

Mark Hampe

**Integration einer multiskaligen Datenbank
in eine Webservice-Architektur**

München 2007

**Verlag der Bayerischen Akademie der Wissenschaften
in Kommission beim Verlag C. H. Beck**

ISSN 0065-5325

ISBN 3 7696 5044 1

**Diese Arbeit ist gleichzeitig veröffentlicht in:
Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Leibniz Universität Hannover
ISSN 0174-1454, Nr. 266, Hannover 2007**

Integration einer multiskaligen Datenbank
in eine Webservice-Architektur

Von der Fakultät für Bauingenieurwesen und Geodäsie
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des Grades
Doktor-Ingenieur (Dr.-Ing.)
genehmigte Dissertation

von

Dipl.-Ing. Mark Hampe

München 2007

Verlag der Bayerischen Akademie der Wissenschaften
in Kommission bei der C. H. Beck'schen Verlagsbuchhandlung München

Adresse der Deutschen Geodätischen Kommission:

Deutsche Geodätische Kommission

Alfons-Goppel-Straße 11 • D – 80 539 München

Telefon +49 - (0)89 - 23 031 -0 / -1113 • Telefax +49 - (0)89 - 23 031 -1283 / -1100

E-mail hornik@dgfi.badw.de • <http://dgk.badw.de>

Prüfungskommission

Referent: Prof. Dr.-Ing. Monika Sester

Korreferenten: Prof. Dr.-Ing. Christian Heipke

Prof. Dr.-Ing. Liqiu Meng

Tag der mündlichen Prüfung: 18.05.2007

© 2007 Deutsche Geodätische Kommission, München

Alle Rechte vorbehalten. Ohne Genehmigung der Herausgeber ist es auch nicht gestattet,
die Veröffentlichung oder Teile daraus auf photomechanischem Wege (Photokopie, Mikrokopie) zu vervielfältigen

ISSN 0065-5325

ISBN 3 7696 5044 1

Zusammenfassung

Viele Phänomene unserer realen Umgebung werden mehrfach erfasst und liegen isoliert voneinander in den Datenbanken bereit. Dabei vollzieht sich die Informationserfassung und -haltung oftmals nicht redundant, sondern es liegt vielmehr eine alternative Sichtweise auf das Phänomen zugrunde. In vielen Fällen bestehen unterschiedliche Anforderungen an die beschreibenden Daten, was in einer multirepräsentativen Datenhaltung resultiert: Es existieren Mehrfach-Beschreibungen zu einem bestimmten Objekt der realen Welt. Dabei wird hier weniger das Problem der Mehrfacherfassung auf Grund von Unwissenheit über die Existenz der Daten betrachtet, als vielmehr die Multiplizität infolge vielfältiger Anforderungen. Geoinformationen desselben Teiles der Erdoberfläche werden beispielsweise in unterschiedlicher Auflösung erfasst und gepflegt und dienen unter anderem als Grundlage für Karten verschiedener Maßstäbe. Werden unterschiedliche Beschreibungen desselben Phänomens parallel genutzt, so ist es sinnvoll, die Informationen nicht nebeneinander vorzuhalten, sondern in ein konsistentes System zu integrieren. Eine MRDB beschreibt ein derartiges System. Diese fasst multirepräsentative Daten in einer konsistenten Datenbasis zusammen, mit dem Ziel der effektiven Datenpflege und -nutzung. Eine konsequente Weiterentwicklung dieser Idee besteht nun darin, die Vorteile integrierter, multirepräsentativer Daten nicht nur für interne Zwecke zu nutzen, sondern dieses Prinzip auf die vielfältigen, über das Internet verfügbaren Informations- und Datenquellen auszuweiten. Durch die Integration verschiedenster Datenquellen in das „spatially enabled web“ besteht derzeit eine große und vielfältige Agglomeration auch an Geoinformationen, welche für jeden Internetnutzer verfügbar sind. Möchte dieser nun mehrere zu einem Phänomen erfasste Daten parallel nutzen, so stellt sich auch hier das Problem der Inkonsistenz der unterschiedlichen Ressourcen. Somit besteht auch an dieser Stelle der Bedarf, die monorepräsentativen Daten in ein multirepräsentatives System zu integrieren, um die Informationen zu bündeln und widerspruchsfreie Analyseergebnisse zu erhalten.

