, bedarf es einen hohen Aufwand im CMS.

Die angeführten Voraussetzungen bilden die Basis dafür, dass Daten von externen Quellsystemen in ein CMS integriert, redaktionell weiterverarbeitet und publiziert werden können. Um eine die Implementation der Integration umzusetzen, müssen zwei wichtige Aspekte betrachtet werden. Dies sind zum einen die Datensynchronisation und zum anderen die Datenredundanzen, die in den folgenden Kapiteln behandelt werden.

4.3.3 Datensynchronisation

In diesem Kapitel wird auf die Problematik eingegangen, die auftritt, wenn Daten aus Quellsystemen in ein CMS integriert, also zwischen mehreren Systemen ausgetauscht, dort verwaltet und gepflegt werden. Es stellt sich dann die Frage, wie die Daten anderer Datenquellen im CMS aktuell gehalten werden können, welche Daten-Abgleichmechanismen notwendig sind, und wie die Konsistenz gleicher Datensätze in den verschiedenen Systemen sichergestellt werden kann.

Bei der Datensynchronisation müssen die Quelldaten dem CMS zur Verfügung gestellt und Änderungen der Daten mit dem CMS beziehungsweise dem Quellsystem abgeglichen (konsistent gehalten) werden. Ziel der Synchronisation ist es, in der gesamten Synchronisationsstruktur die beteiligten Daten in einen konsistenten Zustand zu bringen. Sind die Daten entsprechend synchronisiert, ist die Verfügbarkeit in einem breiteren Umfeld gewährleistet.

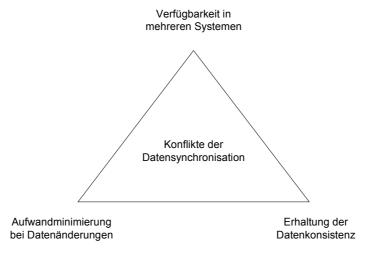


Abbildung 12: Zielkonflikte bei der Datensynchronisation aus [Thorsten Reincke 1999]

Die Ziele zur jeweiligen Vereinfachung der drei Parameter: Verfügbarkeit der Daten, Datenkonsistenz und Aufwandsminimierung bei Änderungen stehen dabei in Konflikt zueinander. Wie Abbildung 12 veranschaulicht, bewirkt jede Verbesserung einer dieser Parameter eine Beeinträchtigung bei der Zielerreichnung der beiden anderen Parameter. Beispielsweise bedeutet eine höhere Verfügbarkeit der Daten, durch den Einsatz mehrerer verteilter Systeme, dass die Einhaltung der Datenkonsistenz und der Aufwand bei Änderungen der Daten komplizierter werden. Dies resultiert daraus, dass die Konsistenz über mehrere Systeme hinweg gesichert werden muss und Änderungen eines Datensatzes in allen System umgesetzt werden muss.

Um den Aufwand für die Lösung der einzelnen Ziele zu minimieren und gegeneinander abzuwägen, existieren mehrere Synchronisations- beziehungsweise Replikationsverfahren. Diese Verfahren werden primär nach folgenden Kriterien unterschieden, die sich auf die Konsistenzerhaltung beziehen:

- optimistische Verfahren und
- pessimistische Verfahren.

Optimistische Verfahren gehen bei der Synchronisation davon aus, dass Inkonsistenzen nur selten auftreten und diese beim Auftreten erkannt und entsprechend aufgelöst werden können. Dabei ist es notwendig, dass diese Verfahren Wissen über den Synchronisationsprozess besitzen, um in Teilprozessen Konsistenzprobleme aufspüren und beseitigen zu können. Pessimistische Verfahren gehen davon aus, dass Inkonsistenzen an allen Stellen auftreten, wo immer dies möglich ist. Um diese Inkonsistenzen zu vermeiden, schränken pessimistische Verfahren dazu Synchronisation entsprechend ein. Auf die beiden Verfahrensansätze wird im Folgenden noch einmal detaillierter eingegangen.

4.3.3.1 Pessimistische Verfahren

Da pessimistische Verfahren davon ausgehen, dass Inkonsistenzen auftreten, müssen sie sicherstellen, dass jede Operation die Daten inkonsistent macht gesperrt wird und nur sichere, konsistenzerhaltende Operationen zugelassen werden. Daraus resultieren die folgend dargestellten Ansätze. ([Thorsten Reincke 1999], [Lenz 97] und [Beuter Dadam 1996])

• eager Replication / ROWA

Beim naheliegendsten Verfahren, dem eager Replication oder Read-One-Write-All (ROWA) Verfahren, werden alle Datensätze, die in verschiedenen Systemen vorhanden sind, gleichzeitig aktualisiert. Es wird also ein synchroner Abgleich der Daten durchgeführt. Dieses Verfahren setzt voraus, dass sämtliche beteiligte Systeme erreichbar sind und die Aktualisierung der Daten durchführen können. Fällt eines der Systeme aus oder ist die Verbindung unterbrochen, kann dieses Verfahren nicht korrekt angewendet werden und es muss für diesen Abgleich auf ein anderes Verfahren umgeschaltet werden. Dieses Verfahren eignet sich besonders für sensible und hochaktuelle Daten, die umgehend und immer aktuell auf allen beteiligten Systemen zur Verfügung stehen müssen. Es wird immer auf die aktuellen Daten zugegriffen, egal in welchem System. Dies wird mit dem Nachteil erkauft, dass es einen recht großen Verwaltungsaufwand in den beteiligten Systemen bedarf, um die Daten in den anderen Systemen entsprechend zu aktualisieren.

• Primary – Verfahren

Bei diesem Verfahren existiert ein System, welches als primäres oder ausgezeichnetes System festgelegt ist und Änderungen der Daten nur dort durchgeführt werden. Alle beteiligten Systeme ändern die Daten in diesem primären System. Sind Daten geändert, dann initiiert das primäre System den Synchronisationsprozess, wobei an alle beteiligten Systeme die geänderten Daten ausgeliefert werden. Die Daten werden asynchron verteilt, was zur Folge hat, dass nicht in allen Systemen die aktuellsten Daten zur Verfügung stehen. Die Verwaltung der Synchronisation ist übersichtlich und robust gegen Fehler. Allerdings besteht der Nachteil, dass die Verbindung zum primären System vorhanden sein muss und dieses System nicht ausgefallen sein darf, wenn Daten geändert werden sollen. Wenn einer dieser Umstände eintritt, können keine Daten geändert werden.

Das Verfahren kann leicht abgeändert werden, indem nicht ein ganzes System als primär oder ausgezeichnet angesehen wird, sondern nur jeweils der Datensatz, der Synchron-Kopien in anderen Systemen besitzt. Somit können die primären Datensätze über die Systeme verteilt sein und jedes System stößt die notwendigen Synchronisationsprozesse an. Dies bedeutet, dass jedes beteiligte System darüber informiert sein muss, für welchen Datensatz wo die primäre Kopie vorhanden ist, was einen hohen Verwaltungsaufwand bedeutet. Der Vorteil ist, dass auch bei Ausfall eines Systems bestimmte Daten trotzdem weiter geändert werden können, da nicht alle Primärkopien in einem System liegen.

• Voting - / Quorum - Verfahren

Dieses Verfahren ist das komplizierteste und lässt sich auch im größten Maße variieren. Bei diesem Verfahren können Daten nur dann gelesen oder verändert werden, wenn eine bestimmte Menge der beteiligten Systeme diesem zustimmt und entsprechend ihre Daten freigeben oder aktualisieren. Systeme, die ihre Daten nicht sofort synchron aktualisieren können, werden in einen asynchronen Synchronisationsprozess eingebunden.

Dieses Verfahren erlaubt diverse Möglichkeiten der Justierung. Es ist zum Beispiel möglich, dass verschiedene Systeme über eine unterschiedliche Gewichtung ihrer Stimme verfügen und so einen größeren oder kleineren Einfluss auf die Datenfreigabe oder Datenänderung nehmen können. Zusätzlich können die Schwellen, wann der lesende oder schreibende Zugriff erlaubt ist, unterschiedlich sein und zum Beispiel in Abhängigkeit der verfügbaren Systeme variiert werden. Für bestimmte Datensätze dürfen einige Systeme keine Stimme abgeben, oder es dürfen externe Systeme, die über keine Kopie des Datensatzes verfügen, in der Rolle eines Schiedsrichters mitstimmen. Es lassen sich diverse Möglichkeiten zusammenstellen, die an die gegebenen Notwendigkeiten angepasst werden müssen.

4.3.3.2 Optimistische Verfahren

Optimistische Verfahren gehen davon aus, dass die angestrebten Operationen auf den verteilten Daten durchgeführt werden können. Sollte dabei festgestellt werden, dass Inkonsistenzen bei der Ausführung der Operation auftreten, können diese Verfahren die Operation entweder rückgängig machen oder die Inkonsistenzen beseitigen.

Zwei wichtige optimistischen Verfahren sind die Log Transformation und das Data-Patch-Verfahren. Bei beiden Verfahren werden die ausgeführten Operationen als Transaktionen betrachtet und die Informationen über diese genutzt, um die Transaktionsausführung zu optimieren und Inkonsistenzen zu beseitigen.

• Transformation Log

Bei diesem Verfahren werden Änderungen an Daten asynchron an die beteiligten Systeme weitergeleitet. Bei auftretenden Inkonsistenzen können diese Änderungs-Transaktionen entweder zurückgesetzt, oder erneut ausgeführt werden. Die erneut auszuführenden Transaktionen werden in einer Art Logbuch gehalten und können dort zur Ausführung abgerufen werden. Um den Umfang dieses Logbuches möglichst gering zu halten, werden folgende Regeln zur Minimierung der enthaltenen Transaktionen angewandt:

Kommutativität

Benachbarte Transaktionen können in ihrer Ausführungsreihenfolge vertauscht werden, wenn sie kommutativ sind, also dadurch keine Inkonsistenzen resultieren.

Überdeckung

Werden alle Auswirkungen einer Transaktion durch die folgende wieder überschrieben, dann kann die erste Transaktion ersatzlos gestrichen werden.

Zurücksetzung

Heben zwei benachbarte Transaktionen sich komplett in ihrer Wirkung auf, dann können beide ersatzlos gestrichen werden.

Durch diese Regeln sinkt die Anzahl der Transaktionen, die nochmalig ausgeführt werden müssen, und die Effizienz der Integration nimmt zu. Ein Nachteil ist der Verwaltungs- und Strukturierungsaufwand. Zusätzlich ist die notwendige Mehrfachausführung bestimmter Transaktionen ineffizient und die Anzahl der Wiederholungen von nicht vorhersehbaren Parametern abhängig (z.B. Benutzereingriffe, Systemumfeld, Hardwareverfügbarkeit, usw.).

• Data-Patch-Verfahren

Beim Data-Patch-Verfahren werden die beteiligten Systeme in Partitionen zusammengefasst, sodass ist in jeder Partition mindestens ein beteiligtes System vorhanden ist.

Wie der Name des Verfahrens schon andeutet, werden nicht die Transaktionen auf den Daten überwacht, sondern die Daten so verändert, das ein konsistenter Gesamtzustand entsteht. Es werden dazu beim Entwurf der beteiligten Systeme beziehungsweise des Ge-

samtsystems Regeln festgelegt, die steuern, wie aus unterschiedlichen Kopien der Datensätze in den beteiligten Systemen wieder ein einheitlicher Datensatz zu erzeugen ist. Diese Regeln legen die Vorgehensweisen für den Fall fest, dass ein Datensatz in nur eine Partition von beteiligten Systemen eingefügt wird (Einfügungsregel) beziehungsweise ein Datensatz in mehrere Partitionen eingefügt oder dort erzeugt wird (Integrationsregel). Diese beiden Regeln müssen jeweils zwischen zwei unterschiedlichen Partitionen festgelegt sein, wobei unterschiedliche Ausprägungen der Regel möglich sind, die folgend kurz dargestellt sind:

Einfügungsregel

Eine der folgend genannten Varianten der Einfügungsregel wird angewandt, wenn ein neuer Datensatz in einer Partition erzeugt wurde. Es gibt folgende Varianten der Regel:

· Keep-Regel

Wird ein Datensatz in einer Partition erzeugt, dann wird er auch in den anderen erzeugt.

Remove-Regel

Ein Datensatz wird gelöscht, wenn er nur in einer Partition erstellt wurde.

· Program-Regel

Ein angegebenes Programm wird mit der Konfliktlösung beauftragt.

Notify-Regel

Sind die Konflikte schwerwiegend, dann wird eine Person benachrichtigt, die den Konflikt manuell beseitigen muss.

Integrationsregel

Eine der Varianten der Integrationsregel wird angewandt, wenn ein Datensatz mit dem gleichen Schlüssel in mehreren Partitionen erzeugt oder in mehreren verändert wurde.

· Latest-Regel

Der zuletzt erzeugte oder geänderte Datensatz ist der gültige für alle Partitionen.

· Primary-Regel

Der Datensatz einer Partition beziehungsweise eines Systems ist der gültige für alle Partitionen.

Arithmetic-Regel

Auf Basis von festgelegten (mathematischen) Formeln wird ein neuer gültiger Datensatz aus den vorhandenen Datensätzen berechnet.

· Program-Regel

Ein angegebenes Programm wird mit der Konfliktlösung beauftragt.

· Notify-Regel

Sind die Konflikte schwerwiegend, dann wird eine Person benachrichtigt, die den Konflikt manuell beseitigen muss.

Dieses Verfahren ist das flexibelste, aber auch das komplexeste und aus diesem Grund bedarf es auch den höchsten Aufwand bei der Konzeption und Umsetzung. Zusätzlich müssen weitere Systeme in das bereits vorhandene Gesamtsystem und in die Konsistenzsicherung eingebunden werden.

Durch den Ansatz und die vielseitigen Ausprägungen der Regeln in diesem Verfahren ist es möglich, auftretende Inkonsistenzen zu erkennen und aufzulösen.

4.3.4 Datenredundanzen vermeiden

Das vorherige Kapitel beschäftigt sich mit dem redundanten Vorhandensein von Datensätzen in verteilten Systemen, den dabei auftretenden Problemen und möglichen Lösungsansätzen. Um die auftretenden Probleme zu lösen, werden mehr oder minder komplexe Systemerweiterungen benötigt, um mögliche Inkonsistenzen zu vermeiden oder aufzulösen. Diese Erweiterungen können vermieden werden, wenn die Datensätze nicht redundant vorhanden sind, sondern lediglich auf einem einzigen System vorgehalten werden.

Dies bedeutet allerdings, dass in einem anderem System bekannt sein muss, dass ein Datensatz in einem System existiert, und wie es darauf zugreifen kann. Um dies sicherzustellen werden Verweise eingesetzt. Verweise können über die bereits beschriebenen Möglichkeiten bei der Bereitstellung und Nutzung von Schnittstellen an den Systemen oder durch Föderation realisiert werden.

Der Vorteil bei der Verwendung von Verweisen besteht darin, dass jedes System sich nur mit der Verwaltung seiner eigenen Datensätze beschäftigen muss. Die Durchführung und Überwachung von Replikationsprozessen entfällt und Inkonsistenzen können nicht auftreten.

Folgend werden die Notwendigkeiten dargestellt, die ein Gesamtsystem mit Verweisen zur Vermeidung von Datenredundanzen sicherstellen muss und welche Probleme dabei auftreten können.

Um mit Verweisen arbeiten zu können, muss sichergestellt sein, das zum einen jedes System eindeutig angesprochen werden kann und zum anderen alle Datensätze eines Systems eindeutig identifizierbar sind. Dies bildet die Basis, um aus Fremdsystemen heraus auf Datensätze verweisen zu können. Weiterhin muss das System, in dem die Datensätze vorhanden sind, über eine Schnittstelle verfügen, über die die Datensätze von anderen Systemen abgerufen werden können. Je nach Konzeption und Einsatz des Gesamtsystems kann es notwendig sein, dass diese Schnittstellen rückkanalfähig sind. Über die Schnittstellen müssen die Datensätze im Quellsystem also aktualisiert, oder eventuell neue Datensätze erzeugt werden können. Die Schnittstellen können dabei im push-, pull-oder einem hybriden Verfahren definiert sein. Systeme, welche die Datensätze benötigen, können somit sowohl die Datensätze abrufen, als auch

über bestimmte Umstände (Änderung bestimmter Datensätze, etc.) vom Quellsystem informiert werden.

Die Schnittstellen können mit unterschiedlichen Systemen realisiert sein. Es können proprietäre API's zur Verfügung gestellt werden, die direkt von den Systemen verwendet werden. Weiterhin ist die Verwendung von Web-Services über HTTP möglich, als auch dateibasierende Verfahren wie FTP, SFTP oder Ähnliches.

Auch hier muss die Verfügbarkeit der verteilten Systeme sichergestellt sein.

Wie die Abbildung 13 zeigt, ist es für System A in diesem Fall weder möglich die Datensätze aus dem Quellsystem abzurufen, noch zu aktualisieren. Um Ausfällen dieser Art vorzubeugen, können die physikalischen Verbindungen zwischen den beteiligten Systemen redundant ausgelegt, oder bereits abgerufene Daten im abrufenden System zwischengespeichert (gecached) werden. Die Rückkanalfähigkeit ist allerdings bei jeder Verbindungsunterbrechung und jedem Ausfall des Quellsystems nicht mehr gegeben.

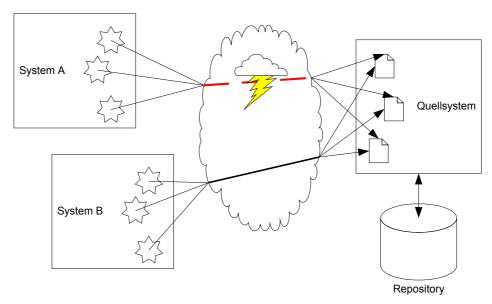


Abbildung 13: Unterbrechung der Verbindung bei Verweisen

Eine weitere Möglichkeit, dass Datensätze nicht mehr abgerufen werden können, besteht darin, dass das Quellsystem, oder einer seiner Komponenten, wie zum Beispiel das Repository,
ausfällt. Um einen solchen Ausfall zu kompensieren kann das Quellsystem, beziehungsweise
seine Komponenten, redundant ausgelegt werden. Hier greifen die zuvor beschriebenen
Möglichkeiten von Replikationsverfahren, um das Quellsystem redundant verfügbar und die
Datenbestände in den Repositories konsistent zu halten.

Sowohl die Quantität der redundanten Verbindungen, als auch der redundanten Quellsysteme kann den Anforderungen entsprechend angepasst werden. So ist es möglich fast beliebige Verfügbarkeiten der Quellsysteme sicherzustellen.

4.4 Dienstintegration in CMS

Im vorherigen Abschnitt wurde die Integration von Daten in CMS behandelt. In diesem Abschnitt soll die Frage beantwortet werden, wie sich Dienste in CMS integrieren lassen. Um dies zu beantworten, muss erst einmal geklärt werden, was der Unterschied zwischen Daten und Diensten ist.

Der Begriff "Daten" wurde bereits im vorherigen Abschnitt definiert (vgl. Kapitel 4.3). Der Begriff "Dienst" wird im Lexikon synonym mit der Erfüllung einer Pflicht behandelt. Im [Internet Lexikon 2005] lässt sich folgende Definition für den Begriff "Pflicht" finden:

Pflicht ist - systemtheoretisch ausgedrückt - die Erwartung eines Systems, dass ein anderes, untergeordnetes System ein bestimmtes näher bezeichnetes Verhalten praktiziert. Eine Pflicht ist also eine das Verhaltensrepertoire eines Systems einschränkende Sache. Ein System, welches diese Pflicht erfüllt, gibt etwas (von seiner Freiheit) auf.

Ein Dienst ist, laut Definition, die Erfüllung einer Pflicht eines bestimmten Systems, also ein von anderen Systemen erwartetes Verhalten und dessen Durchführung. Die Art und der Umfang des Dienstes wird dabei vom diensterbringenden System vorgegeben.

Bei der Integration eines Dienstes wird also nicht, wie bei der Integration von Daten, die Grundlage von Informationen (die Daten) in ein System integriert, sondern das Verhalten eines anderen Systems in ein System eingebunden oder in ihm abgebildet. Daraus folgt, dass bei einer Integration von Daten deren Einsatz und Verwendung einzig und allein vom integrierenden System gesteuert werden. Die Daten können in den Systemen also zu verschiedenen Zwecken dienen, wie ihrer Darstellung, Weitergabe, Verknüpfung mit anderen Daten und dergleichen.

Wird ein Dienst integriert, ist dessen Grundlage, seine verwendeten Daten, im Regelfall nicht verfügbar. Das integrierende System kann nur über die vom Dienst zur Verfügung gestellten Funktionalitäten, Methoden und Ausgaben verfügen, sie verwenden und nach außen hin darstellen. Der Hauptunterschied zwischen Daten- und Dienstintegration ist demnach, dass dienstzugehörige Daten bei der Dienstintegration vom integrierenden System nicht frei verwendet werden können, sondern nur im Rahmen der vom Dienst zur Verfügung gestellten Möglichkeiten.

4.4.1 Alternativen der Dienstintegration

Dienste lassen sich ähnlich wie Daten auf unterschiedliche Art und Weise in andere Systeme integrieren. Generell lassen sich zwei Integrationsansätze unterscheiden:

• Kopie des Dienstes

Der Dienst läuft komplett auf den Systemen, die den Dienst integrieren. Der Dienst wird dabei auf allen Systemen implementiert, die ihn integrieren. Um den Dienst zu erbringen, müssen die behandelten Daten von dienstintegrierenden System verwaltet werden. Diese

Behandlung der Daten (erzeugen, speichern, löschen, usw.) kann sowohl von dienstintegrierenden, als auch vom diensterbringenden System vorgenommen werden.

Wird die Datenbehandlung vom integrierenden System vorgenommen, kann man nicht mehr von einer Dienstintegration sprechen, da das Quellsystem¹ nicht mehr an der Erbringung des Dienstes beteiligt ist. Der Dienst wäre somit eine neuer, eigenständiger Dienst auf einem anderen (dem integrierenden) System.

Die Datenbehandlung kann aber auch von dem Quellsystem vorgenommen werden. Dieses System ist also dafür zuständig, dass die Verarbeitung der Daten durchgeführt wird. Hier lassen sich wieder die zuvor beschriebenen Möglichkeiten zur Datensynchronisation einsetzen (siehe Kapitel 4.3.3), die sicherstellen, dass die Daten an das System übermittelt und dort konsistenzerhaltend behandelt werden.

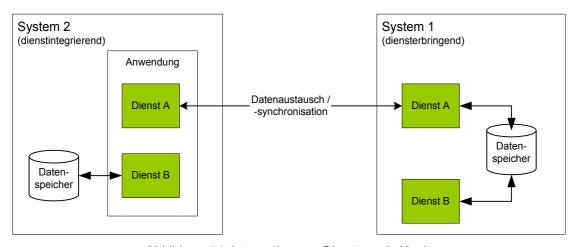


Abbildung 14: Integration von Diensten als Kopie

Die Abbildung 14 zeigt zwei Systeme, wobei System 1 die Dienste A und B zur Verfügung stellt, und System 2 diese als Kopie in eine ihrer Anwendungen integriert. Zu erkennen ist, dass nur für Dienst A die Datenverarbeitung im Quellsystem (System 1) durchgeführt wird. Dienst A auf System 2 bedient sich dabei der auf System 1 zur Verfügung gestellten Datenverarbeitung, indem die Daten ausgetauscht beziehungsweise synchronisiert werden. Die Datenverarbeitung für Dienst B wird getrennt in beiden Systemen durchgeführt und ist jeweils autonom. Dies veranschaulicht, weshalb nur die Integration von Dienst A als eine Integrationsmethode bezeichnet werden kann.

Die Vorteile dieses Integrationsansatzes sind die Kontrolle des integrierenden Systems über den Dienst, die nur eingeschränkte Abhängigkeit von anderen Systemen und die Flexibilität bei Änderungen. Von Nachteil sind der hohe Aufwand für die Planung und Implementation eines bereits vorhandenen Dienstes, generell das Vorhalten eines redundanten Dienstes und das Implementieren einer Datensynchronisation zwischen den beteiligten Systemen.

¹ Das diensterbringende System wird als Quellsystem bezeichnet und die dienstintegrierenden als Zielsysteme.

• An-/ Einbindung des Dienstes

Der Dienst läuft nicht auf dem Zielsystem, sondern auf dem System, welches diesen Dienst zur Verfügung stellt (dem Quellsystem). Die behandelten Daten sind dabei nur auf dem Quellsystem vorhanden, sodass sich das Zielsystem sich nicht um deren Verwaltung kümmern muss.

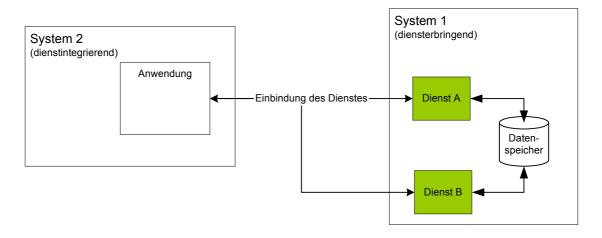


Abbildung 15: Integration von Diensten als Einbindung

Die Abbildung 15 zeigt, dass nur die Anwendung in System 2 die beiden Dienste aus System 1 einbindet beziehungsweise verwendet. Die Anwendung muss nur noch die Einund Ausgaben der Dienste darstellen. Die Dienstausführung und Datenbehandlung erfolgt hier ausschließlich auf dem Quellsystem (System 1).

Vorteil bei diesem Ansatz ist, dass keinerlei Planung und Implementation der Dienste auf dem Zielsystem notwendig ist und kein Dienst redundant implementiert wird. Außerdem kann die Verteilung und Verwaltung der Dienste wesentlich einfacher auf dem diensterbringenden System kontrolliert, gesteuert und überwacht werden. Nachteil ist, dass individuelle Anpassungen der Dienste nur mit Aufwand im diensterbringenden System umsetzbar sind und so eventuell Einfluss auf andere Systeme haben, die dessen Dienste integrieren.

Generell ist zu sagen, dass die Anbindung eines Dienstes vorteilhafter ist, als die Kopie, da Funktionalität nicht mehrmals implementiert werden muss, sondern von unterschiedlichen Anwendungen genutzt werden kann. Dies wird allerdings damit erkauft, dass die Individualisierung der Dienste in den verschiedenen Anwendungen erschwert wird.

Um die Individualisierung der Dienste in den Anwendungen zu erleichtern, werden teilweise Mischformen der beiden dargestellten Ansätze verwendet. Das diensterbringende System stellt also komplexe Vorgänge innerhalb des Dienstes als Einbindung zur Verfügung, die teilweise auch für bestimmte Anwendungen personalisiert werden können. Außerdem stellt es für bestimmte Daten, die innerhalb des Dienstes Verwendung finden, Möglichkeiten zum Austausch oder zur Synchronisation zur Verfügung. Auf diese Weise müssen komplexe Vorgänge in-

nerhalb der Dienste nicht redundant implementiert werden, aber es besteht dennoch die Möglichkeit individuelle Anpassungen in den Anwendungen vorzunehmen. Bestimmte Vorgänge werden also im Zielsystem abgebildet und entsprechend den Anforderungen verändert oder erweitert. Die Daten werden dann mit dem diensterbringenden System ausgetauscht. Hier spielen dann Daten- und Dienstintegration zusammen.

In folgenden Abschnitt wird nur die An- beziehungsweise Einbindung von Diensten und die erwähnte Mischform betrachtet, da die Kopie von Diensten als Integrationsansatz aufwändig und nicht relevant für die spätere Umsetzung des Projektes ist. Vorhandene Dienste sollen genutzt und nicht kopiert werden (vgl. Kapitel 6).

Für die konkrete Umsetzung der An- beziehungsweise Einbindung von Diensten werden häufig die folgenden drei Alternativen eingesetzt.

Die einfachste Form einen Dienst einzubinden ist es, den Dienst und dessen Ausgabe auf dem diensterbringenden System zu implementieren. Die Ausgabe wird im dienstintegrierenden System dargestellt, so dass es dem Benutzer erscheint, als ob er dort ausgeführt wird. Sehr gut funktioniert dies im Umfeld von Webanwendungen, bei denen vom diensterbringenden System erzeugter HTML-Code in einem Frame auf der Website des dienstintegrierenden Systems angezeigt wird. Von Vorteil ist es hier, dass kaum Implementationsaufwand notwendig ist, um den Dienst auf der Website darzustellen und die Sicherstellung der Diensterbringung vom diensterbringenden System gewährleistet wird. Von Nachteil ist, dass individuelle Anpassungen am Layout der Ausgabe sowie in der Funktionsweise des eingebundenen Dienstes nur vom Betreiber des diensterbringenden Systems vorgenommen werden können. Angewendet wird diese Form sehr oft bei Diensten, die auf Webseiten dargestellt werden. Zum Beispiel können Wetterinformationen auf der eigenen Website angezeigt werden, wie es von wetter.com¹ angeboten wird. Die Imobiliensuchmaschine immopool.de² bietet von der Suche, über die Darstellung von Immobilien, bis hin zur Kontaktaufnahme mit dem Anbieter einen Dienst an, der auf dieser Methode basiert.

Um eine Individualisierung der Darstellung zu erreichen, kann die Ausgabe eines Dienstes vom dienstintegrierenden System aufbereitet werden. Dazu muss vom diensterbringenden System eine Schnittstelle zur Verfügung gestellt werden, über die Funktionen des Dienstes ausgeführt werden können. Die Ausgabewerte, die aus den Funktionen resultieren, werden vom diensterbringenden System zurückgegeben und im dienstintegrierenden System angezeigt. Dabei müssen die Funktionen in der für die Erbringung des Dienstes notwendigen Reihenfolge ausgeführt und die benötigten Daten (zum Beispiel Eingabedaten der Nutzer) mit übergeben werden. Das diensterbringende System gibt also die möglichen Funktionsschritte, die an den Dienst zu übermittelnden Daten, sowie die zurückgegebenen Ausgabewerte der einzelnen Funktionen vor. Diese Art der Dienstintegration wird zum Beispiel häufig bei Abwicklungs-

¹ http://www.wetter.com

² http://www.immopool.de

diensten für Kreditkartenzahlung oder Lastschriftverfahren verwendet, die über eine Website angeboten werden.

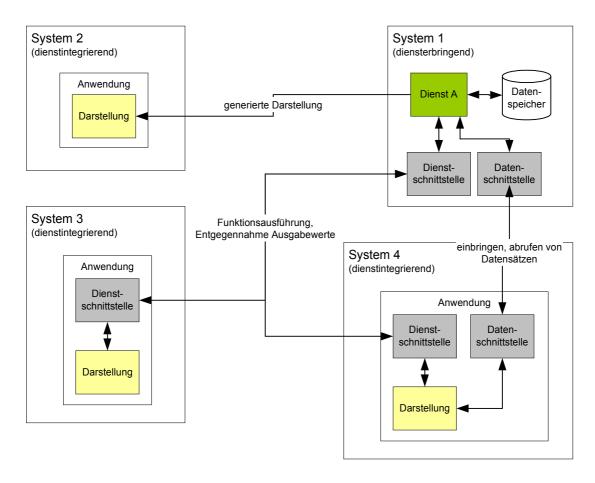


Abbildung 16: konkrete Alternativen bei der Dienstintegration

Die dritte Möglichkeit besteht darin, dass die Datenbehandlung im diensterbringenden System durchgeführt wird. Das diensterbringende System verfügt dazu über Schnittstellen, über die Daten in den Dienst eingebracht, abgerufen und verändert werden können. Diese Daten stehen dann innerhalb des Dienstes wieder zur Verfügung und können so von anderen Systemen genutzt werden. Das diensterbringende System gibt dabei vor, welche Datensätze durch welches dienstintegrierende System abgerufen und verändert werden können. Das diensterbringende System kann dabei zusätzlich über komplexere Teilfunktionen verfügen, die von den dienstintegrierenden Systemen verwendet werden können, um die verwendeten Datensätze zu verarbeiten. Dieses Verfahren hat den Vorteil, dass bestimmte Datensätze auf den dienstintegrierenden Systemen zur Verfügung stehen und dort anderweitig verwendet werden können, um zum Beispiel zusätzliche Funktionen damit zu erbringen, die der Dienst nicht leistet. Solche Datenschnittstellen werden unter anderem von CMS angeboten, über die Datensätze aus dem Repository des CMS abrufen beziehungsweise in das Repository eingespielt werden können. Zusätzlich stellt das CMS komplexere Funktionen zur Verfügung, wie eine Versionierung, automatische Verfahren zur Benachrichtigung und dergleichen.

Die Alternativen werden in der Abbildung 16 veranschaulicht.

Bewertung der Alternativen

Mit der Reihenfolge der aufgeführten Alternativen der Dienstintegration steigt auch der Aufwand, der betrieben werden muss, um die drei Alternativen einzusetzen. Alle drei Alternativen haben ihre Vor- und Nachteile, die mit dem Einsatzzweck, der zur Verfügung stehenden Infrastruktur und des vorhandenen Fachwissen abgewogen werden müssen.

Die erste Variante ist eindeutig vorzuziehen, wenn das Budget entsprechend klein, eigenes Fachwissen für die Einbindung des angestrebten Dienstes gering und der Anspruch an eine individuelle Anpassung des Dienstes, in Bezug auf Funktion und Darstellung, für das eigene System nicht besonders hoch ist.

Die zweite Variante stellt höhere Anforderungen an das vorhandene Fachwissen. Es müssen umfangreichere Planungen und Implementierungen im eigenen System vorgenommen werden, um den Dienst einzubinden. Im Gegensatz zur ersten Variante erkauft man sich dafür eine wesentlich höhere Flexibilität der Darstellung des Dienstes im eigenen System, sowie die Möglichkeit die Funktionsschritte des Dienstes für das eigene System individuell zu verwenden. Ein Beispiel wäre ein Assistent im eigenen System, welcher erst am Ende seiner Ausführung einen Funktionsschritt des Dienstes verwendet. Bedienung und Darstellung lässt sich so individuell gestalten, was bei der ersten Variante nicht oder nur mit Hilfe des Dienstanbieters möglich ist.

Die dritte Variante macht sich die Vorteile der zweiten zu Nutze und ermöglicht es gleichzeitig, bestimmte Datensätze des integrierten Dienstes im eigenen System zur Verfügung zu haben. Bedienung und Darstellung des integrierten Dienstes lassen sich so nicht nur individualisieren, sondern die Datensätze lassen sich mit systemeigenen Datensätzen verknüpfen und daraus zusätzliche Funktionen ableiten. Einsetzbar ist diese Variante zum Beispiel bei der Umsetzung eines Archivierungssystems, welches die Datensätze verschiedener Systeme verwalten und katalogisieren soll. Hier können dann, neben Datensätzen des Archivierungssystems und den Datensätzen der zu archivierenden Systeme, auch die Datensätze der zu archivierenden Systeme miteinander in Verbindung gebracht werden. So können Datensätze mit semantischen Beziehungen aus verschiedenen Systemen gemeinsam einem Nutzer präsentiert werden.

Die dritte Variante ist sicherlich die aufwendigste, bietet aber die größte Flexibilität, wird aber durch die Funktionen des diensterbringenden Systems weitreichend unterstützt. Aus diesen Grunde wird diese Variante später bei der Umsetzung des Projektes eingesetzt werden (vgl. Kapitel 6).

4.4.2 Dienstsynchronisation und Dienstredundanz

Wie der vorherige Abschnitt zeigt, treten Synchronisations- und Redundanzprobleme bei Diensten nur auf, wenn der selbe Dienst auf mehren Systemen implementiert ist und diesem Diensten auch die gleichen Daten zugrunde liegen, beziehungsweise zugrunde liegen sollen.

Redundant ausgelegte Dienste bieten den Vorteil, dass sie über größere Durchsatzraten bei Zugriffen verfügen. Um hier Synchronisationsproblemen aus dem Wege zu gehen, kann die Datenhaltung für die Dienste zentral, also nicht redundant realisiert werden. Allen Diensten steht also zum Beispiel ein und dieselbe Datenbank zur Verfügung. Hier müssen dann allerdings Sperrmechanismen implementiert werden, um zu vermeiden, dass mehr als ein Dienst gleichzeitig einen Datensatz ändert. Es wird eine Art Loadbalancing eines Dienstes vorgenommen. Er wird auf mehreren Systemen gleichzeitig betrieben, arbeitet auf den gleichen Daten und verhält sich auf allen Systemen gleich. Für GIS Dienste ist dieses klonen von Diensten sinnvoll, da sie sehr aufwändige geografische Berechnungen erfordern, und viele Zugriffe auf mehrere Systeme verteilt werden müssen. Falls der Datenspeicher (die Datenbank) bei großer Last ebenfalls einen Engpass darstellt kann auch dieser redundant ausgelegt werden und die Last über einen Loadbalancer verteilt werden.

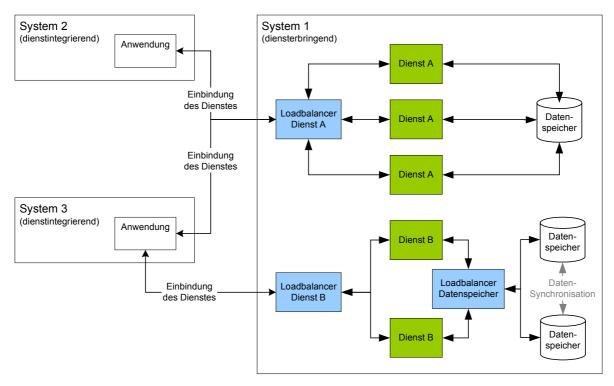


Abbildung 17: redundant ausgelegte Dienste mit loadbalancing

Die Abbildung 17 zeigt die Lastverteilung auf mehrere Dienstinstanzen mit unterschiedlichen Diensten (A und B), die über die beiden Lodbalancer angesteuert werden. System 1 ist hier als ein Systemverbund von mehreren Hardwareinstanzen zu verstehen, die die Loadbalancer, die Dienste und die Datenspeicher beherbergen. Bei Dienst B greifen die Dienstinstanzen auf einen

Loadbalancer für die zur Verfügung stehenden Datenspeicher zu. So verteilt sich zusätzlich die Last bei der Datenhaltung. Die Daten der beiden Datenspeicher müssen allerdings miteinander synchronisiert werden. Dies kann, wie in der Abbildung dargestellt, durch ein Synchronisationsverfahren (vgl. Kapitel 4.3.3) erfolgen. Der Einsatz von Loadbalancern hat den Vorteil, dass die Dienste nicht beachten müssen, dass parallel noch weitere gleiche Dienste vorhanden sind. Ein vorhandener Dienst kann also einfach auf einer zweiten Hardwareinstanz aufgesetzt werden und benötigt keinerlei Anpassungen. Ein Datenspeicher kann nicht einfach mit dem gleichen Datenbestand auf eine andere Hardwareinstanz kopiert werden, wie erwähnt sind hier Synchronisationsverfahren notwendig.

Wird durch die Dienste nur lesend auf den Datenspeicher zugegriffen, dann ist ein Daten-Replikationsverfahren ausreichend. Greifen die Dienste auch schreibend auf den Datenspeicher zu, sind aufwändige Synchronisationsverfahren zwischen den Instanzen des Datenspeichers notwendig. Für die Synchronisation der Daten kann eines der in Kapitel 4.3.3 beschriebenen Verfahren eingesetzt werden.

4.4.3 Dienstintegration in CMS

Die in Kapitel 4.4.1 dargestellten Alternativen der Dienstintegration stellen an ein CMS unterschiedliche Anforderungen, wenn eine dieser Alternativen zur Integration eingesetzt werden soll.

Wird der Dienst eines anderen Systems kopiert, dann muss er neu implementiert werden, wenn er innerhalb des CMS umgesetzt werden soll. Das CMS muss dazu über eine API verfügen, um die für den Dienst notwendigen Funktionalitäten, Benutzerschnittstellen und Synchronisationsprozesse zwischen den Diensten zu implementieren. Wie schon in Kapitel 4.4.1 gezeigt, ist diese Herangehensweise aufwändig. Komplexe Dienste werden in der Praxis meist ausserhalb des CMS implementiert und dann, wie folgend dargestellt, in das CMS eingebunden.

Um einen externen Dienst in ein CMS einzubinden reicht es aus, dass im CMS eine Referenz auf den Dienst angelegt werden kann. Die gesamte Funktionalität, samt der benötigten Benutzerschnittstellen für die Bedienung beziehungsweise Verwendung des Dienstes, wird vom diensterbringenden System zur Verfügung gestellt. Betrachtet man als Anwendung eine Website, die durch ein CMS gepflegt wird und einen Dienst integrieren soll, dann wird der Dienst zum Beispiel über einen Link in einem neuen Fenster dargestellt, oder innerhalb der Website in einen Frame. Greift das diensterbringende System auch noch das Layout und die Farbgebung der Website auf, dann ergibt dies für den Benutzer ein einheitliches Erscheinungsbild. Der Dienst kann so für mehrere Websites eingesetzt werden, jeweils mit einem individuellem Erscheinungsbild, aber gleichem Funktionsumfang. Um den Dienst auf diese Art und Weise zu integrieren muss der Redakteur die URL des Dienstes in das CMS eingeben und auf einer der Webseiten publizieren.

Um einen Dienst individueller zu gestalten und auf der CMS gepflegten Website einzubinden kann eine Dienstschnittstelle verwendet werden, die auf Funktionen des Dienstes zurückgreift.

Die Darstellung wird dabei im CMS implementiert. Dazu benötigt das CMS eine API, um die Daten von der Dienstschnittstelle entgegen zu nehmen und diese darzustellen. Die Dienstschnittstelle kann sowohl im CMS, als auch ausserhalb des CMS implementiert werden. Über die CMS-API wird die Schnittstelle angesteuert, diese übermittelt die Operationen an den Dienst, die API nimmt die Rückgabedaten entgegen, bereitet sie auf und stellt sie dar.

Soll der Dienst ganz individuell gestaltet werden, oder werden die Daten des Dienstes im CMS für eine weitere Verarbeitung benötigt, dann kann eine Datenschnittstelle zum Dienst implementiert werden. Über diese werden die reinen Datensätze, die dem Dienst zugrunde liegen, an das CMS übermittelt. Dies kann wieder über eine CMS eigene API implementiert werden, die die Datensätze in das Repository des CMS speichert, wo sie weiter verarbeitet werden können. Die Daten können dann zum Beispiel dazu verwendet werden, den Dienst im CMS abzubilden.

Wie der vorherige Abschnitt zeigt, ist es nicht notwendig, dass ein CMS über umfangreiche Möglichkeiten verfügen muss, um einen Dienst einzubinden. Eine Referenz auf einen Dienst ist ausreichend, und diese Möglichkeit kann in allen existierenden CMS eingesetzt werden. Wird keine sehr individuelle Gestaltung der Benutzerschnittstelle benötigt, ist dies auch die effizienteste und kostengünstigste Variante und wird überwiegend eingesetzt. Beispiele hierfür sind Wettervorhersagedienste (wetter.com¹) und Auktionsportale (ebay²), die ihren Dienst mehr oder minder individualisiert und personalisiert für die Integration in andere Websites aufbereiten.

Für die Umsetzung des Projekts wurde eine etwas umfangreichere Variante gewählt. Im CMS wird zwar ebenfalls nur eine Referenz auf den Dienst gespeichert, über diese Referenz können aber zusätzlich Daten abgerufen und Funktionen ausgeführt werden, die es erlauben die Darstellung der Benutzerschnittstelle beliebig zu gestalten. Sie wird dabei sowohl aus abgerufenen Darstellungselementen der verwendeten Dienste, als auch aus erzeugten Darstellungselementen über die CMS-API zusammengesetzt. Um dies zu erleichtern wird die CMS-API um Methoden erweitert, die den Abruf der Darstellungselemente von den verwendeten Diensten unterstützen beziehungsweise übernehmen.