Auf dem Weg dorthin gilt es, zunächst die Besonderheiten sowie Probleme aufzudecken, welche diese Konstellation mit sich bringt. Die monorepräsentativen Daten müssen zu einem multirepräsentativen System kombiniert und dem Internetnutzer verfügbar gemacht werden. Auf der einen Seite entstehen hierdurch neue Anforderungen an die MRDB, welche nicht nur die Konsistenz der integrierten Daten sicherstellen, sondern darüber hinaus dem Anwender als multirepräsentative Informationsbasis dienen soll. Diese sollte durch eine entsprechende Modellierung dem Zugriff aus einer Web-Umgebung heraus angepasst sein. Auf der anderen Seite gilt es, die Webservice-Architektur so zu adaptieren, dass diese die Möglichkeiten eines multirepräsentativen Systems unterstützt. Aus einer szenarienbasierten Entwicklung heraus werden die Anforderungen an ein derartiges System abgeleitet. Dabei stehen vor allen Dingen die Ansprüche des Anwenders an eine Geo-Informationsbasis im Fokus der Betrachtungen. Hieraus wird der Bedarf abgeleitet, dem internetgebundenen Nutzer nicht nur den Zugriff auf eine bestimmte Informationsmenge zu gewährleisten, sondern zusätzlich sicherzustellen, dass diese Informationen in einem konsistenten System integriert sind. Neben der Konsistenz ist es auch wichtig, durch entsprechende Datenstrukturen den Zugriff auf alle zu einem Phänomen verfügbaren Informationen zu gewährleisten.

Eine prototypische Umsetzung eines derartigen Systems, welches sich an der gängigen Webservice-Architektur orientiert und sowohl die multirepräsentative Informationsmanagement-Komponente als auch die Applikationsebene und schließlich eine Nutzerapplikation implementiert, validiert die beschriebenen Anforderungen und Lösungen. Der Datenbankentwurf berücksichtigt die Anforderungen eines multirepräsentativen Designs sowie zusätzlich die Ansprüche einer Webumgebung und schließlich die Bedürfnisse des Nutzers. Neben dem konzeptionellen Entwurf wird ein Weg zum Aufbau einer multiskaligen Datenbank aufgezeigt, welche durch Generalisierung abgeleitete und integrierte Geodaten verschiedener Auflösungsstufen bereithält. Die Entwicklung der Informationsmanagementkomponente zeigt, dass bestehende Web-Komponenten zwar einen Zugriff auf eine MRDB erlauben, dagegen nicht in der Lage sind, die Anforderungen und Vorteile einer derartigen Informationsbasis vollständig auszuschöpfen. Diese berücksichtigen nicht die multirepräsentative Charakteristik der Datenbasis. Daher wird eine bestehende Schnittstellen-Spezifikation des OGC für den Zugriff auf multiskalige Geodaten entsprechend erweitert. Ein exemplarisch umgesetzter Mapservice, welcher auf diese multiskalige Datenbasis zurückgreift und eine auf SVG-Basis realisierte mobile Applikation belegen schließlich die Anwendbarkeit des Systems und demonstrieren die Vorteile einer derartigen Informationsbasis. Der mit dem Internet verbundene, mobile Client greift mit Hilfe dieses Mapservices über die neu definierte Schnittstelle auf die multirepräsentativen Daten zurück und erhält somit Zugriff auf alle zu einem Phänomen verfügbaren Informationen. Auf diese Weise kann unter anderem die für die aktuelle Situation relevante, geometrische Auflösung gewählt werden. Diese Möglichkeit wird dazu verwendet, für mobile Geräte adaptierte Visualisierungen der Geoinformationen zu generieren. Die

unterschiedlichen Repräsentationen können derart kombiniert werden, dass überflüssige Informationen, respektive geometrische Details, an Stellen weggelassen werden, an denen diese nicht benötigt werden. Auf diese Weise entstehen multiskalige Darstellungen, welche die Limitierungen der kleinen Bildschirme mobiler Geräte revidieren. Zusätzlich hat der Anwender jederzeit Zugriff auf höher aufgelöste Informationen oder alternative Sichten auf bestimmte Objekte. Auch der nicht mobile Nutzer eines GIS kann eine multirepräsentative Datenbasis beispielsweise für Analysezwecke nutzen, wie der Prototyp demonstriert.

Schlagwörter: *Multiskalige Datenbanken, MRDB, Geo-Webservices, Geodatenvisualisierung, Generalisierung*

Abstract

Many phenomena of our environment have been recorded multiply but are maintained separately and isolated from each other. Each of the data sets maintains a particular representation of the real world, custom-tailored by the demand and perception of the associated applications. For example geodata of the same region are collected and maintained with different resolutions and serve as a source for maps with different scales. While the real world is supposed to be unique, its representation depends on the intended purpose. Building a global view results in the coexistence of several heterogeneous representations of the environment. Data-collection efforts are compromised by the difficulty in reconciling and integrating information from the different existing heterogeneous data sets. To avail oneself of the different descriptions of the same real world phenomena, it is necessary to integrate these data into a consistent database system. A multiple representation database (MRDB) integrates the heterogeneous data into a homogeneous, consistent database schema to achieve an effective maintenance and use of the data.

The webservice-technologies and the introduction of generally accepted geo-webservice specifications offer the possibility to access remote data sources. By generating the „spatially enabled web“ an immense congeries of geodata becomes available for the user.