Stellen die einzubindenden Dienste also die entsprechenden Schnittstellen zur Verfügung, kann die Anwendung im CMS sowohl beliebig in Funktion und Darstellung, als auch in der Implementation individualisiert werden.

http://www.wetter.com

² http://www.ebay.de

Kapitel 5

Geografisches Informationssystem (GIS)

Viele Informationen, ca. 80%, haben einen Raumbezug und eigenen sich zur geografischen Visualisierung und somit für einem visuellen Informationstransport.

Raumbezogene Informationen werden auf allen Ebenen in Verwaltung, Wirtschaft, Wissenschaft und vom Bürger, benötigt. Sie sind die Grundlage planerischen Handelns und ihre Verfügbarkeit bildet eine maßgebliche Voraussetzung für Standort- und Investitionsentscheidungen (vgl. [Krause u. a. 2003]).

Raumbezogene Daten werden daher von vielen öffentlichen Verwaltungen und Unternehmen digital erhoben und gepflegt. Ein geografisches Informationssystem schafft für Benutzer die Möglichkeit, sich auf Basis solcher digitalen raumbezogenen Daten ortsbezogene Informationen zu beschaffen. So können diese Informationen einem größeren Nutzerkreis zugänglich gemacht werden. Es wird eine Alterung der Daten vermieden, da die raumbezogenen digitalen Daten von Verwaltungen und Unternehmen gepflegt werden. Änderungen können über Datenaustausch nachgeführt werden und so über das GIS aktuell zur Verfügung stehen.

Die Entwicklung geografischer Informationssysteme begann Mitte der fünfziger Jahre, als individuelle Insellösungen und Spezialanwendungen für Großrechner benötigt und entwickelt wurden. Anfang der siebziger bis Mitte der achziger Jahre wurden Entwicklungen überwiegend von Behörden vorangetrieben und erste Geobasisdaten¹ in digitaler Form gespeichert. In den achziger Jahren entwickelten sich die ersten Firmen, die den GIS Markt als neues Geschäftsfeld entdeckten, und die Entwicklung auf die leistungsfähiger werdenden Workstations ausrichteten. In den neunziger Jahren wurden geografische Informationssysteme, durch geschaffene Basisfunktionalitäten, modularer und leichter für spezielle Anforderungen anpassbar. Heutzutage wird der GIS Markt überwiegend durch Angebot und Nachfrage geregelt und ist nicht mehr so sehr von den erfassten Geodaten abhängig, sondern mehr und mehr von

¹ Geobasisdaten sind grundlegende amtliche Geodaten, welche die Landschaft (Topographie), die Grundstücke und die Gebäude anwendungsneutral beschreiben. [Wikipedia 2005]

Anwendungen die benötigt und angeboten werden (vgl. [de Lange 2002] und [Strobl/Griesebner 2003]).

5.1 Umfang und Aufgaben eines GIS

Was hat ein GIS für Aufgaben? Hierzu einige Definitionen, was ein GIS ausmacht, aus der Literatur:

"Ein Geographisches Informationssystem (kurz auch Geo-Informationsystem) ist ein Informationssystem, das der Bereitstellung von Fachinformationen unter Berücksichtigung ihres Raumbezuges dient. Es muss daher Funktionen zur Erfassung, Bearbeitung und Darstellung von raumbezogenen Daten anbieten. Es integriert geometrische graphische thematische Beschreibungen Primitive, und raumbezogenen Objekten. Die Verknüpfung von thematischen Daten mit Informationen zum Lagebezug, die i.d.R. kartographisch dargestellt werden, unterscheidet GIS von reinen Kartier- oder CAD-Systemen." (Page/Häuslein/Greve, 1993).

"Ein Geo-Informationssystem ist ein rechnergestütztes System, das aus Hardware, Software, Daten und den Anwendungen besteht. Mit ihm können raumbezogene Daten digital erfasst und redigiert, gespeichert und reorganisiert, modelliert und analysiert sowie alphanumerisch und graphisch präsentiert werden." (Goodchild und Kemp, 1990).

"Automated systems for the capture, storage, retrieval, analysis and display of spatical data" (Clarke, 1990).

Besonders letztere Definition zeigt kurz und prägnant, was ein GIS leisten soll, beziehungsweise beinhaltet. Es muss möglich sein, geografische Daten aufzuzeichnen oder in ein System einzuspielen, sie zu speichern und zu verwalten Als Informationssystem muss es über Analysewerkzeuge und eine Präsentationsmöglichkeit der Daten und Analyseergebnisse verfügen.

Ein GIS soll es also ermöglichen Geodaten schnell und einfach auf Basis eines Datenbestandes zu analysieren und abzurufen.

Geoinformationen gewinnen heutzutage zunehmend an Bedeutung in den Bereichen Raumplanung, Verkehrslenkung, Umwelt- und Naturschutz, Land- und Forstwirtschaft, Versorgung und Entsorgung, Landesverteidigung und Versicherungswesen. Sie sind wichtig bei öffentlichen Informationen, um BürgerInnen an wichtigen politischen und sozialen Prozessen teilnehmen zu lassen, da Handlungen von Entscheidungsträgern sich auf zahlreiche Mitspracheberechtigte auswirken, und so Geoinformationen einen besonders hohen Stellenwert haben (siehe auch [VuGIS 2003], [Krause u. a. 2003] und [de Lange 2002]).

Weiterhin sind Geoinformationen auch für Unternehmen von zentraler Bedeutung, da räumliche Dimensionen bei ihren Entscheidungen zunehmend mit einbezogen werden. Dies gilt zum Beispiel bei der Entscheidung über einen Unternehmensstandort, ein Angebot an örtliche Charakteristiken anzupassen, den Kundenkreis zu lokalisieren und zu bestimmen oder den Ab-

lauf von Versorgung und Verteilung zu optimieren. Hier werden Geoinformationen als ein strategisches Mittel in der Informationsbeschaffungs- und -verarbeitungsstruktur von Unternehmen eingesetzt (vgl. [Fraunhofer MediaSystems 2004]).

5.2 Einsatzszenarien

Geografische Informationssysteme sind zusammen mit den verschiedensten inhaltlichen Zusammenhängen, in unterschiedlichen Umgebungen und Infrastrukturen einsetzbar. Darunter sind Systeme für zum Beispiel Gebirgskartografie, Forstwirtschaft, Stadtplanung, Flur-, Bau- und Landschaftspläne, demografischische und topologische Informationen und dergleichen sinnvoll.

Für diese Systeme gibt es konkrete Anwendungsszenarien. Beispielsweise möchte der Patient in Erfahrung bringen, wo der nächste Facharzt in seiner Nähe ist. Den Lawinenforscher interessiert, wie groß die Ausdehnung von Lawinengebieten ist und wie sie sich verändert. Der Gewerbetreibende möchte Standortinformationen über Flächen, auf denen er ein neues Werk errichten kann. Der Bürger möchte Informationen einsehen, die ihn über das neue, von der Stadt geplante, Einkaufszentrum informieren. Ein Vermesser benötigt Informationen über ein neu zu erschließendes Gebiet. Es lassen sich viele Beispiele finden, bei denen geografische Informationen zu einem erheblichen Mehrwert führen.

Um die Anforderungen solcher Anwendungen zu erfüllen, müssen geografische Informationssysteme an den verschiedensten Orten, zu unterschiedlichsten Zeiten, in verschiedenen Formen beziehungsweise Darstellungen und über unterschiedliche Wege abrufbar sein. Die Informationen müssen mit speziellen oder gängigen Applikationen, über das Inter- oder Intranet, Funk- oder Kabelverbindungen, mit Navigationssystemen, Handys, Rechnern oder anderen speziellen Geräten jederzeit abrufbar sein.

Man muss bei den Überlegungen zu Einsatzszenarien zur Kenntnis nehmen, dass GIS häufig auch von Firmen und Personen eingesetzt werden, die kein geografischs Fachwissen besitzen. Der überwiegende Teil von zur Verfügung stehender Information hat immer auch einen geografischen Aspekt. Mit einem GIS lässt sich dieser Aspekt visualisieren und so die Information für Firmen und Personen verdichten beziehungsweise präzisieren.

5.3 Erfassung und Pflege von Geodaten und -Informationen

Geografische Daten, seien es nun Rasterdaten (wie zum Beispiel Bilder) oder Vektordaten, müssen von entsprechend ausgebildeten Personal erstellt und gepflegt werden. Dies geschieht im Zuge von Projekten, wie zum Beispiel Bauvorhaben, Forschungen und dergleichen, wie auch durch allgemeine Geschäftstätigkeiten in Behörden, wie zum Beispiel Katasterämtern, Planungsämtern und dergleichen. Zusätzlich produzieren Planungs-, Vermessungbüros und ähnlichen Firmen geografische Daten.

Geografische Daten werden in GIS digital gespeichert und verarbeitet. Analoges Kartenmaterial, wie zum Beispiel Zeichnungen und Pläne, wird digitalisiert und mit einer entsprechenden Anwendung bearbeitet. Neben der Digitalisierung von analogem Kartenmaterial werden Geodaten auch direkt in digitales Geo-Informationssystem eingegeben. Geografisch erfasste Gebiete unterliegen, durch natürliche Einflüsse, wie zum Beispiel Erosion oder durch Eingriffe des Menschen, wie beim Bau neuer Strassen, einem regelmäßigen Wandel, welcher regelmäßig verzeichnet und in den Geodaten aktualisiert werden muss.

Geodaten müssen also durch Digitalisierung oder direkte Eingabe in ein GIS überführt werden und durch regelmäßige Pflege aktuell gehalten werden.

Die Digitalisierung analogen Kartenmaterials wird normalerweise über eine Bildschirmdigitalisierung oder mit Hilfe eines Digitalisiertisches durchgeführt.

Bei der überwiegend verwendeten Methode der Bildschirmdigitalisierung, wird das analoge Kartenmaterial mit Hilfe von Scannern in ein digitales Rasterbild umgewandelt. Die Strukturen des gescannten Kartenmaterials werden mit Hilfe einer Software übernommen. Dabei werden Objekte wie Strassen, Gewässer, Gebäude usw. in unterschiedlichen Ebenen gespeichert. Zusätzlich können den digitalisierten Geo-Objekten auch Sachdaten, wie zum Beispiel Beschreibungen, zugeordnet werden.

Bei der Digitalisierung mit Hilfe eines Digitalisiertisches wird das analoge Kartenmaterial auf dem Tisch fixiert. Mit Hilfe eines Fadenkreuzes werden die Strukturen abgefahren und dabei digital abgespeichert.

Jede der beiden Methoden hat Vor- und Nachteile. Zum Beispiel lassen sich kleine und filigrane Objekte mit der Bildschirmdigitalisierung wesentlich exakter digitalisieren, während man sich beim Digitalisiertisch das Einscannen des analogen Kartenmaterials erspart.

Wichtig ist bei beiden Methoden, dass eine Georeferenzierung vorgenommen wird, bevor mit der Digitalisierung begonnen wird. Bei der Georeferenzierung werden Punkte des Kartenmaterials mit bereits real vermessenen Punkten in einem Referenzsystem zur Deckung gebracht. Dazu werden markante Punkte im Material identifiziert und markiert. Sollte die Vorlage, also die analoge Karte, durch Beanspruchung oder Ähnliches gedehnt oder gestaucht sein, dann bieten die entsprechenden Anwendungen Transformationsmöglichkeiten, um dieses auszugleichen. Ein gewisser Toleranzbereich ist bei jeder Digitalisierung zum realen Zustand vorhanden, der etwa bei einem Maßstab von 1:1000 bei maximal 50cm liegen sollte. Bei zu starken Ungenauigkeiten, etwa durch das Alter der Kartenvorlage, deren Behandlung, mangelhafter Erfassung oder geänderte reale Umstände bleibt nur eine Neuvermessung der kartografischen Situation durch ein Außenteam, welches relativ hohe Kosten verursacht (vgl. auch [GIS Ingenieur] und [de Lange 2002]).

Speicherung

Geo-Informationen bestehen im Wesentlichen aus zwei Klassen von Daten. Einmal die Vektor- oder Rasterdaten, welche die Geometrien oder das Bild für die Darstellung beinhalten und

zweitens bestehen sie aus objektbezogenen Sachdaten. Zusätzlich werden Geodaten teilweise Metadaten zugeordnet, die Eigenschaften, Definition, Herkunft, Gültigkeit, Genauigkeit, Einsatz- und Nutzungsmöglichkeiten usw. dieser Daten beschreiben.

Viele GIS benutzen proprietäre Datenformate oder eigene Datenbanken zur Speicherung der Geodaten. Zusätzlich unterstützen GIS andere Formatet. Rasterdaten werden üblicherweise in Grafikformaten wie TIFF, JPG, BMP oder Ähnlichen gespeichert. Vektordaten werden entweder in den proprietären Formaten gespeichert, wie zum Beispiel Shape Dateien¹, DXF Dateien², in Datenbanken wie zum Beispiel Oracle Spatial³ oder in andere Datenbanken und Dateiformate, die raumbezogene Strukturen unterstützen, exportiert.

Die objektbezogenen Sachdaten können den jeweiligen Raster- beziehungsweise Vektorobjekten zugeordnet werden. GIS speichern diese Sachdaten in unterschiedlichen Datenbanken wie Oracle Spatial, tabellenartigen Dateiformaten und Datenbanken wie in MS Access, CSV- und d-Base-Format oder benutzen dazu ihr eigenes Datenbanksystem.

Es gibt diverse Softwarewerkzeuge, welche ein Geometrieformat in ein anderes umwandeln. Je nach Quellformat oder Datenpflege müssen Sachdaten manuell zugeordnet werden. Teilweise können die Sachdaten nicht in das Zielformat übertragen werden, weil diese vom Zielformat beispielsweise nicht unterstützt werden. Dadurch sind Informationsverluste nicht ausgeschlossen (zu Werkzeugen und Formaten siehe auch [Bauer 2005]).

Verwaltung

Bei der Verwaltung von digitalen Geodaten bedarf es, wie auch bei analogem Kartenmaterial, der Sorgfalt. Zwar können digitale Daten nicht wie analoge Karten mit der Zeit durch Beanspruchung leiden, müssen aber auch entsprechend katalogisiert und dokumentiert werden, um auch zukünftig schnell zugreifbar zu sein und dem Benutzer einen klaren Überblick über ihre Aktualität und ihre Qualität zu bieten.

Durch den enormen Einfluss, den Geodaten heutzutage auf das gesamte Informationsangebot haben, ist es notwendig, dass diese Daten nicht nur einzeln in lokalen Institutionen oder Firmen zur Verfügung stehen, sondern dass sie zugänglich gemacht werden und so mit anderen Geodaten in Verbindung gebracht beziehungsweise kombiniert werden können. Aus diesem Grund herrscht ein reger Handel mit Geodaten vor, der es ermöglicht, den größtmöglichen Informationsgehalt aus verschiedenen Geodatenbeständen zu generieren und zur Verfügung zu stellen.

Analyse

Die Analyse von Geodaten ist für Wirtschaft und Wissenschaft das zentrale Thema und der wichtigste Grund für die Erfassung und Pflege von Geodaten.

¹ http://www.wasy.de/deutsch/produkte/wgeo/hilfe/html/shape_daten.htm

² http://www.hhk.de/produkte/gw/schnitt/gg-dxf.html

³ http://www.oracle.com/technology/products/spatial

Es ist entscheidend zu wissen, welche Abmessungen zum Beispiel Gebäude oder Grundstücke haben, wo welche Leitungen liegen, wie die demografische Entwicklung in bestimmten Bereichen aussieht, wie Flüsse verlaufen und dergleichen. Diese Auswertungen und die daraus folgenden Planungen für Umstrukturierungen, Neubauten, Abriss, Politik, Standortentscheidungen und so weiter lassen sich mit Hilfe von Geodaten unterstützen und eventuell auch sofort in den Geodatenbestand aufnehmen.

Die grafische Darstellung ist die leistungsfähigste Form für die Übertragung und Analyse von Informationen. Daher ist die Visualisierung und Interpretation komplexer Geodaten eine erfolgreiche Methode zum inhaltlichen Verständnis globaler Zusammenhänge (vgl. [Schlotz]).

An ein GIS werden erhebliche Anforderungen gestellt. Die Darstellung der räumlichen Informationen in Form von Karten oder kartenähnlichen Produkten, sei es auf Papier, Bildschirm oder auf Projektionswänden, muss den qualitativen Ansprüchen (Farben, Auflösung, geometrische Präzision, Geschwindigkeit, Symbolik etc.) der Benutzer genügen und dem Einsatzgebiet angemessen und angepasst sein (vgl. [Schlotz] und [de Lange 2002]).

5.4 Beispielsysteme

Aufgrund historischer Entwicklung und voranschreitender Hardwaretechnologie haben sich GIS von Großrechner-, über Workstation- und Desktopapplikationen heute zu verteilten Anwendungen mit verteilter Datenhaltung entwickelt. Diese dezentralen Anwendungen befinden sich derzeit weiter in der Entwicklungs- und Ausbauphase und werden miteinander verknüpft. Weiterhin bilden vorherige Entwicklungsstufen noch die Basis für Datenerfassung, -pflege, -haltung und -analyse. Gerade im Bereich der Datenpräsentation wird derzeit viel Entwicklungsarbeit in die dezentrale Verfügbarkeit investiert.

Viele der heute noch eingesetzten proprietären Systeme werden für die Erfassung, Pflege und teilweise zur Analyse von Geodaten verwendet.

Es gibt eine Vielzahl von Anwendungen, die den Anwender auf unterschiedliche Art und Weise und mit unterschiedlichen Funktionalitäten in der Erstellung, Pflege, Speicherung, Analyse und Präsentation von Geodaten unterstützen.

Zu den wichtigsten und überwiegend verwendeten GIS gehören die Anwendungen der Firmen Autodesk[®], ESRI[®] und Intergraph[®] (siehe dazu [Bauer 2005]).

Intra- / Internet basierende Systeme

Fast alle Anbieter proprietärer GIS bieten auch Web-Schnittstellen oder Web-Implementationen für Intranet- beziehungsweise Internetanwendungen an. Diese Web-Werkzeuge reichen von einfachen Präsentationssystemen bis hin zu interaktiven Bearbeitungswerkzeugen, die es ermöglichen, Operationen auf den Geodaten auszuführen. Weiterhin existieren Anbieter von

5.4 Beispielsysteme 67

GIS, die Web-Schnittstellen oder Web-Implementationen, teils auch als openSource¹ Lösungen, anbieten.

Zur Bereitstellung und Nutzung von Geodaten via Intra-/Internet gibt es verschiedene Lösungsansätze. Dazu gehören GIS-Web-Services wie: Geodatenserver, Web-Map-Services, Online Retrieval Systeme, Online GIS, GIS Function Server und weitere mehr. Die folgende Tabelle gibt eine Übersicht über die verschiedenen Rückgabetypen der GIS-Web-Services (aus [Schuster 2003]).

	räumliche Daten	grafische Daten	Textdaten	Funktionen
Geodatenserver	X			
Web-Map-Services		X		
Online Retrieval Systeme		X	X	
Online GIS	Х	X	Х	
GIS Function Server	Х	Х	Х	X

Tabelle 1: Rückgabetypen der GIS-Web-Services

5.5 Open Geospatial Consortium (OGCTM)

Das Open Geospatial Consortium (OGCTM) ist ein nicht kommerzielles internationales Industriekonsortium bestehend aus über 250 Firmen, öffentlichen Einrichtungen und Universitäten. Es wird unter anderem an Spezifikationen für raumbezogene Dienste und Anwendungen in den verschiedensten Bereichen gearbeitet. Diese Spezifikationen sind öffentlich zugänglich und können von jedem eingesehen, verwendet beziehungsweise implementiert werden. Ziel dieser Spezifikationen ist es, sie zukünftig als Standard zu etablieren.

5.5.1 Ziele des Konsortiums

Durch die historische Entwicklung von Geo-Anwendungen und -Informationssystemen sind viele Insellösungen mit eigenen Datenstrukturen entstanden, die keinem einheitlichen Standard entsprechen. Daher ist es aufwendig, diese verschiedenen Datenstrukturen über einen dezentral zugänglichen Kommunikationskanal, wie zum Beispiel dem Internet, zu verwenden.

¹ Der Ausdruck Open Source steht für quelloffen, einerseits in dem Sinne, dass der Quelltext eines Programms frei erhältlich ist, andererseits für 'offene Quelle', also dass ein Werk frei zur Verfügung steht. Software gilt als Open Source, wenn sie bestimmte Kriterien erfüllt, die in ihrer Open-Source-Lizenz geregelt sind. [Wikipedia 2005]

Das OGC™ wurde ins Leben gerufen, um Standards im Bereich der raumbezogenen Informationsverarbeitung zu schaffen, die eine Interoperabilität von Geo-Diensten und Anwendungen in verteilten Umgebungen ermöglichen.

Die Zukunftsvision des OGCTM ist es, dass Nutzer auf Basis definierter Standards jederzeit über Netzwerke, verschiedene Plattformen und Anwendungen hinweg, Geodaten und -Informationen in verschiedenen Formen austauschen und abrufen können.

Dazu sollen Wünsche und Vorstellungen von Entwicklern, Integratoren und Benutzern in Einklang gebracht und in Spezifikation umgesetzt werden. Diese Spezifikationen sollen die Entwicklung der Interoperabilität von Geo-Produkten und Anwendungen unterstützen. Diese werden dazu als OpenGIS® Spezifikationsstandard der Allgemeinheit zur Verfügung gestellt.

Zur Erreichnung dieser Ziele, besteht das OGCTM aus einer Community¹, ähnlich die der OpenSource Gemeinde. Das OGCTM analysiert den bestehenden Markt und identifiziert Ansatzpunkte, die durch die Definition und den Einsatz von offenen Schnittstellen und Standards die Interoperabilität von Geo-Anwednungen steigern.