If the user wants to use the data sources synonymously these heterogeneous information also have to be integrated into a consistent environment. It is important to provide not only access for a certain amount of information but also to integrate these information into a homogeneous system and to offer a possibility to obtain all information related to a certain real world phenomenon. That means, the techniques of setting up multiple representation databases are also needed for webbased data sources. This would be the next logical step in the use of the MRDB concept. Besides the integration process further obstacles occur when an MRDB is implanted into a webservice environment. Which additional requirements need to be considered when designing the database schema? Which functionalities have to be integrated into the access interface? And finally, which possibilities will emerge when the user can access multiple representations and views related to a certain phenomenon?

This thesis deduces the requirements of such a system serving multiple representations from a scenario-based development. A scenario-based design ranks the user first and then the technical requirements. The design is put into practice by implementing a prototype system. The prototype reveals technical problems and demonstrates the usability of the system. This system is geared to the common webservice architecture as well as existing OGC-specifications of geo-webservices. It implements the information management component, consisting of an MRDB and an access interface, as well as an application layer providing mapservice functionalities and finally the client layer with an application for a mobile client as well as a non mobile GIS client.

The database design considers the requirements of multirepresentational data, the demands of a webservice environment as well as the needs of a (mobile) user accessing the database via the internet. A conceptual database design is presented as well as a way to set up a multiscale database by integrating generalised geodata. The integration is directly derived from the process of generalisation. At the same time the webservice interface has to be adapted for the requirement of serving multiple representations associated with a certain object. Existing geo-webservices allow to access and provide the data of an MRDB but they are not able to tap the requirements and the full potential of a multiple representation database. They do not support the multiplicity of the objects. Therefor an existing OGC-webservice specification will be extended to be used for accessing multiple-resolution geodata over the internet.

A prototypical mapservice, which accesses the MRDB via the webservice interface, as well as a mobile application highlight the advantages and new possibilities when falling back on an integrated, multirepresentational database. The client has access for all information related to a certain phenomenon. Furthermore the user can select the data resolution, adequate for the actual task or device. This possibility can be used to adapt the visualisation for example for mobile devices. Applications must be designed carefully to meet the diverse needs of the users. The basic design restriction of small displays is their poor resolution, compared to desktop computer displays or paper maps. This places new demands on the cartographic visualisation. With a multiple resolution database the visualisation can be adapted by choosing a certain representation containing a certain amount of information, or rather geometric details. Furthermore the user has always the possibility to request high-resolution information as well, if needed. A non mobile GIS-user can combine the multiple representations for analysis functions as every

representation contains individual information. Furthermore latent information can be discovered by comparing the alternative datasets.

Keywords: *Multiscale databases, MRDB, geo-webservices, geodata visualisation, generalisation*

Inhaltsverzeichnis

Zusammenfassung	3
Abstract	5
Inhaltsverzeichnis	7
Abkürzungen	11
1 Einleitung	13
1.1 Motivation	13
1.2 Zielsetzung der Arbeit	15
1.3 Aufbau der Arbeit	16
2 Grundlagen: Webdienste und Geo-Webdienste	17
2.1 Webservices - Eine Definition	17
2.2 Architektur eines Webdienstes	18
2.2.1 Client-Server-Architektur	18
2.2.2 Die Web Service Architecture (WSA) des W3C	20
2.2.3 Das Semantische Web	21
2.3 Geo-Webdienste	22
2.3.1 Standards für Geo-Webdienste	23
2.3.2 Die OGC Web Services Architektur	26
2.3.3 Infrastrukturen und Anwendungsbereiche von Geo-Webdiensten	30
2.4 Mobile Webdienste	36
2.4.1 Mobile Geräte	38
2.4.2 Anbindung mobiler Geräte - Drahtlose Netzwerke	40
2.4.3 Ortsbezogene Dienste	43
2.5 Adaption der Inhalte	44
2.6 Datenbanktechnische Aspekte eines Webservices	46
2.6.1 Datenbankinhalte	48
2.6.2 Qualität und Aktualisierung	49
2.7 Zusammenfassung	50