Die OGCTM ist einem stetigen Erweiterungs- und Wachstumsprozess unterworfen und erweitert und vertieft dementsprechend ihre Betätigungsfelder im Bereich der Geo-Anwendungen.

5.5.2 Geography Markup Language (GML)

Die Geography Markup Language (GML) ist ein Datenformat auf XML Basis zur Speicherung und Übertragung von geografischen Features. Ein geografisches Feature (kurz Feature) ist eine digitale Repräsentation einer raumbezogenen Entität, also eines Geoobjektes. So genannte einfache Features verfügen über Operationen und kontextabhängige Attribute, die sie näher beschreiben. Dazu gehören immer die raumbezogenen Informationen, um den Ort des repräsentierten Geoobjektes bestimmen zu können. Bei so genannten komplexen Features können zusätzlich Assoziationen zu anderen Features bestehen.

Ein Geoobjekt definiert [de Lange 2002] wie folgt:

Ein raumbezogenes Objekt oder Geoobjekt , kurz Objekt, - häufig synonym mit Feature verwendet - beschreibt eine physisch, geometrisch oder begrifflich begrenzte Einheit der Natur und besitzt eine individuelle Identität

In GML lassen sich Features inklusive ihrer Geometrie und ihrer Attribute darstellen, und eindeutig anhand ihrer Identität erkennen und zuordnen. Zusätzlich können Assoziationen zwischen Features mittels GML abgebildet werden. Somit können in GML geografische Zusammenhänge gespeichert und übertragen werden.

¹ Eine Community (engl. community, Gemeinschaft) ist eine Gruppe von Personen, die gemeinsames Wissen entwickeln, Erfahrungen teilen und dabei eine eigene Identität aufbauen. Communities profitieren von dem Grundsatz, dass alle Teilnehmer zum Erfolg beitragen, indem sie ihr Wissen einbringen. [Wikipedia 2005]

Unter Verwendung von GML wurde von der OGCTM die Styled Layer Descriptor-Spezifikation (SLD) definiert. Diese Spezifikation definiert das Format einer Stilsprache für die Erzeugung von georeferenzierten Karten. Durch diese Stilsprache können georeferenzierte Karten in einem benutzerdefinierten Darstellungsstil generiert und visualisiert werden. SLDs kontrollieren die grafischen Ausgabe von Geodaten – Karten lassen sich also individueller gestalten

Weitere Informationen zu GML und davon abgeleiteten Formaten finden sich in [Bauer 2005] und unter [OGC 2005].

GML ist eines der Basisbestandteile der folgend dargestellten OGCTM Web-Dienste und findet, gemeinsam mit SLD, auch Verwendung im später dargestellten Integrationssystem um geografische Daten über Webportale zugänglich zu machen.

5.5.3 OGCTM konforme Web-Dienste

Die OGCTM hat diverse Spezifikationen erarbeitet, die eine Reihe von unterschiedlichen Funktionalitäten auf geografischen Daten, deren Schnittstellen und Datenformaten festlegen. So wurden Sprachstandards für die Beschreibung von geografischen Daten festgelegt, wie die Geography Markup Language (GML). GML bildet die Basis für weitere durch die OGCTM beschriebene Dienste und andere Spezifikationen. Es existieren einige Produkte und Projekte, die die von der OGCTM definierten Dienste implementieren. Die geforderten Eigenschaften der Dienste werden darin umgesetzt und je nach Umfeld, oder Auftrag die benötigten optionalen Eigenschaften, die in den Spezifikationen festgelegt sind. Werden zusätzliche Dienste oder Eigenschaften in Diensten benötigt, soll dies der OGCTM mitgeteilt werden, die dies dann in die Spezifikationen integriert. Zwei Projekte, die die von der OGCTM definierten Dienste teilweise als Referenz implementieren, sind das Framework Deegree [deegree 2005] und der UMN Mapserver [UMN Mapserver 2005].

Für diese Arbeit sind vor allem die Spezifikationen einiger Web Dienste von Bedeutung. Die wichtigsten werden im Folgenden kurz dargestellt. Die gesamten Spezifikationen finden sich auf [OGC 2005].

Alle beschriebenen Dienste stellen World Wide Web Services dar, die über das HTTP-Protokoll mit Uniform Resource Locators (URL's) angesprochen werden können und somit mit Standard-Webbrowsern aufrufbar sind. Für die Dienste sind XML- beziehungsweise GML-Schemata für die Ein- und Ausgabe definiert.

5.5.3.1 Web-Map-Service

Der Web-Map-Service (WMS) spezifiziert den Zugriff auf die grafische Darstellung von Geodaten. Er liefert Rasterbilder aus Geodaten, wie zum Beispiel Stadtpläne, Bebauungspläne, etc.

Diese Rasterbilder können für externe Anwendungen in verschiedenen grafischen Ausgabeformaten zur Verfügung gestellt werden. Zusätzlich soll der WMS die Zusatzinformationen der Features (siehe Kapitel 5.5.2) ausliefern.

In der Spezifikation des WMS sind auch die Eigenschaften für einen kaskadierenden WMS festgelegt. Ein kaskadierender WMS kann zusätzlich als ein Klient für andere WMS agieren. Er aggregiert die Services anderer WMS und stellt sie als einen Service zur Verfügung. Beispielsweise legt er Karten unterschiedlicher WMS übereinander und liefert sie als Rasterbild ausliefern (siehe auch [Beaujardiere 2004]).

Der WMS verfügt unter anderem über drei Anfragen: GetCapabilities, GetMap und GetFeatureInfo. Alle Anfragen besitzen spezifizierte Parameter, die Art und Inhalt des Rückgabergebnisses beeinflussen.

Folgend eine kurze Beschreibung der Anfragen:

GetCapabilities

Diese Anfrage liefert Informationen über die Möglichkeiten des WMS zurück. Dazu gehören die Information über den Service selber, verfügbaren Informationen (Karten) und Anfrageparameter, die akzeptiert werden. Die Antwort auf die Anfrage ist ein definiertes XML Dokument, welches die entsprechenden Informationen über den Service beinhaltet und vom anfragenden Klienten ausgewertet werden kann.

Jede über den WMS abrufbare Karte muss über einen WMS-weit eindeutigen Namen verfügen, um sie eindeutig von den anfragenden Klienten referenzieren zu können.

GetMap

Auf diese Anfrage hin wird eine Kartendarstellung zurückgeliefert, die aus einer Rastergrafik besteht. Der WMS kann hierbei mehrere Karten übereinander legen, die er zur Verfügung hat. Dabei sind die Karten, die übereinander liegen, transparent, so dass die Informationen aller Karten gleichzeitig sichtbar sind. Ein Element innerhalb einer Karte, welches
über einem anderen liegt, verdeckt in seinen Ausmaßen alle darunter liegenden Elemente.
Karten zu überlagern ist nur dann sinnvoll, wenn die einzelnen übereinander gelegten
Karten Informationen über den gleichen geografischen Raum liefern und sich die so dargestellten Informationen aus mehreren Karten sinnvoll ergänzen.

Um eine Kartendarstellung zu generieren, benötigt der WMS mehrere Anfrageparameter. Dazu gehört die Liste der zu überlagernden Karten, die zu verwendenden Styles¹ für jede Karte, das geografische Referenzsystem², den darzustellenden Ausschnitt, die Größe der Darstellung und das Ausgabeformat.

¹ Art, wie eine Karte dargestellt wird. Zum Beispiel die Farbgebung von Strassen oder Schraffur von Flächen.

² siehe dazu [Bezugssysteme]

GetFeatureInfo

Es werden Informationen über einzelne Elemente (fachlich Features genannt; Linien, Polygone, usw. - siehe Kapitel 5.5.2) innerhalb von Karten zurückgeliefert. Einzelne Elemente von Karten können bei der Erstellung oder durch Bearbeitung mit zusätzlichen Daten angereichert worden sein. Sie können zusätzlich definierende Bezeichnungen, wie zum Beispiel Leitungs- oder Flurstückkennungen beinhalten, aber auch zusätzliche Daten, wie kurze Beschreibungen oder Hinweise.

Diese Daten sind durch den WMS abrufbar und werden, aufbereitet in einer GML- oder XML-Darstellung, zurückgeliefert. Die zurückgelieferten GML- oder XML-Daten können durch den anfragenden Klienten ausgewertet und weiterverwendet werden.

Als Anfrageparameter werden die gleichen benötigt, wie bei der "getMap" Anfrage und zusätzlich der X- und Y-Koordinatenwert des Punktes, zu dem die Elementinformationen geliefert werden sollen. Der gewählte Punkt muss sich innerhalb des durch die Anfrage festgelegten Ausschnittes befinden.

Für eine detailliertere Betrachtung siehe [Bauer 2005] und [OGC 2005].

5.5.3.2 Web-Feature-Service

Der Web-Feature-Service (WFS) erlaubt es über das HTTP Protokoll geografischen Daten als GML abzurufen und Datenmanipulationen an ihnen vorzunehmen. Der Service erlaubt folgende Manipulationen:

- erzeugen neuer Features,
- entfernen von Features,
- ändern von Features und
- Abruf oder Abfrage von Features, basierend auf räumlichen und nicht-räumlichen Bedingungen (zum Beispiel Punkten auf den Karten, oder Eigenschaften der Features).

Die Anfrage an den WFS besteht dabei aus einem Abruf von Features oder aus Datenmanipulationen, die auf ein oder mehrere Features angewendet werden.

Der WFS liefert dem anfragenden Klienten geografische Daten und Informationen im GML-Format.

Wie auch der WMS verfügt der WFS über verschiedene Anfragen. Dazu gehören die folgenden:

GetCapabilities

Der WFS muss sich selber und seine Fähigkeiten beschreiben können. In dem zurückgelieferten XML-Dokument gibt er an, welche Feature-Typen er unterstützt, und welche Operationen auf jedem Typ möglich sind.

DescribeFeatureType

Der WFS liefert Informationen über die Struktur aller Feature-Typen, die er unterstützt.

GetFeature

Zu festgelegten Features liefert der WFS Informationen, die auf dem Klienten ausgewertet werden können.

Transaction

Mit diesem Modus können Operationen auf geografischen Features durchgeführt werden. Dazu gehören: erstellen, ändern und löschen von Features.

LockFeature

Für den Verlauf einer Transaktion ist es möglich, ein oder mehrere Features für den exklusiven Zugriff zu sperren. Dann ist es möglich exklusiv Operationen auf dem gesperrten Feature auszuführen.

Die aufgeführten Anfragen besitzen weitere Parameter, welche die Anfrage jeweils näher spezifizieren und dem WFS eine Auswertung erlauben. Nähere Angaben finden sich dazu in [Bauer 2005] und [OGC 2005].

Kapitel 6

Integration von CMS- und GIS-Diensten

CMS und GIS dienen beide dem Zweck, Inhalte so aufzubereiten und darzustellen, dass sie für die Übermittlung von Information geeignet sind. Beide Systeme unterscheiden sich dabei in der Grundlage der übermittelten Information (den Inhalten) und ihrer Darstellung.

Bei einem CMS werden als Grundlage normalerweise textuelle oder multimediale Inhalte verwendet, die von einem oder mehreren Redakteuren gepflegt werden können. Diese Inhalte können über verschiedene Kanäle und in unterschiedlichen Darstellungsformen ausgegeben werden um die gewünschte Information zu transportieren. Ein CMS hat normalerweise nicht die Möglichkeit geografische Daten zu visualisieren, ausser über die Einbindung von statischen Rasterbildern oder Spezialimplementationen. Mit Rasterbildern kann der Betrachter nicht interagieren, um sich Detailinformationen zu verdeutlichen. Die Spezialimplementationen sind aufwendig und überwiegend nicht einfach auf andere CMS portierbar. Weitere Quellen mit geografischen Daten sind nur mit Aufwand anzubinden. Da aber ein Großteil von Inhalten einen geografischen Bezug haben, lassen sich durch eine kartografische beziehungsweise grafische Darstellung wesentlich einfacher die dahinterliegenden Informationen transportieren.

Im Gegensatz zu Rasterbildern verfügt ein GIS über geografische Daten, die überwiegend spezielle Darstellungen und Sachverhalte beinhalten. Visualisierte geografische Daten lassen sich fast beliebig detailliert betrachten. Durch Visualisierung der Geodaten und Darstellung von strukturierten textuellen Inhalten kann die beabsichtigte Information einem wesentlich größeren Kreis von Betrachtern verständlich gemacht werden.

Die Inhalte von CMS und GIS sind grundlegend verschieden, können aber in Kombination zu einem transparenteren Informationsfluss führen.

GIS benutzen als Grundlage kartografische, beziehungsweise geografische Daten, die oftmals mit tabellarischen, textuellen Datensätzem angereichert sind, die spezielle Elemente oder Bereiche der geografischen Daten näher beschreiben. Die größtmögliche Information bieten geografische Daten für den Menschen, wenn sie visualisiert werden, also aus ihnen Karten gene-

riert werden. Diese Karten erleichtern es Informationen aus den Daten zu ziehen. Es ist ohne eine Spezialanwendung in einem GIS nicht möglich diese Karten zusätzlich mit textuellen Inhalten anzureichern, die einen Informationsmehrwert für den Betrachter bieten. Dieser Informationsmehrwert sollte sich dazu noch frei verwalten lassen und auf die angestrebte Zielgruppe und deren Wissensstand ausrichten lassen. Ein Weg, diesen Informationsmehrwert zu generieren, ist es, GIS Funktionalitäten und CMS Inhalte miteinander zu verknüpfen, sie zu integrieren.

Die Anforderungen an die Integration müssen von mehreren Seiten aus betrachtet werden. Einerseits sowohl aus Richtung des CMS, sowie aus der des GIS und aus der eines Drittsystems. GIS Inhalte müssen sowohl in einem CMS, als auch CMS Inhalte in einem GIS verfügbar gemacht werden können. Für ein Drittsystem sollte es zusätzlich möglich sein, sowohl Inhalte aus dem CMS, als auch Inhalte aus dem GIS verwenden und verarbeiten zu können. Um dies zu gewährleisten, müssen Inhalte aus dem CMS und Inhalte aus dem GIS miteinander in Beziehung gebracht und eindeutig referenziert werden können, wenn sie semantisch miteinander in Beziehung stehen.

Die einzelnen Systeme sollen dabei eigenständig bleiben und über offene Schnittstellen miteinander kommunizieren. So kann es weiteren Systemen erlaubt werden, diese Schnittstellen zu verwenden und an den vorhandenen Daten zu partizipieren.

6.1 Lösungsansatz

Ein Integrations-System zur Zusammenführung und Integration von CMS und GIS muss abhängig von den geplanten Eigenschaften auf unterschiedliche Weise angegangen werden: Wie ist der primäre Einsatzzweck des Integrations-Systems? Welche Personen sind an der Pflege beteiligt? Wer nutzt das spätere Angebot? Welche Anforderungen bestehen an die Nutzung?

Das im Zusammenhang mit den Webpräsenzen der Gemeinden der Metropolregion Hamburg angestrebte Integrations-System soll es ermöglichen geografische Informationen darzustellen, die auf in der Metropolregion erzeugten Geodaten beruhen. Diese grafisch zu visualisierenden Geodaten sollen zusätzlich mit redaktionellen Inhalten angereichert werden, die in einem CMS vorliegen oder dort erzeugt werden. Dabei soll es den Redakteuren der Websites ermöglicht werden, die grafisch visualisierten Geodaten selber auszuwählen und im vorgegebenen Rahmen selber zusammenzustellen und zu publizieren. Dies soll der Redakteur leisten können, auch wenn er keinerlei Erfahrung in der Erstellung, Pflege oder Verwaltung von Geodaten besitzt. Durch eine einfache Schnittstelle soll er befähigt werden, diese Geodaten im Internet oder in Intranets zu publizieren und dabei das CMS weiterhin als redaktionelles Werkzeug zu verwenden. Neben der reinen Publikation der Geodaten über das CMS soll er die Möglichkeit haben, bestimmte Bereiche in den visualisierten Geodaten hervorzuheben und sie mit redaktionellem Inhalt aus dem CMS zu verknüpfen.

6.1 Lösungsansatz 75

Die so aufbereiteten und angereicherten Geodaten sollen von anderen Nutzern einsehbar und abrufbar sein. Diesen Nutzern soll es zusätzlich ermöglicht werden, Annotationen¹ zu bestimmten Bereichen in der grafischen Visualisierung der Geodaten zu erstellen. Dazu müssen sie den zu annotierenden Bereich bestimmen, und diesem ihre Annotation zuordnen können. Auch hier sollen keine Kenntnisse von geografischen Systemen vorausgesetzt werden.

Der Ansatz soll so gestaltet sein, dass sowohl Redakteure, als auch Nutzer mit Browsern ohne zusätzliches Plugin² arbeiten und die notwendigen Operationen durchführen können. So wird sichergestellt, dass die Erreichbarkeit der Websites nicht durch Notwendigkeiten auf der Clientseite eingeschränkt werden. Dies ist auch gerade vor dem Hintergrund notwendig, dass in den Behörden der Metropolregion Hamburg technische Einschränkungen vorherrschen. Gerade für Mitarbeiter dieser Behörden ist der Zugang zu geografischen Informationen sinnvoll (vgl. [Schlotz]).

Das Integrations-System soll die oben genannten Anforderungen erfüllen, aber auch flexibel und offen genug sein, um für abweichende und erweiterte Anforderungen potentieller weiterer Anwendungsgebiete eingesetzt, beziehungsweise erweitert werden zu können.

6.2 Systemanforderungen

Das Integrations-System soll in unterschiedlichen Systemlandschaften eingesetzt werden. Es sollen plattformunabhängig unterschiedliche Systemarchitekturen verwendet und unterschiedliche Klienten-Systeme mit Inhalten bedient werden können. Die am System beteiligten Komponenten, wie Datenbank, Webserver, WMS, WFS, CMS, GIS Server und weiteres müssen skalierbar sein, beziehungsweise über unterschiedliche Hardwaresysteme verteilt werden können. Dies ist aufgrund der hohen Anforderungen der GIS Anwendungen an die Hardware notwendig. Für WMS und WFS wird neben hoher Rechengeschwindigkeit viel Arbeitsspeicher benötigt. Die weiteren Komponenten benötigen vergleichsweise weniger Ressourcen.

Die Kommunikation zwischen den Komponenten soll dabei an die Bedürfnisse angepasst werden können. Das Integrations-System soll eine offene Architektur besitzen, um so an alle inhaltlichen Anforderungen anpassbar zu sein und beliebige weitere Systeme integrieren zu können. Zusätzlich sollen Schnittstellen zu Drittsystemen implementierbar sein, die es ermöglichen, dort die systemeigenen Daten zusammen mit denen des Integrations-Systems zu verwenden.

Wichtig ist, dass das Integrations-System nicht speziell an eine soft- oder hardwarewaretechnische Infrastruktur angepasst wird. Es soll die Möglichkeit bestehen, neben beliebigen

¹ Annotation bedeutet "Anmerkung"", "Beifügung", "Hinzufügung". In diesem Sinn haben Annotationen bei Stichworten, Begriffsklärungen oder ausführlichen Texten den Charakter der Erklärung beziehungsweise Ergänzung. [Wikipedia 2005]

² Plugin (von engl. to plug in - einstöpseln, anschließen) oder Ergänzungs- oder Zusatzmodul ist eine gängige Bezeichnung für ein Softwareprogramm, das in ein anderes Softwareprodukt "eingeklinkt" wird. [Wikipedia 2005]

CMS auch beliebige GIS, GEO-Formate und Datenbanksysteme zu verwenden und anzubinden.

Das Integrations-System soll keine Replikation der Daten implementieren, also weder CMS Daten, noch GIS Daten selber verwalten. Es soll lediglich die Verknüpfungen zwischen den CMS und GIS zur Verfügung stellen und zugreifbar machen. Dadurch wird eine redundante Datenhaltung vermieden und die entsprechenden Inhalte können unabhängig in den speziellen Anwendungen verwaltet und gepflegt werden (siehe dazu auch Kapitel 4.3 und 4.4). Trotz der teilweise regelmäßigen Pflege und Erweiterung der Geodaten, gerade im Bereich von Bebauungsplänen usw., und der ständigen redaktionellen Bearbeitung von CMS Inhalten, ist es somit nicht notwendig, laufend Abgleichverfahren zwischen den Datenbeständen ablaufen zu lassen.

Aufgrund des Umfeldes für das Pilotprojekt mussten GEO-Systeme der unterschiedlichsten Hersteller angebunden werden können. Dazu gehören Intergraph®, Geomedia®, ESRI® ArcGIS®, Autodesk MapGuide® und Oracle Spatial®. Als CMS wurde für das Pilotprojekt CoreMedia® Smart Content Technology eingesetzt. Dieses System wurde bereits für mehrere Projekte am Arbeitsbereich Softwaresysteme der TU-Harburg ([STS 2005]) eingesetzt und eine entsprechende Erfahrung ist demnach vorhanden.

6.2.1 Standards und Kompatibilität

Generell soll bei der Konzeption und Umsetzung des Integrations-Systems darauf geachtet werden, dass definierte Standards verwendet werden, die sicherstellen, dass das System skalierbar, zukünftig verwendbar und erweiterbar bleibt.

Zu diesen Standards gehören die von der OGC™ festgelegten Formate und Webservices, die es ermöglichen, GIS unterschiedlicher Hersteller anzubinden und deren Daten als Karten zu visualisieren. So wird es zukünftig möglich sein, alle Systeme von GIS-Anbietern zu verwenden, die diese OGC™ Standards unterstützen. Weiterhin sollen standardisierte Datenbankverbindungen, wie JDBC, Verwendung finden. Es besteht so die Möglichkeit, die eingesetzten Datenbanksysteme gegen andere auszutauschen.

Die Kommunikation zwischen den einzelnen Komponenten soll so weit wie möglich über das HTTP-Protokoll mit XML-Datenstrukturen als Datenaustauschformat abgewickelt werden. Außerdem können durch diese Standards und Webservices eigene Applikationen auf die Informationen, die das Integrations-System zur Verfügung stellt, zugreifen.