3 Grundlagen: MRDB	53
3.1 Definition und Probleme	53
3.2 Vorausgegangene Arbeiten auf dem Gebiet der MRDB	55
3.2.1 Eine Forschungsinitiative des NCGIA	55
3.2.2 Das EU-Projekt MurMur	57
3.3 Nutzen einer MRDB	58
3.3.1 Darstellung von Geodaten in unterschiedlichen Maßstäben	58
3.3.2 Webmapping	58
3.3.3 Automatische Aktualisierung von Geodaten	59
3.3.4 Analyseverfahren	61
3.4 Mögliche Wege zum integrierten, multirepräsentativen Datenmodell	61
3.4.1 Schritte in Richtung eines integrierten Systems	62
3.4.2 Möglichkeiten der Modellierung	64
3.4.3 Grundsätzliche Modellierung verschiedener Verknüpfungskardinalitäten	69
3.4.4 Beispiele für Integrationsmodelle	69
3.4.5 Matching der Instanzen	71
3.5 MRDB Standardoperationen und Nutzerinterfaces	75
3.6 Zusammenfassung und Schlussfolgerungen	79
4 MRDB-Webdienst: Definition, Probleme und Lösungen	81
4.1 Szenarienbasierte Systementwicklung	82
4.1.1 Bedarfsanalyse	83
4.2 Systemarchitektur	88
5 MRDB-Webdienst: Aufbau einer multiskaligen Datenbasis	91
5.1 Anforderungen und Charakteristika eines DBMS	91
5.2 Datenbankkonzept einer MRDB	92
5.2.1 Festlegung der konzeptionellen Auflösungsebenen	92
5.2.2 Beschreibung der Testdatensätze	93
5.2.3 Datenbankentwurf	97
5.2.4 Metadatentabelle	109
5.3 Datengenerierung und -integration	110
5.3.1 Die Arbeitsumgebung: Java, JTS, JUMP	110
5.3.2 Generalisierung der Datensätze	112
5.3.3 Verknüpfung vorhandener Datensätze durch Matching	119
6 MRDB-Webdienst: Funktionalitäten und Schnittstellendefinition	123
6.1 Erforderliche Funktionalitäten eines MRDB-Services	123
6.2 Probleme bei Verwendung des OGC-WFS in einer multirepräsentativen Umgebung	124
6.3 Erweiterung des OGC-WFS zur Abfrage multirepräsentativer Daten	125
6.3.1 Architektur des MRDB-Service	125
6.3.2 Funktionen und Implementierung eines MRDB-Service	126

7 MRDB-Webdienst: Applikations-Ebene	135
7.1 Systemarchitektur der Applikations-Ebene	135
7.2 Ein Webmapservice für mobile Clients	135
7.2.1 Architektur eines Webmapservice	135
7.2.2 SVG und XSLT	137
7.2.3 Funktionalitäten	137
7.3 MRDB-Service im Vergleich zur Online-Generalisierung	143
7.3.1 Test Gebäudevereinfachung	144
7.3.2 Test Gebäudezusammenfassung	144
7.3.3 Test Typifizierung und Verdrängung	144
7.3.4 Beurteilung der Testergebnisse	145
8 MRDB-Webdienst: Client-Ebene	149
8.1 Eine Applikation für mobile Nutzer	149
8.1.1 Funktionalitäten	149
8.2 Desktopgebundener GIS-Client	153
9 Ergebnisse und Ausblick	157
9.1 Ergebnisse der Arbeit	157
9.2 Diskussion und Ausblick	158
Literaturverzeichnis	161
Danksagung	169
Wissenschaftlicher Werdegang	171

Abkürzungsverzeichnis

ADT	Abstract Data Type	IMAGI	Interministerieller Ausschuss für Geoinformationen
ALK	Automatisierte Liegenschaftskarte	INSPIRE	Infrastructure for Spatial Information in Europe
AMPS	Advanced Mobile Phone Service	ISO	International Organization for Standardization
API	Application Programming Interface	IT	Informationstechnologie
ATKIS	Amtlich Topographisch-Kartographisches Informationssystem	IuK	Information und Kommunikation
BITKOM	Bundesverband Informationswirtschaft, Telekommunikation und neue Medien	J2ME	Java2 Micro Edition
BLG	Binary line generalisation tree	JCS	JCS Conflation Suite
CAD	Computer Aided Design	JDBC	Java Database Connectivity
CDMA	Code Division Multiple Access	JEE	Java Platform, Enterprise Edition
CEO	Chief Executive Officer	JPEG	Joint Photographic Expert Group Format
DBMS	Datenbank Managementsystem	JTS	JTS Topology Suite
DBS	Datenbanksystem	JUMP	JUMP Unified Mapping Platform
DDGK5	Digitale Deutsche Grundkarte 1:5.000	Kbps	Kilobits per second
DDL	Data Definition Language	LAN	Local Area Network
DHDN	Deutsches Hauptdreiecksnetz	LBS	Location based services
DLM	Digitales Landschaftsmodell	LoD	Level of detail
DML	Data Manipulation Language	MADS	Modeling of Application Data with Spatio-temporal features
DOM	Document Object Model	ME	Micro Edition
DSK10	Digitale Straßenkarte 1:10.000	MIME	Multipurpose Internet Mail Extensions
DSL	Digital Subscriber Line	MMS	Multimedia Messaging Service
DTK	Digitale Topographische Karte	MRDB	Multiple Representation Database
ER-Modell	Entity Relationship-Modell	MR-WFS	Multiple Representation Web Feature Service
FDGC	US Federal Geographic Data Committee	MurMur	Multiple representations multiple resolutions
FTP	File Transfer Protocol	NCGIA	National Center for Geographic Information
GDI	Geodateninfrastruktur	NIAM	Nijssen's Information Analysis Method
GIF	Graphics Interchange Format	NMEA	National Marine Electronics Association
GiMoDig	Geospatial info-mobility service by realtime data-integration and generalisation	NSDI	National Spatial Data Infrastructure
GIS	Geoinformationssystem	OGC	Open Geospatial Consortium
GLONASS	Globalnaya Navigatsionnaya Sputnikovaya Sistema	OK	Objektartenkatalog
GML	Geography Markup Language	OLAP	Online Analytical Processing
GOS	Geospatial One-Stop	OLED	Organic Light-Emitting Diode
GPRS	General Packet Radio Service	OMT	Object Modeling Technique
GPS	Global Positioning System	ONE	Open Network Environment
GSM	Global System for Mobile Communications	OWL	Web Ontology Language
GUI	Graphical User Interface	OWS	OGC Web Services
HTML	Hypertext Transfer Markup Language	PDA	Personal Digital Assistant
HTTP	Hypertext Transfer Protocol	PHP	PHP Hypertext Preprocessor
IEEE	Institute of Electrical and Electronics Engineers	PNG	Portable Network Graphics
IGN	Institut Geographique National	PR-File	Priority Rectangle File
		RDF	Resource Description Framework