Durch den breiten Einsatz von standardisierten Werkzeugen und Protokollen ist auch ein breites Entwicklerpotential vorhanden, welches eine leichtere Verwaltung, Anwendung und Erweiterung des Integrations-Systems ermöglicht.

Um zu gewährleisten, dass das Integrations-System weitgehend plattformunabhängig eingesetzt werden kann, soll das System Java-basierend implementiert werden. Dadurch ist es

möglich, das Integrations-System in fast allen hard- und softwaretechnischen Systemumgebungen zu implementieren und zu betreiben.

Durch die Möglichkeiten des dezentralen Betriebes der einzelnen Komponenten sind lokale Anpassungen vornehmbar und einzelne Komponenten können durch andere gleicher Funktionalität ersetzt werden. Die einzelnen Komponenten werden intern ebenfalls modular aufgebaut, sodass intern Schnittstellen definiert sind, die die Kompatibilität zwischen verschiedenen speziellen Implementierungen oder Produkten und so eine kontinuierliche Weiterentwicklung sicherstellen.

6.2.2 Systemarchitektur

Die Gesamt-Systemarchitektur kann abstrahiert von einer konkreten Problemstellung entwickelt werden. Das Integrations-System integriert sich als eigenständiger Baustein in das Gesamtsystem.

Durch die Forderung nach Systemunabhängigkeit, sowohl auf der Seite des CMS, als auch auf der Seite des GIS, ist es nicht möglich das Integrations-System in eine dieser beiden Komponenten einzubetten. Es muss mit diesen Komponenten kommunizieren und sie miteinander verbinden können. Zusätzlich soll es auch Drittsysteme mit Daten und Diensten versorgen können.

Das zu entwickelnde Integrations-System wird zur einfacheren Unterscheidung im Folgenden GIS-Integrator¹ genannt werden.

Aus den Forderungen und den einzusetzenden Komponenten resultierte die in Abbildung 18 dargestellte grobe Systemarchitektur der beteiligten Systeme und des GIS-Integrators.

Der GIS-Integrator bildet die Schnittstelle zwischen einem CMS, dem GIS und eventuellen Drittsystemen. Alle beteiligten Systeme können unabhängig voneinander existieren, sollen aber an den Informationen und Inhalten der anderen Systeme partizipieren können.

Die Redakteure des CMS sollen wie gewohnt ihre Inhalte über die vom CMS angebotenen Eingabeschnittstellen pflegen können. Zusätzlich erhalten sie die Möglichkeit über den GIS-Integrator die Karten festzulegen, die später zusammen mit den CMS Inhalten angezeigt werden sollen. Zusätzlich können CMS Inhalte positionsbezogen den Karten zugeordnet werden.

Dem Benutzer, der die gemeinsamen Inhalte von CMS und GIS dargestellt bekommt, erscheint dies transparent, als würde er die Informationen von einem einzigen System erhalten.

Der GIS-Integrator verfügt über definierte Schnittstellen zum GIS und zum CMS, mit dem dafür jeweils notwendigen Funktionsumfang. Sowohl das GIS, als auch das CMS müssen dazu ebenfalls über geeignete Schnittstellen verfügen, die vom GIS-Integrator angesprochen werden können

¹ Name des prototypischen Systems, welches im Laufe des Projekts implementiert wurde.

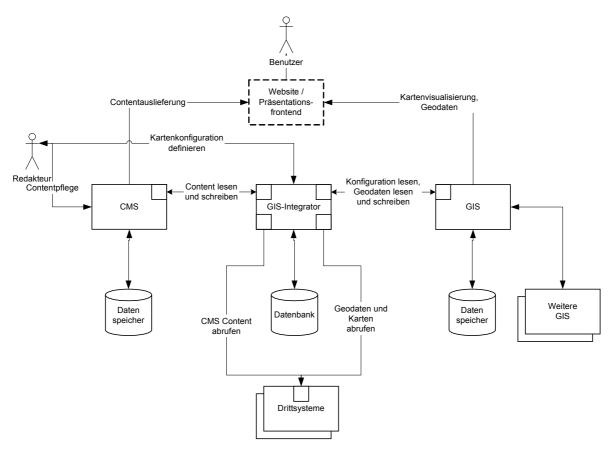


Abbildung 18: Gesamtarchitektur des GIS-Integrationssystems

Die Schnittstelle zum GIS ist im Regelfall durch die OGCTM festgelegten Standards definiert und sollte demnach nicht angepasst werden müssen. Über diese Schnittstelle werden zum Beispiel die generierten Karten abgerufen.

Die Schnittstelle zum CMS muss im Regelfall angepasst werden, um den Zugriff durch den GIS-Integrator zu ermöglichen. Durch Austausch der Schnittstelle zum CMS kann so jedes beliebige CMS Verwendung finden. Zu Karten annotierte Inhalte können so zum Beispiel im CMS gespeichert werden und stehen dort ebenfalls zur Verfügung. Um sicher zu stellen, dass der GIS-Integrator die notwendigen Operationen über die Schnittstelle im CMS durchführen kann, muss die Schnittstelle einen bestimmten Funktionsumfang aufweisen. Erfüllt die Schnittstelle des CMS die Anforderungen nicht, oder ist es nicht möglich die Schnittstelle des CMS entsprechend anzupassen, dann kann eine zusätzliche externe Schnittstelle zwischen CMS und GIS-Integrator aufgesetzt werden, die sich der Funktionalitäten der CMS API bedient oder direkt in das Repository des CMS schreibt.

Neben dem Zugriff des GIS-Integrators auf Funktionen des CMS, ist es auch für das CMS und andere Drittsysteme möglich, Funktionen des GIS-Integrators zu nutzen. Es können zum Beispiel die Inhalte des CMS und die, über den GIS-Integrator dazu verknüpften, GIS-Daten oder -Karten angezeigt werden. Die notwendigen Daten werden dazu über die Schnittstelle des GIS-Integrators abgerufen und, zum Beispiel mit Hilfe eines Webseiten-Generators des CMS, als HTML gemeinsam dargestellt.

Auf diese Art kann prinzipiell jedes CMS und auch jedes GIS, sei es OGCTM konform oder nicht, mit dem GIS-Integrator interagieren. Zusätzlich haben auch externe Eigenentwicklungen, über den Zugriff auf den GIS-Integrator, die Möglichkeit Daten aus CMS und GIS gemeinsam abzurufen und darzustellen.

Die Kommunikation der beteiligten Systeme untereinander soll auf Standards basieren. Dazu kommen webbasierende Schnittstellen zum Einsatz, die auch von der OGCTM favorisiert werden. Sie basieren auf dem HTTP Protokoll und verwenden zum Datentransport überwiegend XML oder eine davon abgeleitete Variante (beispielsweise GML). Durch die Wahl der webbasierenden Schnittstellen ist der GIS-Integrator standortunabhängig und kann über das Internet verwendet werden. Außerdem bieten bereits einige CMS und proprietäre GIS Systeme entsprechende webbasierende Schnittstellen an, die angebunden werden können.

Abbildung 19 stellt die zu verwendenden Protokolle und Datenaustauschformate zwischen den einzelnen Systemen und Beteiligten in der Gesamtarchitektur des GIS-Integrationssystems dar.

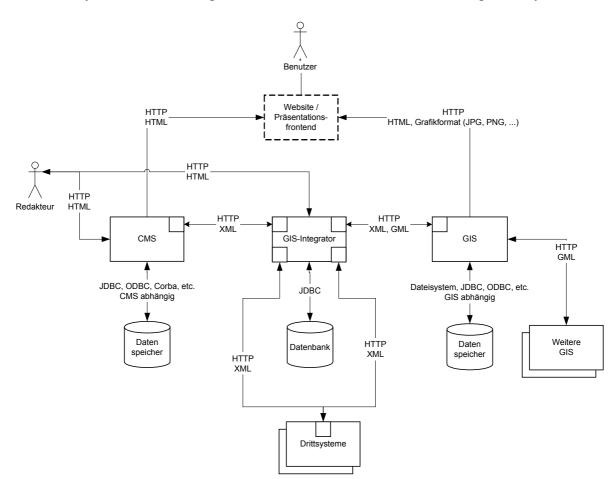


Abbildung 19: Austauschprotokolle im Gesamtsystem des GIS-Integrationssystems

Wie in Abbildung 19 zu erkennen ist, erfolgt der Datenaustausch zwischen den Systemen untereinander über das HTTP Protokoll. Als Datenformat ist fast überall XML, oder eine XML ähnliche Struktur (zum Beispiel GML) gewählt, um größtmögliche Kompatibilität und Stan-

dardisierung zu erreichen. Die Übermittlung der zusammengeführten Daten aus CMS und GIS ist hier über das HTTP Protokoll und in HTML beziehungsweise in einem Grafikformat an das Präsentationsfrontend vorgesehen. Im Benutzerumfeld einer normalen Workstation und des Internets wäre dies die Standardlösung. Es ist aber auch denkbar, die Informationen über ein anderes Protokoll (WAP, etc.), und / oder andere Datenformate (XHTML, XML, etc.) an andere Endgeräte, wie zum Beispiel Mobile Devices, zu übermitteln.

6.2.3 Use Cases

Um den notwendigen Funktionsumfang des GIS-Integrators abzuleiten, müssen die Use Cases betrachtet werden. Es gibt beim GIS-Integrator generell zwei Akteure, die unterschiedliche Ansprüche haben. Die Akteure sind zum einen die Benutzer, die geografische Informationen abrufen wollen, und zum anderen die Redakteure, die diese Informationen zur Verfügung stellen, redaktionell verwalten und erweitern wollen. Aus den Use Cases der Benutzer leiten sich einige Notwendigkeiten in den Use Cases für die Redakteure ab. Zuerst werden deshalb die Use Cases für die Benutzer betrachtet.

Wenn man das Gesamtsystem aus der Benutzersicht betrachtet, dann möchte der Benutzer generell zwei Dinge. Einerseits möchte er mit der Karte interagieren können, und andererseits zusätzliche Informationen dazu erhalten. Daraus ergibt sich das in Abbildung 20 dargestellte Use Case Diagramm:

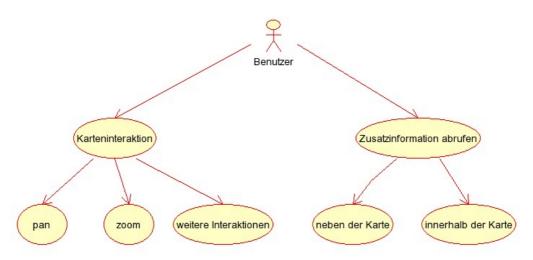


Abbildung 20: Use Case Diagramm Benutzer – Karteninteraktion

Wie die Abbildung 20 zeigt, möchte der Benutzer über verschiedene Interaktionsmöglichkeiten verfügen. Dazu gehören zum Beispiel das Verschieben des Kartenausschnittes, hineinoder herauszoomen der Karte, einblenden bestimmter Elemente und ähnliches. Außerdem möchte er zusätzliche Informationen abrufen können. Diese Informationen sollten dabei textuell oder auch multimedial sein können. Diese Zusatzinformationen sollten sowohl neben der Karte, als auch innerhalb der Karte zur Verfügung stehen, um zum Beispiel allgemeine In-

formationen über die Karte an sich, als auch Informationen über bestimmte Objekte auf der Karte zu bekommen.

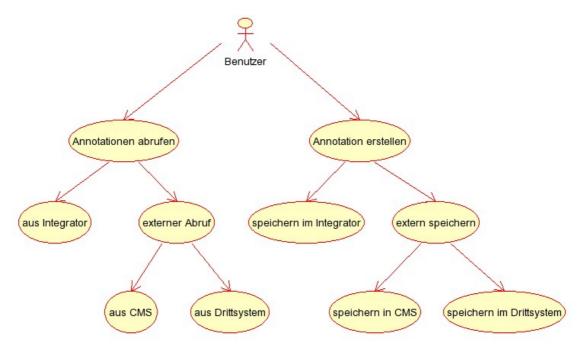


Abbildung 21: Use Case Diagramm Benutzer - Annotationen

Dem Benutzer soll neben dem Abrufen von Information auch die Möglichkeit gegeben werden, selbst Informationen zu hinterlegen. Diese Informationen soll er mit Bereichen auf den Karten oder Objekten innerhalb der Karten in Verbindung bringen können. Er soll also Annotationen mit geografischen Bezug erstellen können. Um solche Annotationen erstellen zu können, muss der Benutzer Objekte oder Bereiche der Karte auswählen und entsprechend mit einer Annotation versehen können.

Das Diagramm in Abbildung 21 zeigt, dass es dem Benutzer transparent erscheint, woher die Annotationen stammen, beziehungsweise in welchem System die von ihm erstellten Annotationen gespeichert werden. Generell soll es möglich sein, Annotationen innerhalb des GIS-Integrators, aber auch in externen Systemen wie einem CMS speichern zu können.

Redakteure sollen die Möglichkeit erhalten die Karten selbst zusammenzustellen, um eine initiale Ansicht der Karte festlegen zu können. Sie müssen festlegen können, welche Kartendaten dargestellt werden sollen (Layer), welcher Kartenausschnitt dargestellt wird und wo die Karte in das CMS eingebunden werden soll. Diese redaktionellen Aufgaben sollen sie auch ohne Kenntnisse von GIS erledigen können. Dazu können sie so genannte WebMaps definieren, die festlegen, welche Layer die einzelnen Karten enthalten und welcher Kartenausschnitt eingestellt ist. Diese WebMaps können über den GIS-Integrator verwaltet und von dort an das CMS übergeben werden. Es ergibt sich für den Redakteur das in Abbildung 22 gezeigte Use Case Diagramm.

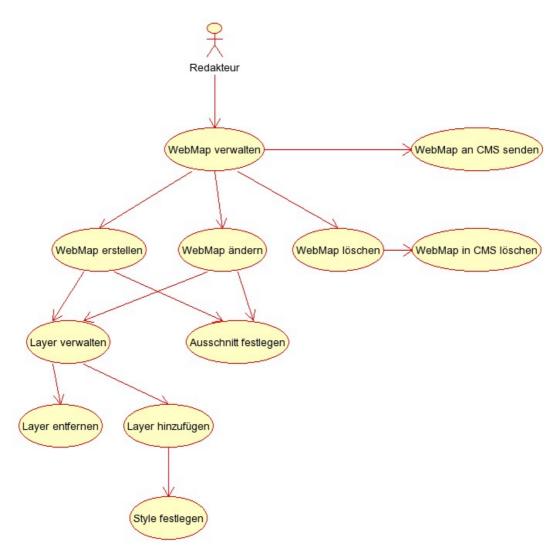


Abbildung 22: Use Case Diagramm Redakteur - WebMaps

Wie Abbildung 22 zeigt, kann der Redakteur festlegen, welche WebMaps über den GIS-Integrator verfügbar sein sollen, und welche er an das CMS übergeben möchte. Zu diesen WebMaps kann der Redakteur jetzt zusätzlich weiteren redaktionellen Inhalt verwalten. Diese Annotationen können Bereichen der WebMaps zugeordnet werden.

Dem Redakteur muss es möglich sein, eigene Annotationen zu erstellen und verwalten zu können. Außerdem muss er die Annotationen der Benutzer verwalten können, um nicht gewünschte Inhalte zu entfernen und gewünschte Inhalte freizuschalten. Das Use Case Diagramm in Abbildung 23 veranschaulicht die Aufgaben, die der Redakteur erledigen können muss.

Der Redakteur verfügt über die drei Grundoperationen: Annotationen erzeugen, ändern und löschen. Bestimmte Bereiche der Karten, wie Punkte, geometrische Figuren oder Objekte der WebMaps werden als Features (siehe auch Kapitel 5.5.2) bezeichnet. Die Zuordnung einer Annotation zu einem Feature wird als Mapping¹ bezeichnet.

¹ Mapping: (engl.) die Abbildung

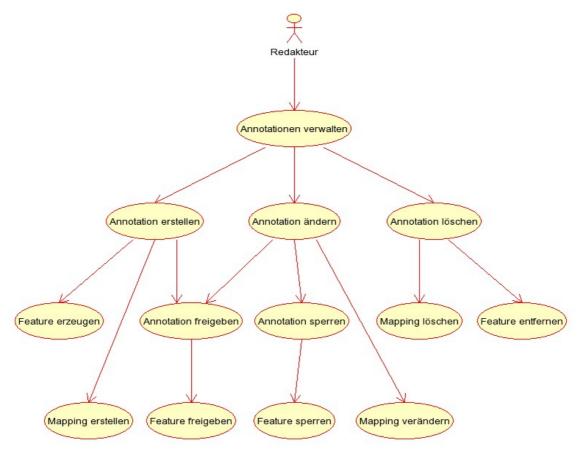


Abbildung 23: Use Case Diagramm Redakteur - Annotationen

Da jede Annotation immer mindestens einem Feature zugeordnet werden muss, müssen die drei Grundoperationen die Mappings erzeugen, verändern oder entfernen können. Zusätzlich muss die Möglichkeit bestehen, dass Annotationen freigegeben und gesperrt werden können, um die Anzeige zu ermöglichen oder zu verhindern. Ebenso müssen Features gesperrt oder freigegeben werden können. Ein Feature soll nur dann angezeigt werden, wenn mindestens eine per Mapping zugehörige Annotation freigegeben ist und der Redakteur die Anzeige des Features innerhalb der WebMap wünscht.

6.2.4 Systemdesign

In der Systemarchitektur (Kapitel 6.2.2) sind bereits einige Vorgaben bezüglich des Systemaufbaus getroffen, die sich auf die Redaktions-Schnittstelle des GIS-Integrators auswirken. Einen direkten Zugang zum GIS-Integrator ist nur den Redakteuren vorbehalten, die über ein entsprechendes Login verfügen. Alle anderen Benutzer greifen über die angebotenen beziehungsweise implementierten Schnittstellen aus dem CMS beziehungsweise der Drittsysteme zu.

Der Redakteur bedient den GIS-Integrator über ein webbasiertes Interface, über das er Web-Maps definieren und verwalten, diese an das CMS übergeben, eigene Annotationen erstellen und die von anderen Benutzern verwalten kann.

Die Übersicht über alle vom Redakteur festgelegten WebMaps bildet den Ausgangspunkt für alle Aktionen, die der Redakteur durchführen kann. Von hier aus ist es ihm möglich, alle Web-Maps zu ändern, zu löschen, neue hinzuzufügen und die Annotationen zu verwalten. Das Aktivitätsdiagramm in Abbildung 24 veranschaulicht die Verwaltung der WebMaps.

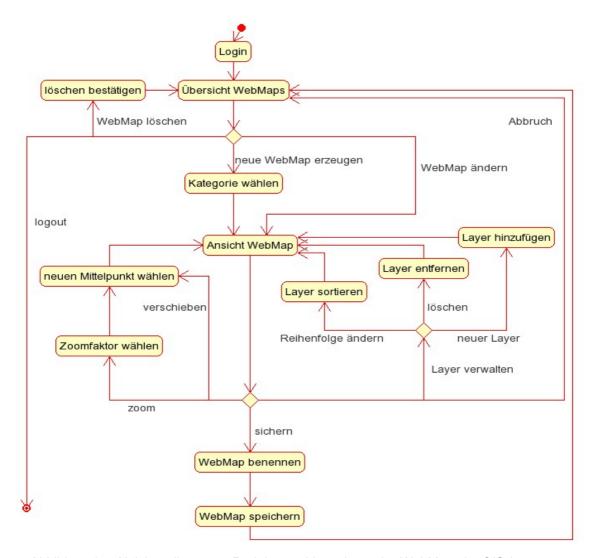


Abbildung 24: Aktivitätsdiagramm Redakteur - Verwaltung der WebMaps im GIS-Integrator

Wie aus dem Aktivitätsdiagramm in Abbildung 24 zu ersehen ist, können alle WebMap-Operationen aus der Übersicht beziehungsweise der Auflistung der WebMaps ausgeführt werden. Um eine neue WebMap zu erzeugen muss der Redakteur wählen, zu welcher Kategorie diese WebMap gehören soll. Eine Kategorie ist in diesem Fall das Umfeld, zu dem die WebMap gehört. Die Kategorien werden durch die Capabilities des verwendeten WMS-Server vorgegeben

(siehe dazu Kapitel 5.5.3.1). Kategorien könnten somit beispielsweise "Bebauungspläne", "Flurstückpläne", "Infrastrukturpläne", usw. sein und werden durch den Administratordes WMS eingestellt. Innerhalb dieser Kategorien stellt der WMS dann zugehörige Kartenlayer zur Verfügung, die in der Ansicht der WebMap als Auswahlmöglichkeiten angezeigt werden. Der Redakteur kann Layer nur aus dieser Auswahl wählen. So wird vermieden, dass der Redakteur versehentlich Layer miteinander kombiniert, die zusammen entweder keinen Sinn ergeben oder nicht gewünscht sind.

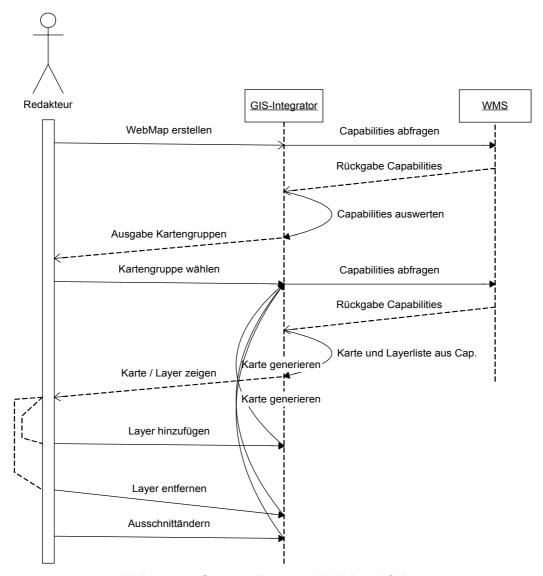


Abbildung 25: Sequenzdiagramm WebMap definieren

In der Ansicht der WebMap kann der Redakteur jetzt beliebig Layer der WebMap hinzufügen oder wieder entfernen. Zusätzlich kann deren Reihenfolge festgelegt werden, um bestimmte Überblendungen der einzelnen Layer einzustellen. Um festzulegen, welcher Ausschnitt für die WebMap eingestellt werden soll, kann der Redakteur über die Zoom- und Verschiebe-Funktion den gewünschten Darstellungsausschnitt der Karte frei festlegen. Hat er alle Einstellungen vorgenommen, wird die WebMap mit einem Namen versehen und gespeichert. Fortan wird sie in der Übersicht über die WebMaps geführt. Änderungen an WebMaps lassen sich ebenso vor-

nehmen, wie die zuvor beschriebene Erstellung einer WebMap. Bei der Änderung einer Web-Map hat der Redakteur nicht die Möglichkeit eine Kategorie zu wählen. Da eine WebMap bereits in einem Umfeld erstellt worden ist, entfällt dieser Schritt.

Das Sequenzdiagramm in Abbildung 25 zeigt den Ablauf in den beteiligten Komponenten bei der Definition einer neuen WebMap.

Wie zu sehen ist, greift der GIS-Integrator bei der Definition einer WebMap immer wieder auf die Capabilities zurück, die der WMS zur Verfügung stellt. Abhängig von diesen Capabilities passt sich für den Redakteur die Auswahl der verfügbaren Layer für die neue WebMap an.

Annotationen kann der Redakteur zu festgelegten WebMaps innerhalb des GIS-Integrators erstellen. Die Annotationen werden einem Punkt auf der Karte zugeordnet. Zusätzlich können an bereits erstellte Annotationen weitere angehängt werden, wodurch eine Chronologie der Annotationen zu einem Punkt auf der Karte möglich wird, womit zum Beispiel eine Art Diskussion zu dem Punkt geführt werden kann.

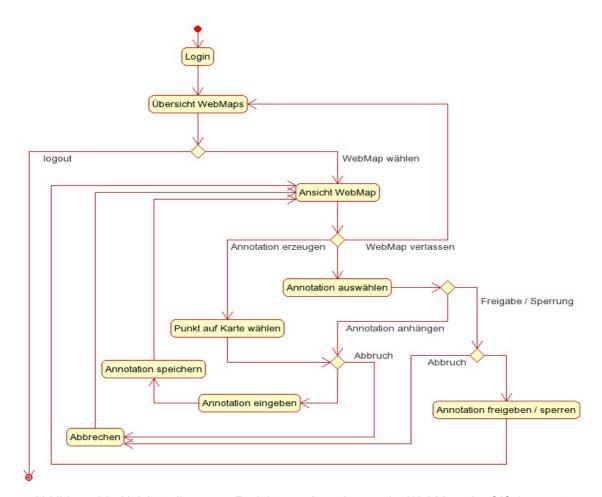


Abbildung 26: Aktivitätsdiagramm Redakteur - Annotierung der WebMaps im GIS-Integrator

Das Aktivitätsdiagramm in Abbildung 26 verdeutlicht die Abläufe, die der Redakteur durchlaufen kann, um Annotationen im GIS-Integrator zu verwalten. Generell hat der Redakteur die Möglichkeit eine neue Annotation zu erzeugen. Dazu wählt er einen Punkt auf der Karte, gibt die gewünschte Annotation ein und speichert diese. Weiterhin kann er eine bereits existierende

Annotation auswählen. An diese ausgewählte Annotation kann er dann entweder eine weitere anhängen, oder die ausgewählte Annotation freigeben beziehungsweise sperren.

Auf diese Weise lassen sich alle definierten WebMaps mit Annotationen versehen und entsprechend freigeben. Freigegebene Annotationen können auch außerhalb des GIS-Integrators abgerufen werden, wohingegen gesperrte nur für den Redakteur im GIS-Integrator sichtbar sind. Es können auf die gleiche Art auch Annotationen eingesehen und bearbeitet werden, die von Benutzern aus dem CMS oder aus einem Drittsystem heraus erstellt wurden.

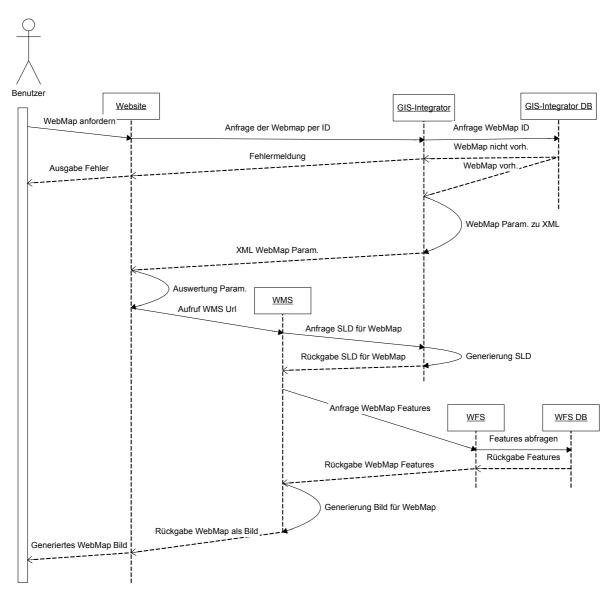


Abbildung 27: Sequenzdiagramm WebMap anzeigen

Benutzer haben keinen direkten Zugriff auf den GIS-Integrator. Sie greifen auf die definierten WebMaps grundsätzlich über deren Einbindung in den Klienten-Systemen zu. Ein Klienten-System kann zum Beispiel ein CMS oder ein anderes beliebiges Drittsystem sein. Die Navigations- und Annotationsmöglichkeiten der dargestellten Karten sind abhängig von der Implementation im Klienten-System. Für bestimmte Karten kann so zum Beispiel im Klienten-System.

tem festgelegt werden, dass Annotationen nicht möglich oder nur bestimmte Navigationsmöglichkeiten vorhanden sein sollen. Auf die notwendigen Daten und Funktionen des GIS-Integrators greift das Klienten-System dabei über eine Client-API zu. Um die Darstellung der
Karten im Klienten-System zu ermöglichen, können also die Methoden der Client-API
verwendet werden, die es vereinfachen eine WebMap abzurufen, darzustellen, zu navigieren
und gegebenenfalls zu annotieren. Prinzipiell können über die Client-API alle Funktionen und
Methoden implementiert werden, die auch die Redaktionsschnittstelle des GIS-Integrators
selber zur Verfügung stellt.

Im Sequenzdiagramm stellt sich der Aufruf einer WebMap aus einem Klienten-System (Website) wie in Abbildung 27 dar. Wie aus dem Diagramm zu erkennen ist, dient die Website als eine Art Schnittstelle für den Benutzer, die die notwendigen Komponenten zur Generierung und Anzeige der WebMap als Karte vor dem Benutzer kapselt. Die Website ruft über die Client-API des GIS-Integrators die durch eine ID spezifizierte WebMap als XML-Struktur vom Webservice des GIS-Integrators ab. Aus diesen XML-Daten generiert die Client-API des GIS-Integrators den WMS Aufruf zur Generierung der Karte als Grafik. Der Aufruf enthält eine Referenz auf einen SLD¹, der durch den GIS-Integrator dynamisch erzeugt wird und die Features enthält, die zu den bereits erstellten Annotationen gehören. So kann der WMS, auf den in der WebMap definierten Layern und dem SLD, die Karte als Grafik generieren und sie wird über den Browser dem Benutzer dargestellt.

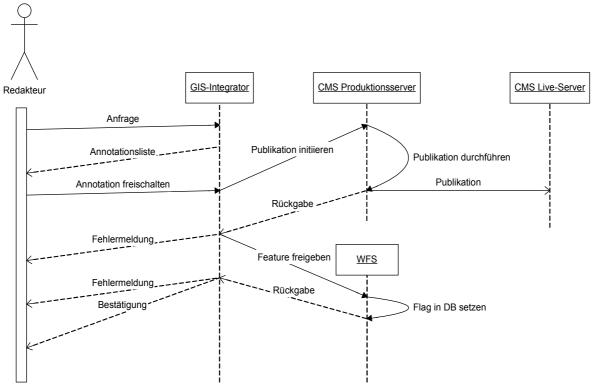


Abbildung 28: Sequenzdiagramm Annotation freigeben

¹ Styled Layer Descriptor, siehe auch Kapitel 5.5.2

Es können alle Operationen im Klienten-System implementiert werden, die für die beabsichtigte Funktionalität benötigt werden. Die Client-API des GIS-Integrators unterstützt dabei den Entwickler mit den notwendigen Methoden, die den Zugriff auf das GIS beziehungsweise den zu den WebMaps gespeicherten Dokumenten im CMS ermöglichen.

Das Sequenzdiagramm in Abbildung veranschaulicht, wie die Erzeugung einer Annotation über eine Website eines Drittsystems (Website) durchgeführt und die Annotation als Dokument im CMS gespeichert wird.

Statt des Benutzers könnte auch ein Redakteur die Annotationen erzeugen. Zusätzlich kann der Redakteur auch direkt über den GIS-Integrator Annotationen erzeugen.

Nachdem eine Annotation erzeugt wurde, kann sie durch den Redakteur entsprechend freigegeben werden. Diese Freigabe ist nur durch den Redakteur über den GIS-Integrator möglich oder durch eine spezielle Implementation in einem Drittsystem, welches den Freigabemechanismus über die Client-API steuert. Im Sequenzdiagramm in Abbildung 28 wird die Freigabe über den GIS-Integrator dargestellt.

Jede Annotation kann vom Redakteur über den GIS-Integrator wieder gelöscht werden. Ist in einem Drittsystem über die Client-API die Funktion zum Löschen von Annotationen implementiert, dann kann dies auch darüber durchgeführt werden. Abbildung 29 zeigt das Sequenzdiagramm, wie der Redakteur über den GIS-Integrator Annotationen löschen kann.

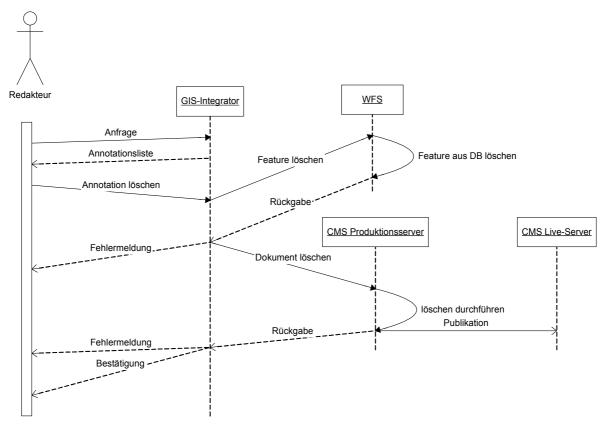


Abbildung 29: Löschen einer Annotation

Die Schnittstellen, die im GIS-Integrator zur Umsetzung der zuvor dargestellten Funktionen zum CMS und zum GIS benötigt werden, sind in den folgenden Kapiteln näher erläutert.

6.2.4.1 Schnittstelle zum GIS

Der GIS-Integrator muss über eine Schnittstelle zum GIS verfügen, damit er den, im vorherigen Kapitel dargestellten, Funktionsumfang erfüllen kann. Dazu gehören die Konfiguration der WebMaps, die Darstellung der WebMaps als Karten und die Annotation der WebMaps. Wie in den Systemanforderungen festgelegt wurde, soll dabei auf OGC™ konforme Dienste zurückgegriffen werden. In Kapitel 6.2.4 sind in den Sequenzdiagrammen die dazu benötigten Dienste bereits erwähnt worden. Der GIS-Integrator greift auf den Web Map Service (WMS) und den Web Feature Service (WFS) zurück. Der WMS wird für die Konfiguration und die Darstellung der WebMaps verwendet und der WFS wird benötigt, um zu den Annotationen gehörige Features zu verwalten.

Die Schnittstelle zum GIS muss also auf verschiedene Komponenten des GIS zugreifen. Abbildung 30 zeigt die GIS Schnittstelle in ihrem Aufbau und mit den notwendigen Verbindungen.

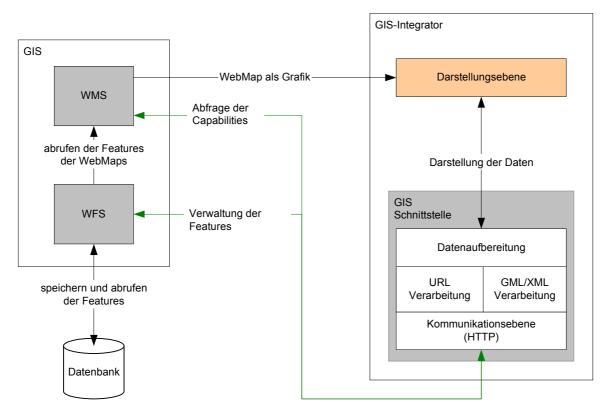


Abbildung 30: Überblick über die GIS Schnittstelle

 Konfiguration der WebMaps
 Für die Konfiguration der WebMaps benötigt der GIS-Integrator nur die Capabilities des WMS. Die Schnittstelle muss dementsprechend die Capabilities abrufen können, die sie

im GML-Format vom WMS geliefert bekommt. Das GML wird von einem XML Parser eingelesen und entsprechend aufbereitet. Daraus wird dem Redakteur dann die Auswahl der Kategorien und der darstellbaren Layer generiert. Außerdem legen die Capabilities fest, in welchem Kartenausschnitt sich die Objekte der darstellbaren Layer befinden.

Während der Konfiguration einer WebMap werden die vom Redakteur ausgewählten Layer als Gesamtkarte angezeigt. Es wird dazu durch die Schnittstelle eine URL generiert, über die der WMS aufgerufen wird. Der WMS generiert über diesen Aufruf die Gesamtkarte als Grafik. Die Karte zeigt die Zusammenstellung der vom Redakteur ausgewählten Layer und den von ihm eingestellten Kartenausschnitt.

• Darstellung einer WebMap

Um eine WebMap darzustellen, muss die Schnittstelle zum GIS, wie bei der Konfiguration der WebMap, eine URL generieren, mit der der WMS aufgerufen wird. Dieser liefert dann die Karte als Grafik zurück, und zwar mit den Layern und dem Kartenausschnitt, die für die anzuzeigende WebMap gespeichert sind. Die URL kann durch die Schnittstelle so erweitert werden, dass die Karte zusätzlich die Features beinhaltet, die zu den Annotationen gehören, welche für die WebMap bereits erstellt wurden.

• Features der Annotationen

Die Schnittstelle zum GIS muss bei der Erstellung von Annotationen Features an den WFS übermitteln, die dieser speichert. Dem WFS muss die Schnittstelle dazu übermitteln, um welche Art Feature (Punkt, Quadrat, sonst. geom. Figur) es sich handelt und wo es positioniert sein soll. Um dies zu gewährleisten, werden die notwendigen Daten an die Schnittstelle übergeben. Die Schnittstelle generiert daraufhin GML, welches an den WFS über eine HTTP POST Anfrage gesendet wird. Der WFS wertet das GML aus und erzeugt das im GML codierte Feature in der Datenbank und gibt eine entsprechende Statusmeldung zurück, die von der Schnittstelle ausgewertet wird. Auf die gleiche Weise können Features wieder entfernt oder verändert werden.

Der WFS wird für die Darstellung der Features, die zu den erstellten Annotationen gehören, ebenfalls benötigt. Wir zuvor beschrieben, wird die Karte durch den WMS generiert. Dieser benötigt dazu die geometrischen Daten der einzelnen Layer, die sich zum Beispiel im Dateisystem, in einer Datenbank oder einem anderen GIS Server befinden können. Um die Features mit in die Karte zu integrieren, ruft der WMS die Daten der anzuzeigenden Features über einen URL Aufruf beim WFS ab und integriert sie in die Karte.

Wie die Abbildung 30 zeigt, kapselt die GIS Schnittstelle die Kommunikation mit dem WMS und dem WFS und stellt der Darstellungsebene nur die notwendigen Daten zur Verfügung. Die Grafik der WebMap wird der Darstellungsebene direkt vom WMS geliefert, da die Grafik nur durch eine URL vom WMS generiert wird, die von der GIS-Schnittstelle für jede WebMap erstellt wird. Alle anderen Anfragen an den WMS oder den WFS werden über URL Auf-

rufe, die gegebenenfalls mit GML Daten erweitert sind, ausgeführt. Die Antwort wird als GML geliefert, welches durch die XML/GML Verarbeitung und die Datenaufbereitung für die Darstellungsebene aufbereitet wird.

6.2.4.2 Schnittstelle zum CMS

Da der GIS-Integrator Unterstützung liefert, um GIS Funktionalität in andere Systeme, unter anderem auch CMS, zu integrieren, benötigt er Schnittstellen, die dies erleichtern. Beim GIS-Integrator wird dabei die in Kapitel 4.4.1 beschrieben Mischform der Dienstintegration verwendet. Dies bedeutet, über seine Schnittstellen können sowohl komplexere Funktionen ausgeführt, als auch Datensätze abgerufen werden.

Da für das, in dieser Arbeit behandelte Projekt, nur die Integration von GIS Diensten in ein CMS im Vordergrund stehen, wird folgend die Schnittstelle nur im Zusammenhang mit einem CMS behandelt werden. Der GIS-Integrator lässt sich aber auch in beliebige andere Drittsysteme über die Schnittstelle einbinden.

Die Schnittstelle zum CMS muss mehrere Eigenschaften aufweisen. Es müssen Referenzen auf im GIS-Integrator definierte WebMaps im CMS gespeichert und vom CMS abgerufen werden können. Außerdem sollen Annotationen erstellt werden können, die der GIS-Integrator speichert und das CMS später abrufen kann. Die Speicherung der Annotationsdaten soll dabei im GIS-Integrator, im CMS selber, oder einem anderen System möglich sein.

Zum Abruf der im GIS-Integrator definierten WebMaps, wird im CMS ein Modul implementiert, welches über einen URL Aufruf an den GIS-Integrator die gewünschte WebMap anfordert. Diese wird vom GIS-Integrator in der Datenbank gesucht und deren Daten als XML zurückgegeben. Die WebMap Daten werden über die XML Verarbeitung und die Datenaufbereitung so aufbereitet, dass sie zu einen WMS Aufruf zusammengesetzt werden können. Diese Funktionen sind in der Client-API des GIS-Integrators implementiert. In der Darstellungsebene kann dann die WebMap als Karte angezeigt werden, die vom WMS als Grafik generiert wurde.

Auf die gleiche Weise werden die Annotationen vom GIS-Integrator abgerufen. Die Daten der Annotationen können sich allerdings in unterschiedlichen Systemen befinden. Sie können im GIS-Integrator selbst gespeichert sein, dann müssen sie lediglich geladen und als XML zurückgegeben werden. Sie können aber auch im CMS gespeichert sein, dann werden über einen Aufruf an das CMS die Annotationsdaten dort als XML abgerufen, umgewandelt und zurückgegeben. Als dritte Möglichkeit können die Annotationsdaten in anderen Drittsystemen gespeichert sein. Um an diese Daten zu gelangen und sie zurückzugeben, geht der GIS-Integrator genau wie beim CMS vor.

Wie erwähnt können die Daten zu den Annotationen vom CMS im GIS-Integrator gespeichert werden. Sie werden im CMS nach der Eingabe vom Benutzer in XML umgewandelt und über

eine URL Anfrage an den GIS-Integrator gesendet. Dieser arbeitet die Annotationsdaten entsprechend auf und speichert sie entweder in seiner eigenen Datenbank, im CMS oder einem anderen System. Zusätzlich verwaltet er eine Zuordnung zwischen den Annotationsdaten und dem dazugehörigen Feature (siehe Kapitel 6.2.4.1). So kann der GIS-Integrator zu den Features die Annotationen liefern und umgekehrt.

Abbildung 31 zeigt eine Darstellung der CMS-Schnittstelle im GIS-Integrator und deren Zugriffe auf das CMS.

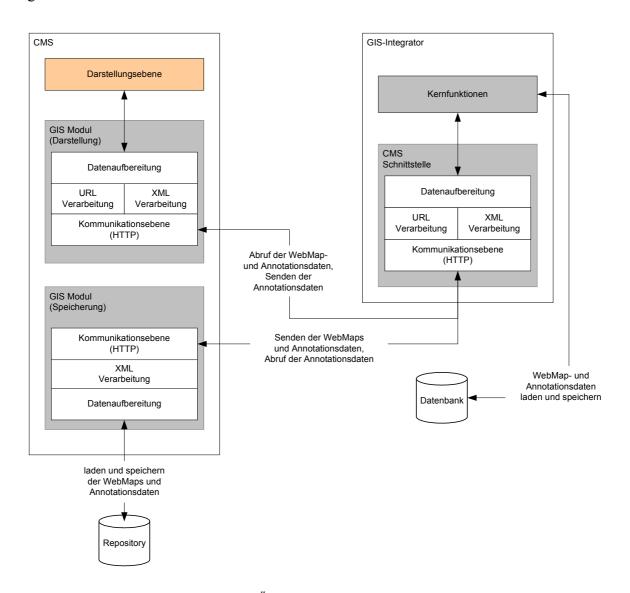


Abbildung 31: Überblick über die CMS Schnittstelle

6.3 Anwendungsbeispiel "Gemeinde Seevetal"

Die Gemeinde Seevetal gehört zur Metropolregion Hamburg und plante mit als erste die Darstellung von Flächennutzungsplänen, Bebauungsplänen und weiteren Plänen innerhalb ihrer In-

ternetpräsenz. Ebenso sollen die gesetzlich vorgeschriebenen Beteiligungsverfahren über das Internet möglich sein. Der Bürger soll Einsicht in die Pläne bekommen, sie sich grafisch anzeigen lassen, mit ihnen interagieren und zu Plänen und bestimmten Bereichen der Pläne seine Einwände und Anmerkungen über das Internet äußern können. Die Darstellung der Pläne soll nicht, wie meist umgesetzt, mit einfachen Screenshots der Karten erfolgen, sondern auf den erstellten Geodaten basieren. Dadurch soll es dem Bürger möglich sein, sich die Karten in einem beliebigen Detailierungsgrad anzusehen, Ausschnitte zu vergrößern und zu verkleinern, Informationen zu einzelnen Objekten zu erhalten und dergleichen. Seine Einwände und Anmerkungen soll der Bürger direkt zu bestimmten Bereichen auf den dargestellten Karten machen können. Es muss gewährleistet werden, dass Datenformate für Geodaten unterschiedlicher Hersteller von Geo-Software verwendet, und diese auch gemeinsam auf einer Karte dargestellt werden können.