RPC	Remote Procedure Call
SDBMS	Spatial Database Managementsystem
SDI	Spatial Data Infrastructure
SIM	Subscriber Identification Module
SMS	Short Message Service
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SVG	Scalable Vector Graphics
TC	Technical Commission
TFT	Thin Film Transistor
TIFF	Tagged Image File Format
UDDI	Universal Description, Discovery and Integration
UML	Unified Modeling Language
UMTS	Universal Mobile Telecommunication System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
USGS	United States Geological Survey
VGA	Video Graphics Array
W3C	World Wide Web Consortium
WAP	Wireless Application Protocol
WCS	Web Coverage Services Implementation Specification
WFS	Web Feature Services Implementation Specification
WiFi	Wireless Fidelity
WiMax	Worldwide Interoperability for Microwave Access
WIPKA	Wissensbasierter photogrammetrisch-kartographischer Arbeitsplatz
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WML	Wireless Markup Language
WMS	Web Map Services Implementation Specification
WPAN	Wireless Personal Area Network
WPS	Web Processing Service
WSA	Web Service Architektur
WSDL	Web Services Description Language
WWW	World Wide Web
XML	Extensible Markup Language
XSD	XML Schema Definition
XSL	Extensible Stylesheet Language
XSLT	Extensible Stylesheet Language Transformation

1 Einleitung

1.1 Motivation

„Alle vielversprechenden Technologien der heutigen Zeit basieren auf der Kommunikation zwischen Computern [...] und da Kommunikation die Basis aller menschlichen Kulturen ist, liegt hier ein enormes Potenzial. Jeder Schritt in Richtung effizienter, universeller Kommunikationsfähigkeit von Informationssystemen ist ein Schritt in die richtige Richtung“ Kevin Kelly, Executive Director, Wired Magazine¹.

Informationen sind häufig nicht einfach vorhanden, vielmehr wird ein Phänomen der realen Welt oftmals mehrfach erfasst und in unterschiedlichen Versionen beschrieben. Die Gründe hierfür sind sehr vielschichtig und resultieren beispielsweise aus den unterschiedlichen Anforderungen der Nutzer an die Daten, der zeitlichen Variation der Informationen oder aber einfach aus der Unwissenheit über die alternativen Quellen. Auch viele verwandte Geodatenätze stammen aus unterschiedlichen Datenquellen verschiedenster Datenanbieter, repräsentieren jedoch dasselbe räumliche Gebiet und unterscheiden sich lediglich in der Form der Beschreibung dieser räumlichen Informationen.

In zahlreichen Anwendungen und Bereichen ist dieses Problem der multiplen Datenerfassung und -haltung bekannt, wie beispielsweise bei den Landesvermessungsämtern und anderen Kartenproduzenten. In der Kartografie werden unter anderem Daten unterschiedlicher geometrischer und semantischer Auflösung und Inhalte vorgehalten, um hieraus kartografische Produkte diverser Maßstäbe und verschiedener Thematiken ableiten zu können. In der Kartenproduktion werden daher unterschiedlich aufgelöste Datenmodelle getrennt voneinander vorgehalten und gepflegt.

Topografische Informationen sowie statistische Phänomene mit einer räumlichen Komponente weisen eine Maßstabsabhängigkeit auf und erschweren hierdurch die Automatisierung kartografischer Prozesse sowie die Analyse geografischer Muster, wenn die Daten getrennt voneinander vorgehalten werden. Der Aufwand in der Erfassung und Pflege singulärer, unabhängiger Datenbankinformationen für jeden gewünschten Präsentationsmaßstab limitiert unter anderem die Verlässlichkeit vieler GIS²-Applikationen inklusive den auf räumlichen Informationen basierenden Entscheidungsprozessen.