Für die Umsetzung des Projekts sind zwei Phasen vorgesehen. In der ersten Phase wurde eine prototypische Website entwickelt, welche die Abteilung Stadtplanung der Gemeinde Seevatal für den Bürger repräsentiert. Hier sollen dem Bürger Informationen zur Stadtplanung innerhalb der Gemeinde Seevetal zur Verfügung gestellt werden. Dies beinhaltet unter anderem allgemeine Informationen zur Stadtplanung, Ansprechpartner, Verfahrensschritte und die Darstellung der geplanten, aktuellen und abgeschlossenen Flächennutzungspläne und Bebauungspläne. Das für die Flächennutzungspläne und Bebauungspläne vorhandene Kartenmaterial wird dem Bürger, in dieser ersten Phase, als statisches Material, in Form von Bildern, PDF Dokumenten, oder Ähnlichem zur Verfügung gestellt. Die erste Phase ist bereits umgesetzt und die Website der Abteilung Stadtplanung [Stadtplanung Seevetal 2005] ist über den Webauftritt der Gemeinde Seevetal [Seevetal 2005] erreichbar.

In der zweiten Phase sollen Bürger interaktive Karten abrufen können, in denen sie Ausschnitte und Bereiche detaillierter betrachten können. Weiterhin sollen die interaktiven Bürgerbeteiligungsverfahren in dieser Phase umgesetzt werden. Laufende Verfahren können also nicht nur eingesehen werden, sondern der Bürger kann online seine Anmerkungen zum Verfahren oder zu bestimmten Punkten der Verfahren eingeben. Diese zweite Phase befindet sich in der Planung. Ein erster Schritt ist das Pilotprojekt des GIS-Integrator, der unter anderem solche Beteiligungsverfahren unterstützen soll.

Ein wichtiger Punkt bei der Umsetzung der Website, der Abteilung Stadtplanung der Gemeinde Seevetal, war es, dass wenig Barrieren für den Bürger bei der Benutzung vorhanden sind. Es sollen demnach keine Plugins Verwendung finden, die vorher erst herunter geladen und installiert werden müssen. Außerdem sollen auch Bürger mit langsamen Internetanbindung sich nicht nur die Website, sondern auch die Karten mit angemessenen Downloadgeschwindigkeiten ansehen und die Bürgerbeteiligung durchführen können.

Die Karten sollen von den Redakteuren der Gemeinde Seevetal zusammengestellt und über die Website publiziert werden können. Dies bedeutet, auch Redakteuren ohne Wissen über GIS und Geodaten soll es möglich sein, Karten und deren Informationsebenen zusammenzustellen und mit redaktionellen Inhalten anzureichern.

Das Projekt ist bei der Planung und Umsetzung an bestimmte Rahmenbedingungen gebunden und es wurden aufgrund von Umständen gewisse Festlegungen getroffen. Die Wahl des CMS fiel auf CoreMedia® CAP in der Version 4.1. Dies lag darin begründet, dass CoreMedia® bereits von der Gemeinde Seevetal als CMS zur Pflege der Website genutzt wurde und so wenig Einarbeitung seitens der Redakteure notwendig war. Außerdem findet CoreMedia® zur Zeit Einsatz in weiteren Websites und Portalen von Gemeinden, Kommunen und Städten der Metropolregion Hamburg. Zusätzlich besteht, aufgrund der historischen Entwicklung und der Verbindung des Arbeitsbereiches STS der TU Harburg [STS 2005] mit der CoreMedia® AG [CoreMedia 2005], ein guter Kontakt, und somit ein guter Know-How Transfer. Bei der Auswahl des WMS und des WFS fiel die Entscheidung zugunsten des open Source Frameworks Deegree [deegree 2005], da durch Dr. Kai Uwe Krause [K.-U. Krause 2005], an der TU Harburg, auch hier eine gute Verbindung zu den Entwicklern des Frameworks besteht. Das Framework Deegree beinhaltet außerdem einige Referenzimplementationen der von der OGCTM definierten Services, was eine sehr standardnahe Implementation sicherstellt. Die Implementation des GIS-Integrators selber wurde in Java vorgesehen, das hier weitreichende Kompetenzen am Arbeitsbereich STS vorhanden sind und sich sowohl CoreMedia, als auch Deegree dieser Programmiersprache in der Basis bedienen. Als Datenbank für den WFS und den GIS-Integrator wird eine Oracle Datenbank verwendet, da eine Installation vorhanden ist, und diese ebenfalls von CoreMedia® genutzt wird.

Es wird im Folgenden nur auf den Projektteil eingegangen werden, der sich mit der Integration der GIS Funktionalität in CoreMedia® beschäftigt. Es wird also nur auf die oben erwähnte zweite Phase eingegangen. Die erste Phase, die Umsetzung der Website des Bereiches Stadtplanung der Gemeinde Seevetal, ist mit einem einfachen Dokumentmodell mit CoreMedia® realisiert und abgeschlossen worden.

6.3.1 Systemaufbau

Da die Website der Abteilung Stadtplaung der Gemeinde Seevetal bereits ohne GIS Funktionalität für den Bürger freigeschaltet ist, wurde eine zweite identische Systemumgebung aufgesetzt, welche für die Entwicklung der Integration der GIS Funktionalität genutzt wird. Es ist also ein Produktivsystem und ein Entwicklungssystem im Einsatz, wodurch keine Störung des öffentlichen Betriebes erfolgt.

Wie auch in der Systemarchitektur des GIS-Integrators, aus dem ersten Teil des Kapitel 6, erkennbar ist, soll es, trotz der gesteckten Rahmenbedingungen (s.o.), möglich sein, nicht nur CoreMedia® und Deegree zu verwenden, sondern auch andere CMS und OGCTM konforme GIS. Der GIS-Integrator wird also nicht in eines der Systeme integriert, sondern ist eigenständig. Schnittstellen zum CMS sind in dem Pilotprojekt entsprechend für CoreMedia® angepasst, sind allerdings für andere CMS austauschbar. Die Schnittstellen zu Deegree sind so ausgelegt, dass Deegree durch ein anderes OGCTM konformes GIS ausgetauscht werden kann, ohne Änderungen am GIS-Integrator vornehmen zu müssen. Die Schnittstelle des GIS-Integrator zur Oracle® Datenbank ist ebenfalls so ausgelegt, dass sie durch eine andere ausgetauscht werden kann, wenn eine andere Datenbank verwendet werden soll.

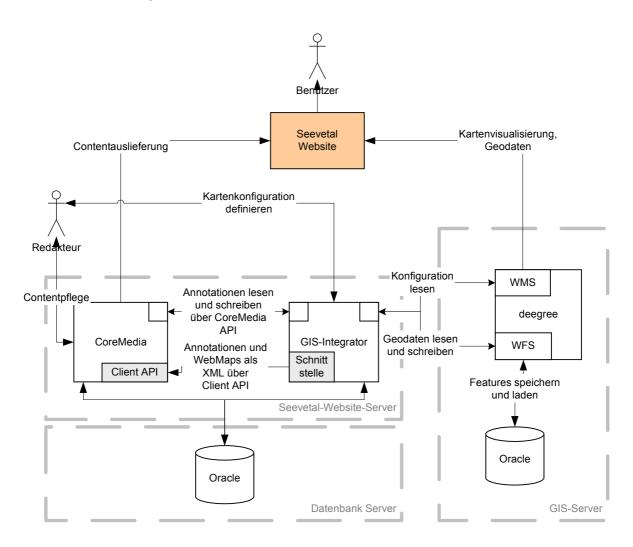


Abbildung 32: Struktur der GIS Integration in die Website der Gemeinde Seevetal

Die Abbildung 32 zeigt die konkrete Struktur, wie sie für das Anwendungsszenario der Gemeinde Seevetal geplant und prototypisch umgesetzt wurde. Der Webauftritt selber läuft unter einer eignen URL, unabhängig vom Hauptauftritt der Gemeinde Seevetal. Ein Corementarionen System, wird für die Pflege des Webauftritts genutzt. Auf dem gleichen System ist auch der

GIS-Integrator aufgesetzt, was die Kommunikation mit dem CoreMedia® System erleichtert. Auf einem Datenbankserver läuft eine Oracle® Datenbank, die von CoreMedia® und dem GIS-Integrator genutzt wird. Deegree läuft in auf einem zweiten System, welches die notwendige Rechenleistung für die Generierung der Karten als Grafiken zur Verfügung stellt. Auf dem gleichen System ist eine zweite Oracle® Datenbank aufgesetzt, die vom WFS des Deegree genutzt wird.

Der Benutzer nutzt die Website wie gewohnt mit seinem Browser. Außer JavaScript wird kein weiteres Plugin für den Browser benötigt, um die Anzeige und Navigation der Karten und die Bürgerbeteiligung zu nutzen.

6.3.2 Konfiguration und Anpassungen

Wichtig für das Pilotprojekt war es, dass für den Bürger keinerlei Barrieren bei der Nutzung der Website aufgebaut werden. Somit wurde das Projekt so ausgelegt, dass jeder Bürger die komplette Website ohne Änderungen, Anpassungen oder Umkonfiguration seiner Browsersoftware nutzen kann.

Für den GIS-Integrator, Deegree, CoreMedia® und die Oracle® Datenbank waren bestimmte Anpassungen und Konfigurationen notwendig.

Der GIS-Integrator basiert im Wesentlichen auf dem in den den Kapiteln 6.1 und 6.2 beschriebenen Aufbau und den dort dargestellten Strukturen und Schnittstellen. Es wurden aber gewisse Vereinfachungen in einigen Bereichen vorgenommen, die sich aus den Projektvorgaben ergeben haben.

Die Redakteure verwenden für die Verwaltung der WebMaps den GIS-Integrator und CoreMedia, um die WebMaps auf der Website zu präsentieren.

Die Bedienung des GIS-Integrator erfolgt über einen Webbrowser. Es werden hier alle Web-Maps verwaltet, die über den GIS-Integrator zur Verfügung gestellt werden sollen. Der Redakteur kann WebMaps erstellen, ändern, löschen und an Corementa übergeben. Um den Redakteuren neue Layer im GIS-Integrator zur Verfügung zu stellen, muss ein Administrator den Deegree WMS konfigurieren. Er stellt die Geodaten, auf denen der neue Layer basiert, für den Deegree WMS bereit. Dazu muss er, je nach Art der Geodaten, diese zum Beispiel als Shape Datei im Dateisystem des Deegree WMS speichern, in eine Datenbank übertragen, oder in den WMS eines kommerziellen Anbieters (zum Beispiel ESRI® oder Autodesk®) integrieren. Die Capabilities des Deegree WMS werden vom Administrator angepasst und die neuen Geodaten dort integrieret. Danach ist der neue Layer im GIS-Integrator, der die Capabilities des WMS ausliest, bekannt und kann von den Redakteuren eingebunden werden. Alle weiteren Tätigkeiten können nun von den Redakteuren durchgeführt werden, und es wird kein Administrator mehr benötigt.

Die Speicherung der WebMaps erfolgt in der Oracle Datenbank. Diese wird direkt vom GIS-Integrator über die JDBC Schnittstelle angesprochen. Die Datenbanktabelle für die WebMaps hat den folgenden Aufbau:

Spaltenname	Spaltentyp	Beschreibung		
ID	NUMBER	eindeutige Kennung		
NAME	VARCHAR2 (512)	Name der WebMap		
LAYERS	VARCHAR2 (4000)	kommagetrennte Layerliste		
STYLES	VARCHAR2 (4000)	kommagetrennte Styleliste		
SERVERURL	VARCHAR2 (512)	URL des WMS		
LAYERFOLDER	VARCHAR2 (512)	Kategorie in den Capabilities		
SRS	VARCHAR2 (64)	geografisches Bezugssystem		
BBOXMINX	VARCHAR2 (64)	Definition Boundingbox		
BBOXMAXX	VARCHAR2 (64)			
BBOXMINY	VARCHAR2 (64)			
BBOXMAXY	VARCHAR2 (64)			

Tabelle 2: Oracle Datenbanktabelle für die WebMaps

Die Angaben in der Datenbanktabelle korrespondieren mit den Angaben in den Capabilities, wie sie im WMS definiert sind. Die Spalten LAYERS und STYLES legen fest, welche Kartenlayer die WebMap enthält, in welcher Reihenfolge sie übereinander liegen und in welchem Style sie vom WMS dargestellt werden sollen. Die Spalte LAYERFOLDER definiert dem GIS-Integrator, zu welcher Kategorie die WebMap gehört. Dies ist notwendig, um die möglichen Layer für diese WebMap aus dem Capabilities zu ermitteln und dem Redakteur als Auswahl anzubieten, wenn er die WebMap ändert. Um sicherzustellen, dass ältere WebMaps durch die Redakteure geändert werden können, muss der Adminstrator also darauf achten, dass er Namen bestehender Kategorien oder Layer nicht ändert oder löscht, sondern nur neue erstellt oder weitere hinzufügt.

Die im GIS-Integrator definierten WebMaps werden vom Redakteur an CoreMedia® übergeben, um dort in die Website eingebunden werden zu können.

Die von den Bürgern auf der Website erstellten Annotationen werden ebenfalls in Coremedia gespeichert. Annotationen sind Punkten auf WebMaps zugeordnet, die vom WFS durch Features visualisiert werden. Die Verbindungen zwischen den Features und den Annotationsdokumenten in Coremedia werden im GIS-Integrator gespeichert. Es müssen dazu Zuordnungen von Annotationen zu Features innerhalb der WebMaps abgebildet werden. Diese Zuordnung wird Mapping genannt und vom GIS-Integrator ebenfalls in der Oracle Datenbank gespeichert. Die Datenbanktabelle der Mappings hat den folgenden Aufbau:

Spaltenname	Spaltentyp	Beschreibung
ID	NUMBER	eindeutige Kennung
WEBMAP_ID	NUMBER	ID der WebMap
FEATURE_ID	VARCHAR2 (512)	ID des Features im WFS
CMS_ID	VARCHAR2 (512)	Dokument ID in Coremedia
PUBLISHED	VARCHAR2 (1)	Flag ob Annotation veröffentlicht
PUBLISHED_FEATURE	VARCHAR2 (1)	Flag Feature ob veröffentlicht

Tabelle 3: Oracle Datenbanktabelle der Mappings

Über die WEBMAP_ID, die FEATURE_ID und die CMS_ID sind WebMap, Feature und COREMEDIA® Dokument (Annotation) einander eindeutig zugeordnet. Die Spalte PUBLIS-HED gibt an, ob die Annotation vom Redakteur freigegeben wurde. Da mehrere Annotationen einem Feature zugeordnet sein können, gibt die Spalte PUBLISHED_FEATURE an, ob mindestens eine der Annotationen, die zu einem Feature gehören, freigegeben ist und somit auch das dazugehörige Feature.

Der Zugriff des GIS-Integrator auf die Oracle® Datenbank ist direkt über die JDBC Schnittstelle implementiert. Der Zugriff des GIS-Integrator auf Coremedia®, um die WebMap Referenzen und die Annotationen zu speichern, ist über die Coremedia® API realisiert. Der GIS-Integrator baut über die API eine Verbindung zu Coremedia® auf, und schreibt die WebMap Referenzen und Annotationen jeweils direkt in einen bestimmten Ordner im Repository von Coremedia®. Abgerufen werden die Annotationen vom GIS-Integrator auf die gleiche Weise. Dies geschieht mit Hilfe der Dokument ID (Spaltenname: CMS_ID) von Coremedia®, die in der Mapping Tabelle gespeichert ist.

6.3.3 Einbindung in CoreMedia[©]

Im vorherigen Kapitel wird deutlich, dass sowohl die Referenzen auf die WebMaps, als auch die Annotationen zu WebMaps in CoreMedia® gespeichert werden. Dazu musste das in der ersten Phase des Projektes entwickelte Dokumentmodell von CoreMedia® entsprechend erweitert werden, um die WebMaps des GIS-Integrators referenzieren und die Annotationen

zu den WebMaps speichern zu können. Daraus ergab sich die in Abbildung 33 dargestellte Erweiterung des Dokumentmodells für CoreMedia.

Wie in der Abbildung 33 zu sehen ist, wird zur Referenzierung der WebMap die ID und die URL des GIS-Integrators im Coremedia Dokument gespeichert. Die auf der Website darzustellende WebMap ist damit eindeutig festgelegt. Die Annotationen bestehen aus einem Betreff und einem freien Text, die vom Bürger eingegeben werden können. Zusätzlich muss er seinen Namen und seine Emailadresse angeben.

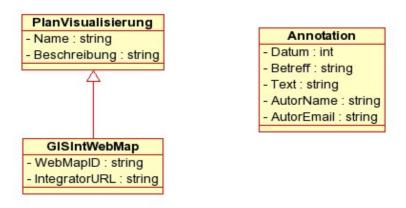


Abbildung 33: Erweiterung des CoreMedia Dokumentmodells

Um den Zugriff auf die Im GIS-Integrator gespeicherten WebMaps und Annotationen zu ermöglichen, verfügt der GIS-Integrator über eine Webschnittstelle, die parametrisierte URL Aufrufe entgegen nimmt. Die Aufrufe werden verarbeitet und ein entsprechendes Ergebnis wird als XML zurückgegeben. Über diese Webschnittstelle können zum Beispiel WebMaps und Annotationen abgerufen werden und weitere notwendige Funktionen des GIS-Integrator durchgeführt werden. Um den Zugriff auf die Webschnittstelle des GIS-Integrators zu vereinfachen, ist eine Java Client-API des GIS-Integrators umgesetzt worden. Diese ist in das Coremedia System eingebunden und ermöglicht es, anhand der in Coremedia erzeugten Referenz auf eine WebMap, deren Daten vom GIS-Integrator abzurufen, einen entsprechenden WMS Aufruf zu generieren, und so die Karte auf der Website darzustellen.

Um den, für das Pilotprojekt definierten Funktionsumfang zu ermöglichen, verfügt die Client-API über einige Methoden, die Unterstützung bei der Implementation der Website lieferten. Dazu gehören die folgende Methoden:

- laden einer WebMap
 Auf Basis einer gegebenen ID wird die WebMap von GIS-Integrator geladen und als Objekt zurück gegeben.
- Generierung eines WMS Aufruf
 Es wird eine URL zurück gegeben, die einen WMS Aufruf darstellt, der die WebMap mit den übergebenen Parametern (Bounding Box, etc.) generiert. Die WebMap kann dabei

mit und ohne die Darstellung der zu den Annotationen gehörenden Features erzeugt werden

- Annotation zu einer Position hinzufügen
 Eine gegebene neue Annotation wird einer übergebenen Position auf der WebMap zugeordnet. Im GIS-Integrator wird dazu das Feature an der entsprechenden Position
 erzeugt, und die Annotation gespeichert.
- Annotation zu einem Feature hinzufügen
 Zu einem bereits existierenden Feature wird eine neue Annotation hinzugefügt und im GIS-Integrator gespeichert.
- Annotationen zu einer Position abfragen
 Zu einer bestimmten Position, auf einer WebMap, werden alle dort erstellten Annotationen zurückgegeben.

Die Implementation ist wesentlich vereinfacht, indem die Positionsangaben keine Geokoordinaten, sondern x-y-Koordinaten auf der generierten Kartengrafik darstellen. Diese werden durch den GIS-Integrator Geokoordinaten umgerechnet und das Feature an dieser Position im WFS gespeichert und Annotationen an diesen Koordinaten werden vom GIS-Integrator zurückgegeben.

Um die WebMaps auf der, durch Coremedia generierten, Website darzustellen, ist ein Coremedia Template implementiert worden. Dieses Template liest die WebMap Referenzierung aus Dokumenten des Typs GISIntWebMap (siehe Abbildung 33) aus. Dieses Dokument wurde zuvor durch den Redakteur über den GIS-Integrator an Coremedia übergeben und dort in die Website eingebunden. Mit den Referenzierungsdaten und der Client-API des GIS-Integrators kann das Template jetzt alle notwendigen Daten zur WebMap und den dazugehörigen Annotationen vom GIS-Integrator abrufen. Das Template ist ein dynamisches Template, so dass jeder Klick eines Benutzers auf die generierte Karte eine entsprechende Aktion auslöst (Kartenausschnitt zoomen, verschieben, Annotationen abrufen, etc.), das Template darauf reagiert und die entsprechende neue Ausgabe der Karte liefert. Damit kann dem Benutzer der Website die WebMap voll interaktiv zur Verfügung gestellt werden und er hat die Möglichkeit Annotationen zu Bereichen auf der Karte zu erstellen. Die Pflege der WebMaps und Einbindung in die Website nimmt der Redakteur auf die gewohnte Weise mit dem Coremedia.

Ohne zusätzlichen Installationsaufwand bei den Benutzern und Schulungsaufwand bei den Redakteuren, sind GIS Funktionalitäten mit dem GIS-Integrator in Coremedia integriert worden und lassen sich über darauf basierenden Websites abrufen.

Kapitel 7

Zusammenfassung

Das Ziel des Gesamtprojektes war es, der Abteilung Stadtplanung der Gemeinde Seevetal eine Website zur Selbstdarstellung und Präsentation ihrer Bau- und Planungsvorhaben zur Verfügung zu stellen. Zusätzlich sollte diese Website dem Bürger die Möglichkeit bieten, an Beteiligungsverfahren online teilzunehmen. Dies sollte prototypisch für Bebauungspläne umd Flächennutzungspläne umgsetzt werden.