Eine MRDB³ stellt hier einen Mechanismus bereit, heterogene Datensätze in ein homogenes System zu integrieren. Diese vereint multiple Repräsentationen in einem konsistenten System, so dass die Informationen nicht mehr nebeneinander existieren, sondern vielmehr eine in sich widerspruchsfreie Datenbasis bilden. Durch eine entsprechende Datenmodellierung ist die Konsistenz des Systems sichergestellt trotz einer multiplen Instanzierung der enthaltenen Objekte. Eine MRDB beschreibt die Zusammenhänge zwischen den einzelnen Repräsentationen und ermöglicht somit beispielsweise eine Datenerfassung und -haltung, welche unterschiedliche Auflösungsstufen nicht mehr separiert, sondern in einem System beschreibt. Somit wird es zum Beispiel möglich, durch Aktualisierung hervorgerufene Veränderungen eines Objektes auf die übrigen Repräsentationen automatisch zu übertragen. Multirepräsentative Datenstrukturen helfen weiterhin bei der Datenanalyse, da verschiedenste Informationen zu einem bestimmten Phänomen, entsprechend der Sicht des Erfassers, verfügbar sind.

Parallel hierzu ist eine rapide Entwicklung der IuK⁴-Technologie in unterschiedlichen Bereichen zu beobachten. Durch die Vernetzung aller an das Internet angeschlossenen Computer steht auf diese Weise, theoretisch und unter gewissen Restriktionen, allen Nutzern der Zugriff auf eine breite Informationsplattform zur Verfügung. Gleichzeitig nutzen die Datenanbieter diese Möglichkeit, um ihre Daten diversen Nutzern zu offerieren. Auf Grund der hohen Akzeptanz des Internets versuchen zahlreiche Unternehmen, Daten und Software in Form von Informationen und Dienstleistungen nicht nur für die internen Prozesse einzusetzen, sondern auch geschäftlichen und privaten Kunden über das Internet anzubieten. Positive Wachstumsraten des E-Commerce und E-Business sind Ausdruck dieser Veränderungen. Es ist ein Trend zu beobachten, weg von voneinander abgeschotteten, rein

¹<http://www.yellowmap.com/technologie/MobileTechnologies.asp>

²Geoinformationssystem

³Multiple Representation Database

⁴Information und Kommunikation

proprietären Systemarchitekturen hin zu offenen Schnittstellen. Das OGC⁵ hat in diese Richtung zahlreiche Bemühungen zum Aufbau offener Systeme in dem Bereich der geografischen Informationsverarbeitung gezeigt. Der Nutzer hat über standardisierte Schnittstellen Zugriff auf eine große Menge an Datenquellen. Gleichzeitig haben sich zahlreiche Portale etabliert, welche themenverwandte Geodatenquellen bündeln und zusätzliche Metadaten bereitstellen.

Durch die Verfügbarkeit dieser enormen Menge an Informationen wird es somit zunehmend wichtig, diese strukturiert vorzuhalten, und einen effektiven Zugriff zu gewährleisten. Es gilt, ein interoperables Netzwerk zu schaffen, welches möglichst vielen Anwendern über das Internet den Zugriff auf eine konsistente Datenbasis erlaubt. Interoperabilität zielt auf die Entwicklung von Mechanismen, um Inkompatibilitäten und Heterogenitäten zu lösen und den Datenaustausch sowie deren Wiederverwendung sicherzustellen. Daher ist es sinnvoll, verschiedenste Datenquellen zu kombinieren, wichtige Informationen zu extrahieren und diese in ein einheitliches System zu integrieren, um die spezifischen Anforderungen verschiedenster Nutzer erfüllen zu können. Auch das wachsende Bedürfnis der Datenanbieter, Datenduplikationen zu eliminieren und den zeitaufwändigen und teuren Prozess der Datenaquisition zu minimieren sowie auf der anderen Seite die voranschreitende Verfügbarkeit digitaler Daten haben zunehmend den Bedarf geweckt, Daten zu teilen und wiederzuverwenden. Der wachsende Austausch von Daten über Netzwerke wie Internet oder Intranet bedarf der Integration der Datensätze aus den verschiedensten Quellen. Auf der anderen Seite benötigt ein Datenanbieter eine möglichst umfassende Informationsbasis, um den Ansprüchen vieler Nutzer gerecht zu werden. Dies wiederum bedingt die Unterhaltung alternativer Beschreibungen spezieller Phänomene. So sollte ein Kartendienst nicht nur eine hochauflösende Kartenbasis bereithalten, sondern möglichst für jede Anwendung und jeden Nutzer entsprechend aufbereitete Geodaten zur Verfügung stellen.

Somit entsteht auch in diesem Umfeld der Bedarf, die multiplen Repräsentationen nicht mehr nebeneinander vorzuhalten, sondern eine multirepräsentative Datenbasis aufzubauen. Eine konsequente Weiterentwicklung der Idee multirepräsentativer Datenhaltung ist somit die Integration dieser in eine Web-Umgebung. Auf diese Weise werden die Vorteile einer MRDB, welche zu einem bestimmten Phänomen verschiedene Repräsentationen und Informationen in einem in sich konsistenten, widerspruchsfreien System vorhält, mit den Vorzügen einer offenen, webbasierten Lösung kombiniert.

Für den Anwender steht auf diese Weise ein umfassender Informationspool bereit. Dieser greift idealerweise lediglich auf *ein* widerspruchsfreies System zu, welches alle zu einem bestimmten Phänomen verfügbaren Informationen vereint. Somit sind umfassende Analysen möglich, welche alle Informationen in den Entscheidungsprozess einbeziehen können und nicht nur, wie in dem monorepräsentativen Fall, einen ausgewählten Datensatz, welcher nur *eine* mögliche Sichtweise auf das Phänomen zulässt oder aber durch Kombination inkonsistenter Datensätze fehlerhafte Interpretationsergebnisse liefert.

Des Weiteren führt die wachsende Bedeutung von Mobilität und Flexibilität zu einer erhöhten Nachfrage an mobilen Informations- und Kommunikationsdiensten, die es erlauben, trotz der räumlichen Distanz in bestimmten Situationen schnell und flexibel auf Informationen zuzugreifen, diese zu verarbeiten und auszutauschen. Diese Entwicklung wird durch die fortschreitende Verbesserung der Mobilfunknetze, wie die derzeitige dritte Generation (3G) der Mobilfunktechnologie in Form des UMTS⁶, sowie auf der anderen Seite durch die ständige Weiterentwicklung und Verbreitung mobiler Geräte vorangetrieben. Dabei werden die Nutzungsmöglichkeiten des Internets nicht einfach von dem desktopgebundenen Computer auf das mobile Gerät transferiert. Es entstehen vielmehr völlig neue Anwendungsfelder, da der mobile Nutzer andere Ansprüche und auch Voraussetzungen mitbringt. Mobile Anwendungen berücksichtigen nicht nur das persönliche Profil des Anwenders, sondern beziehen zusätzlich den aktuellen Kontext in das Serviceangebot ein. Vor allen Dingen die derzeitige Position des Nutzers hilft einem Geodaten- oder Kartendienst, die Informationen entsprechend aufzubereiten. Der mobile Nutzer benötigt räumliche Informationen zu der aktuellen Umgebung sowie eventuell seinem Zielort. Gleichzeitig bedingen mobile Anwendungen ein gewisses Umdenken in der Datenaufbereitung, da das mobile Netzwerk nicht mit den gleichen Datenmengen belastet werden kann, wie das kabelgebundene Netz. Außerdem sind mobile Geräte bezüglich ihrer Rechenleistung und auch der Größe und Auflösung der Bildschirme limitiert im Vergleich zu den PCs. Zur Vermittlung geografischer Informationen müssen somit neue Wege gefunden werden. Die Möglichkeit der Personalisierung bietet in diesem Fall eine große Chance.

Der Nutzer kann auch hier von einer multirepräsentativen Datenbasis profitieren. Auf der einen Seite helfen Daten unterschiedlicher Informationsdichte, die auf die aktuelle Situation optimal abgestimmten Angaben zu finden. Der mobile Nutzer benötigt eine geringere Informationsmenge als der PC-Nutzer. Es wird ein Kompromiss gesucht, der zum einen die Kommunikation zwischen dem Serviceanbieter, beispielsweise einem Kartendienst, und dessen

⁵Open Geospatial Consortium

⁶Universal Mobile Telecommunication System

Nutzer nicht durch einen Informationsüberfluss stört, auf der anderen Seite jedoch genügend Informationen bietet, um den Klienten zu befriedigen. Liegen Daten unterschiedlicher Auflösung, respektive Informationsdichte, in einer MRDB vor, so kann diese für jede Situation eine passende Lösung bereitstellen, ohne die Datensätze durch aufwändige Online-Prozesse, beispielsweise einer Generalisierung, aufbereiten zu müssen. Auf der anderen Seite kann auch der mobile Nutzer auf eine breite Datenbasis zurückgreifen und die für seine Ansprüche sinnvollen Informationen auf einfache Weise aus dem System extrahieren. Gerade in dem Bereich der mobilen Anwendungen möchte dieser die relevanten Informationen durch möglichst wenig Klicks und Zeitverlust erreichen.