Die Gemeinde Seevetal in der Metropolregion Hamburg (MRH) angesiedelt und am Internetportal der Metropolregion [MRH 2005] beteiligt. Aus diesem Grund ist in Kapitel 2 das Webportal der MRH näher erläutert. Es werden die Entwicklungsschritte des Webportals (Kapitel
2.1), die darin enthaltenen Inhalte und die damit verfolgten Ziele (Kapitel 2.2) dargestellt.
Weiterhin ist in Kapitel 2.2.2 ein Überblick über Dienste im Bereich eGovernment gegeben,
die den Bürgern der MRH zur Verfügung gestellt werden beziehungsweise in Planung sind.
Um den redaktionellen Aufwand für das MRH-Portal möglichst gering halten zu können, soll
der Inhalt des Portals hauptsächlich aus Diensten und Inhalten der Partner zusammengestellt
werden. Dies ist in den Kapiteln 2.2.3 und 2.2.4 dargestellt, welche die Bereiche Syndikation
und Integration von Daten und Diensten zwischen den Inhaltslieferanten in der MRH aufgreifen.

Kapitel 3 liefert eine Einführung über Content Management Systeme (CMS), inklusive eines Überblicks über deren Aufgaben und Einsatzgebiete, sowie deren Funktionsweise und Leistungsumfang.

Im Kapitel 4 wird die Integration von Daten und Diensten in Web-Portale behandelt. Dort sind verschiedene Topologien (Kapitel 4.1) und Strategien (Kapitel 4.2) dargestellt, sowie Verfahren, mit denen eine Integration von Daten und Diensten umgesetzt werden kann (Kapitel 4.3 und 4.4). Aus der Kombination dieser Integrationsmöglichkeiten leitet sich der Ansatz für die Umsetzung des Pilotprojektes ab.

104 Kapitel 7 Zusammenfassung

Das Einsatzumfeld und der Funktionsumfang von Geografischen Informationssystemen (GIS) ist in Kaitel 5 überblickartig dargestellt. Zusätzlich wird das Open Geospatial Consortium (OGCTM) und der offene OpenGIS-Standard des OGC vorgestellt, sowie deren Bedeutung für das Pilotprojekt herausgearbeitet.

In Kapitel 6 ist die Integration von CMS- und GIS-Komponenten dargestellt sowie die Umsetzung des Pilotprojektes beschrieben. Kapitel 6.1 und 6.2 beschäftigen sich mit dem CMS- unabhängigen Integrationsansatz und gehen auf die besonderen Systemanforderungen, die an eine solche offene Lösung gestellt werden, ein. Kapitel 6.3 beschreibt die konkrete Realisierung der Einbindung der OGC Web Services des Deegree Frameworks in das CMS Coremedia.

Das Gesamtprojekt ist in zwei Phasen eingeteilt. In der ersten Phase wurde die Website der Abteilung Stadtplanung der Gemeinde Seevetal entwickelt, die den Bürgern einen Überblick über die Tätigkeiten der Abteilung bietet. Im Rahmen des Webauftritts werden Flächennutzungspläne und Bebauungspläne, neben der Auslegung im Rathaus, hier veröffentlicht. Die Veröffentlichungen erfolgen als statische Dokumente, die von den Bürgern eingesehen oder herunter geladen werden können. Diese Phase ist im Rahmen des Gesamtprojekts und dieser Arbeit abgeschlossen worden.

In der zweiten Phase wurde die Website erweitert, um dem Bürger die Möglichkeit zu bieten, Flächennutzungspläne und Bebauungspläne interaktiv zu analysieren. Zusätzlich wurden Techniken erarbeitet, mit deren Hilfe die gesetzlich vorgeschriebenen Bürgerbeteiligungsverfahren über die Website mittels rückkanalfähiger *Digitize-on-Screen-*Technik abgewickelt werden können. Die Machbarkeit dieser Erweiterung ist im Rahmen des Projektes gezeigt und prototypisch umgesetzt worden.

Unter [Stadtplanung Seevetal 2005] steht dem Bürger eine Website zur Verfügung, die ihm alle notwendigen Informationen zur Abteilung Stadtplanung der Gemeinde Seevetal und ihren Tätigkeiten liefert. Der Website liegt das CMS Corementagen zugrunde, mit dessen Bearbeitungswerkzeugen die Abteilung ihre Inhalte pflegen kann. Flächennutzungspläne und Bebauungspläne werden auf der Website derzeit statisch dargestellt.

Der als Prototyp entwickelte GIS-Integrator bietet Redakteuren die Möglichkeit, Karten zusammenzustellen und diese über CoreMedia® in eine Website zu integrieren. Ohne Fachwissen über die Erstellung und Pflege von Geodaten kann der Redakteur interaktive Karten auf Websites publizieren, die auf professionell erstellten geografischen Daten basieren und von Open-GIS-konformen Diensten ausgeliefert werden. Über solche Websites ist es möglich Punkte auf den Karten zu markieren, und diese mit textuellen Annotationen zu versehen, ohne die ursprünglichen Geodaten zu manipulieren. Um dies zu realisieren wurde sowohl die Dienst-, als auch die Datenintegration eingesetzt. Für die Zusammenstellung der Karten und deren in-

Kapitel 7 Zusammenfassung 105

teraktive Darstellung werden die OpenGIS-Dienste Web Map Service (WMS) und Web Feature Service (WFS) des Deegree Frameworks verwendet. Deegree wurde dazu als kaskadierender WMS und transaktionaler WFS verwendet und mittels des GIS-Integrator in Corementarionen. Neben der Auslieferung der Karten und Verwaltung der geografischen Features für die Anotationen stellen diese Dienste auch nichtgrafische Daten zur Verfügung, wie zum Beispiel die Capabilities des WMS. Da beide Leistungen der Dienste vom GIS-Integrator verwendet werden, handelt es sich bei dieser Art von Integration um eine Mischform von Dienst- und Datenintegration.

Der GIS-Integrator stellt einen Dienst zur Verfügung, der es ermöglicht, vordefinierte Layer von anderen Systemen abzurufen und zu Kartenkonfigurationen zusammenzustellen. Zusätzlich bietet der GIS-Integrator Funktionen, die es erlauben Annotationen zu erstellen und abzurufen. Der GIS-Integrator bietet somit auch eine Mischfom von Dienst- und Datenintegration. Er bietet Funktionen, wie die Erzeugung von geografisch zugeordneten Annotationen, aber auch den Abruf von Daten, die eine Kartenkonfiguration beschreiben, an. In der vorliegenden Implementation sind diese Dienst- und Datenfunktionen in einer Client-API gekapselt. Es besteht zusätzlich die Möglichkeit diese Dienste und Daten direkt über HTTP beim GIS-Integrator abzurufen.

In dem Projekt wurden alle Möglichkeiten der Dienst- und Datenintegration verwendet. Es wurde vermieden, Daten redundant zu speichern. Alle Daten, unter anderem Geodaten, Annotationsdaten und Konfigurationsdaten der WebMaps sind nur einmal in jeweils einem der beteiligten Systeme vorhanden. Diese Vorgehensweise vereinfachte die Datenverwaltung, da es nicht notwendig wurde Synchronisationsprozesse zu implementieren.

Die Architektur des Gesamtsystems wurde durch Verwendung von Standards so gestaltet, dass es möglich ist die beteiligten Systeme durch kompatible zu ersetzen. Dies gilt für die verwendeten Datenbanken genauso wie für das CMS und das GIS. Die HTTP-Schnittstelle des GIS-Integrators sendet und empfängt die notwendigen Daten über eine definierte XML-Struktur. Die Verarbeitung der XML-Anfragen übernimmt die in Java implementierte Client-API. Die Client-API liefert Methoden, die auf der Website die Anzeige der WebMaps erleichtern, Navigationsmöglichkeiten unterstützt sowie die Erzeugung und den Abruf der Annotationen ermöglicht. Dies bildet die technische Grundlage, um eine Bürgerbeteiligung über die Website der Gemeinde Seevetal zu realisieren.

Kapitel 8

Ausblick

Wie in der Zusammenfassung dargestellt, sind die grundlegenden Funktionen für die Umsetzung eines Bürgerbeteiligungsverfahrens vorhanden. Von Seiten der Abteilung Stadtplanung sind die folgenden zusätzlichen Anforderungen beziehungsweise wünschenswerte Unterstützungen notwendig.

Ein großes Potential an Erweiterungen liegt im Bereich der Beteiligungsverfahren. Dazu können Bemerkungen vom Bürger beziehungsweise den Redakteuren erstellt werden. Von den Bürgern erstellte Bemerkungen (Annotationen) müssen von den Mitarbeitern der Gemeinde Seevetal gefiltert und ausgewertet werden. Eine Filterung ist notwendig, um sinnlose Bemerkungen aussortieren zu können. Dies wird derzeit rudimentär durch die redaktionell geprüfte Freischaltung der Bemerkungen realisiert. Die Bemerkungen müssen übersichtlich aufbereitet und druckfähig zu Verfügung gestellt werden, um sie einsehen, verarbeiten und archivieren zu können. Dabei sollte auch der betreffende Kartenausschnitt dargestellt werden, um eine genaue Zuordnung zu ermöglichen. Auf Bemerkungen sollte durch andere Bürger oder Redakteure geantwortet werden können, wodurch eine Diskussion zu einem Thema geführt werden kann und eine Klärung der Sachverhalte unterstützt wird.

Weiterhin wäre die Erweiterung der Bürgerbeteilgung zu einer Behördenbeteiligung denkbar. Für eine solche Beteiligungsplattform wäre die Authentifikation der Benutzer gegenüber dem System erforderlich, um die Bemerkungen eindeutig Benutzern zuordnen zu können.

Derzeit können Bemerkungen nur zu Punkten auf den Karten erstellt werden. Hier ist eine Erweiterung angedacht, in der die Bürger einen Bereich (in Form von Linien und polygonen) auf der Karte markieren und zu diesem ihre Bemerkung erstellen können.

Mögliche Erweiterungen betreffen auch die Benutzungsschnittstelle, um dem Benutzer die Interaktion mit den Karten zu erleichtern. Dazu gehören Möglichkeiten, um bestimmte Layer ausblenden, eigene Kartenkonfigurationen speichern zu können und zusätzliche Navigations-

108 KAPITEL 8 AUSBLICK

möglichkeiten (Gummiband, Ansicht zurücksetzen, bestimmte Detailansichten, usw.) der Karten anzubieten.

Einige dieser Erweiterungen bedürfen nur einer Änderung in der Darstellungsschicht. Dazu müssen weitere Funktionalitäten auf der Website implementiert werden, wie zum Beispiel die Erweiterung der Navigationsmöglichkeiten. Hierbei muss beachtet werden, dass die Website weiterhin ohne proprietäre Plugins realisiert wird. Daher sind die Erweiterungen auf die Möglichkeiten von HTML und JavaScript begrenzt. Andere Erweiterungen, wie zum Beispiel die Verwaltung der erstellten Bemerkungen, verlangen eine Anpassung des GIS-Integrators. Hier macht die Anpassung der Freischaltungsfunktionalität der Annotationen eine Erweiterung des Frontends und des Backends des GIS-Integrators notwendig. Um benutzerbezogene Bemerkungen erstellen zu können, sind Änderungen im Back- und Frontends des GIS-Integrators, der Client-API und der Website notwendig. Bei der Umsetzung solcher umfangreichen Erweiterungen, sind auch Veränderungen der Architektur des GIS-Integrators und der Client-API sinnvoll.

Weitere technische Änderungen könnten vor allem die zusätzliche Verwendung von Standards im Backend des GIS-Integrators und der Schnittstellen zu den beteiligten Systemen betreffen. Um den internen Zugriff auf die Daten der Konfigurationen, WebMaps, Annotationen und dergleichen zu erleichtern, wäre die Standardisierung der Zugriffsmethoden sinnvoll. Hier wäre der Einsatz der Java Content Repository API (JCR) denkbar, welche ein Schnittstelle definiert, die alle Datenelemente im System als Content verwaltet und einen einheitlichen Zugriff darauf bietet. Detaillierte Informationen sind dazu im Java Specification Request 170 [JSR 170] zu finden. Diese API kann auch in der Client-API Verwendung finden, womit sich die Implementation im GIS-Integrator und in der Client-API vereinheitlichen würde.

Die Schnittstellen des GIS-Integrators basieren derzeit auf Standardprotokollen wie HTTP und verwenden XML als Datenaustauschformat. Die XML-Strukturen sind derzeit allerdings proprietär implementiert. Hier wäre der Einsatz von, durch das W3C standardisierten, Web Services sinnvoll. Hervorzuheben ist unter diesem Gesichtspunkt die Verwendung des Simple Object Access Protocols (SOAP), um den Datentransfer per XML zu standardisieren.

Die Datenstrukturen der Annotationen (Bemerkungen) sind derzeit statisch im GIS-Integrator implementiert, diese sollten durch Konfigurationseinstellungen anpassbar sein, um Bemerkungen unterschiedlicher Art verwalten zu können.

Eine umfassende Ergänzung des GIS-Integrators wäre die Erweiterung um eine Mandantenfähigkeit. Dies bedeutet, dass ein Mandantenmanagement integriert werden müsste, welches eine Benutzerverwaltung, und diverse Konfigurationseinstellungen für jeden Mandanten beinhaltet. Dazu gehört unter anderem eine Zugriffskontrolle auf den GIS-Integrator, die individuelle Zuordnung und Erstellung von WebMaps und Erweiterungsoptionen für Schnittstellen zu anderen CMS.

[Alonso/Casati/Kuno/Machiraju2004] Gustavo Alonso, Fabio Casati, Harumi Kuno, Vijay Machiraju: Web Services. Concepts, Architectures and Applications. Springer Verlag 2004

[Baumann 2001] Sabine Baumann: Begriffsdefinitionen im Content Management, Webseite, http://www.contentmanager.de/magazin/artikel_59_begriffsdefinitionen_im_content_management.html, 2001

[Bauer 2005] Olaf Bauer: Integration verteilter Geodatendienste: Ein Internetportal für die Metropolregion Hamburg, Diplomarbeit, TU Hamburg Harburg, 2005

[Becker u.a. 1999] Prof. Dr. J. Becker, Prof. Dr. H. L. Grob, Prof. Dr. St. Klein, Prof. Dr. H. Kuchen, Prof. Dr. U. Müller-Funk, Prof. Dr. G. Vossen: Arbeitsberichte des Instituts für Wirtschaftsinformatik – Integrationsmanagement, Universität Münster, 1999

[Beuter Dadam 1996] Thomas Beuter, Peter Dadam: Prinzipien der Replikationskontrolle in verteilten Datenbanksystemen, aus Informatik, Forschung Entwicklung 11(4): S. 203-212, Springer 1996

[Beaujardiere 2004] Jeff de L. Beaujardiere: Web map server interface ISO/DIS 19128 (OGC WMS 1.3.0). OGC (Organisation), 2004

[Bezugssysteme] Bezugssysteme für die Lagebeschreibung von Punkten, Webseite: http://www.vermessungsseiten.de/vermessungstechniker/bezsyst.htm

[CMS Evaluation] Content Management System Evaluation http://www.atnf.csiro.au/computing/web/cms_eval.html

[CoreMedia 2005] Homepage: CoreMedia AG, Hamburg, http://www.coremedia.de

[Dadam 1996] Peter Dadam: Verteilte Datenbanken und Client/Server Systeme – Grundlagen, Konzepte, Realisierungsstrukturen, Springer Verlag 1996

[DavidS.Linthicum2003] David S. Linthicum: Next Generation Application Integration, Addison Wesley 2003

[deegree 2005] Homepage: deegree, http://deegree.sourceforge.org

[de Lange 2002] Norbert de Lange: Geoinformatik in Theorie und Praxis. Springer-Verlag, 2002

[E-Government-Handbuch] Bundesamt für Sicherheit in der Informationstechnik, http://www.bsi.bund.de/fachthem/egov/3.htm

[ESRI] Homepage: ESRI, http://www.esri.com

[Fraunhofer MediaSystems 2004] Gutachten "Regionen und Branchen im Wandel", 2004

[Geonord 2005] Freie und Hansestadt Hamburg Landesbetrieb Geoinformation und Vermessung, http://www.geonord.de

[GIS Ingenieur] Homepage: http://www.gis-ingenieur.de/

[GSB 2005] Homepage: Government Site Builder, http://www.government-site-builder.de.

[Hamburg.de 2005] Homepage: hamburg.de, http://www.hamburg.de

[Holler 2003] Reinhard Holler, Konzept und Einsatz eines Content-Management-Systems am Beispiel der WWW-Präsenz der FH-Augsburg, Diplomarbeit, 2003

[Intergraph 2005] Homepage: Intergraph, http://www.intergraph.com

[Internet Lexikon 2005] Homepage: http://www.ilexikon.com

[JSR 170] Java Specification Request 170: Content Repository for JavaTM technology API, http://www.jcp.org/en/jsr/detail?id=170

[K.-U. Krause 2005] Dr. Ing. Kai-Uwe: Kommunale Informationssysteme, http://www.tu-harburg.de/stadtplanung/html/ab/ab 105/ag 1/personen/krause.htm

[Krause u. a. 2003] Kai-Uwe Krause; Olaf Bauer, Niels Hoffmann: Verteilte Geo- und Planungsdienste für das Portal www.metropolregion.hamburg.de mit Hilfe der CoreMedia Content Application Platform. In: [Strobl und Griesebner, 2003], S. 67-75 [latlon 2005] Homepage: latlon, Bonn, http://www.latlon.de/

[Lenz 97] Richard Lenz.: Adaptive Datenreplikation in verteilten Systemen. Teubner Verlag, Leipzig 1997.

[Lüpsen] Haiko Lüpsen: Webseite, http://www.uni-koeln.de/rrzk/kompass/91/k9110.html

[Mantel u.a. 2000] Stephan Mantel, Bernd Knobloch, Thorsten Rüffer, Martin Schissler, Klaus Schmitz, Otto K. Ferstl, Elmar J. Sinz, Analyse der Integrationspotenziale von Kommunikationsplattformen für verteilte Anwendungssysteme, Universität Bamberg, 2000

[Materna 2005] Homepage: Materna, http://www.materna.de

[MKT 2005] Homepage Bundesministerium für Wirtschaft und Arbeit: http://www.mediakomm-transfer.de/

[Meyers Lexikon 1990] Meyers Großes Taschenlexikon, Meyers Verlag, 1990

[MRH 2005] Homepage: Metropolregion Hamburg, http://www.metropolregion. hamburg.de

[MRH Machbarkeit 2003] Machbarkeitsstudie zum E-Government in der Metropolregion Hamburg http://www.komfit.de/Projekte/02-229 Info E-Government Metropolregion.pdf, 2003

[OGC 2005] Homepage: Open Geospatial Consortium, http://www.opengeospatial.org

[Orcale 2005] Homepage: Oracle Corp., http://www.oracle.com

[Plone] Plone - OpenSource Content Management System, http://plone.org/

[REK 2000] Regionales Entwicklungskonzept 2000: Leitbild und Handlungsrahmen, Metropolregion Hamburg. Hamburg, Hannover, Kiel: Gemeinsame Landesplanung Hamburg/Niedersachsen/Schleswig-Holstein, 2000

[rothfuß/ried2001] Gunther Rothfuß, Christian Ried (Hg.): Content Management mit XML. Grundlagen und Anwendungen. Springer Verlag, 2001

[Schlotz] Uwe Schlotz: GIS als Werkzeug zur Effektiven Erstellung und Pflege eines Pavement-Management-Systems für Kommunale Anwender, http://www.big-geoinfo.de/PMSBeitraege/S7_036_gl.pdf

[Schuster 2003] Grit Schuster: Entwicklung von Komponenten eines Geoportal-Frameworks, Diplomarbeit, Hochschule Vechta, 2003 [seevetal 2005] Homepage: Seevetal, http://www.seevetal.de

[Spilker 2004] Daniel Spilker, Content Unification am Beispiel von CoreMedia SCI und Java Content Repository, Diplomarbeit TU Hamburg Harburg 2004 [Stadtplanung Seevetal 2005] Homepage http://seevetal.sts.tu-harburg.de

[stefan conrad1997] Stefan Conrad: Föderierte Datenbanksysteme. Springer Verlag 1997

[Strobl/Griesebner 2003] Josef Strobl, Gerald Griesebner: geoGovernment: Öffentliche Geoinformationsdienste zwischen Kommune und Europa. Wichmann Verlag, 2003

[STS 2005] Homepage http://www.sts.tu-harburg.de

[Sybase 2000] Dokumentation: Replikation und Synchronisierung, Sybase 2001

[Thorsten Reincke 1999] Thorsten Reincke, Dynamische Replikationsverfahren für den mobilen Datenbankeinsatz, Diplomarbeit Universität Rostock 1999

[Tomcat 2005] Homepage: Apache Tomcat, http://jakarta.apache.org.

[Typo3] Typo3 - Open Source Content Management Framework: http://typo3.org/

[Ullman 97] Jeffrey D. Ullman, Information Integration - Using Logical Views, Stanford University 1997.

[UMN Mapserver 2005] Homepage, University of Minnesota, http://mapserver.gis.umn.edu/

[VuGIS 2003] Guido Gerding, Eva Klien, Michael Lutz, Jörn Möltgen, Joachim Scheiner, Carsten Schürmann, Björn Schwarze, Klaus Spiekermann: VuGIS Verkehrs- und Geoinformationssystem, Dortmund und Münster, 2003

[Web Services] Web Service Activity des W3C, http://www.w3.org/2002/ws/

[Westerfeld 2000] Markus Westerfeld: Ein middlewarebasierter, generativer Ansatz zur Unterstützung von Schemaevolution in komplexen Ingenieurinformationssystemen, Diplomarbeit, Universität-Gesamthochschule Paderborn

[Ziegler 2002] Christian Ziegler: Entwicklung eines Content Management Systems für die MIO-Lernumgebung, Diplomarbeit, Institut für Informatik der Universität Zürich 2002

[Zope] Zope - OpenSource Application Server: http://www.zope.org