Bisherige Webanwendungen ermöglichen inzwischen den Zugriff auf unterschiedliche Datenquellen. Um jedoch die oben genannten Vorteile der MRDB innerhalb einer Webanwendung nutzen zu können, bedarf es Lösungen zur Integration multirepräsentativer Datenquellen. Auf der anderen Seite sind die bisherigen Entwicklungen in dem Bereich der MRDB auf die Verwendung dieser innerhalb interner Prozesse ausgelegt. Die Datenanbieter benötigen ein System, welches ihre gesamten Daten in einem System integriert, um die Datenhaltung und -fortführung zu erleichtern und den Arbeitsablauf zur Ableitung von Folgeprodukten wie Karten zu optimieren. Die Probleme getrennter, monorepräsentativer Datenhaltung treten jedoch nicht nur bei internen Prozessen, sondern genauso auch bei der externen Nutzung der verschiedenen Datenquellen auf. Eine Verbesserung der Situation entsteht durch die Bereitstellung von multirepräsentativen Daten innerhalb eines offenen Netzwerkes, in dem der Nutzer über definierte Schnittstellen Zugriff auf derartige Daten hat, entsprechend dem Konzept des OGC. Aus dem Blickwinkel des Nutzers heraus entstehen neue Anwendungsfelder und Lösungen der oben aufgezeigten Probleme.

1.2 Zielsetzung der Arbeit

Das oberste Ziel der vorliegenden Arbeit besteht in der Integration einer MRDB in eine Webservice-Umgebung. Es sollen hiermit verbundene Probleme aufgedeckt und entsprechende Lösungen erarbeitet sowie auf der anderen Seite die neuen Möglichkeiten für den Nutzer evaluiert werden. Gleichzeitig können durch den Vergleich mit bestehenden Lösungen, soweit vorhanden, die Vorteile einer multiskaligen Datenbasis begründet werden. Dabei konzentriert sich die Arbeit auf die Integration multiskaliger Geodaten in eine MRDB, also von Geodaten unterschiedlicher geometrischer und semantischer Auflösung, wobei die Möglichkeit zur Übertragung auf den allgemeinen Fall einer multirepräsentativen Datenbasis jederzeit erhalten bleiben soll.

Zur Evaluierung der Möglichkeiten und Probleme soll der gesamte Arbeitsablauf zum Aufbau eines Webdienstes, basierend auf einer MRDB, beleuchtet werden, angefangen bei der Modellierung der Datenbank, der Integration bestehender sowie neu aufgebauter (generalisierter) Datensätze, über die Entwicklung einer offenen Schnittstelle, bis hin zum Aufbau eines entsprechenden Webdienstes sowie einer Anwendung für den mobilen und den immobilien Nutzer. Dabei steht bei der gesamten Entwicklung die webbasierte Datennutzung im Fokus der Überlegungen.

Das Thema multirepräsentativer Datenbanken wird damit auch aus der Sicht des Nutzers beleuchtet, neue Formen der Informationsvermittlung und -visualisierung werden aufgezeigt. Wie kann eine derartige Datenbank aufgebaut werden, welche Möglichkeiten und Besonderheiten treten hier auf? Auf welche Weise kann einer möglichst breiten Nutzerschicht der Zugriff auf multirepräsentative Daten offeriert werden und welche neuen Möglichkeiten erwachsen hieraus? Der Autor der vorliegenden Arbeit geht somit davon aus, dass die Symbiose aus einem offenen, webbasierten System auf der einen und einer multirepräsentativen Datenbasis auf der anderen Seite die Vorteile beider Systeme summiert. Gleichzeitig erfordern die neuen Voraussetzungen eine Anpassung beider Elemente aneinander.

Das in dieser Arbeit vorgestellte Konzept ermöglicht dem Nutzer erstmalig über eine Webservice-Schnittstelle auf die Daten und die Struktur einer MRDB gezielt zuzugreifen und diese zu seinem Vorteil zu nutzen. Vor allen Dingen die Verknüpfungen zwischen den alternativen Repräsentationen bieten ein neues Informationspotenzial. Diesbezüglich sollen insbesondere die neuen Möglichkeiten der Informationsübermittlung und Visualisierung von Geodaten auf mobilen Geräten beleuchtet werden.

Das beschriebene System wird in Form eines Prototypen aufgebaut. Daher verfolgt die vorliegende Arbeit neben den theoretischen Grundlagen den chronologischen Aufbau eines derartigen Systems, begleitet von strategischen Überlegungen.

1.3 Aufbau der Arbeit

Das aktuelle Kapitel beschreibt die Motivation zur Erforschung der beschriebenen Sachverhalte und Probleme und skizziert die Ziele und Leistungen der vorliegenden Arbeit.

Kapitel 2 führt ein in die Grundlagen der Webdienste und insbesondere der Geo-Webdienste. Es werden die Vorteile dieser neuartigen Technologie sowie grundlegende Konzepte bezüglich der Systemarchitektur und des Datenaustausches innerhalb eines Webdienstes beschrieben. Dabei spielen Geo-Webdienste eine besondere Rolle unter den Webservices, da diese oftmals die Grundlage eines Dienstes bilden. Vor allen Dingen das OGC hat hier verstärkte Bemühungen zur Integration der Geokomponente in das WWW⁷ gezeigt. Schließlich werden innerhalb dieses Kapitels die Besonderheiten und Probleme der mobilen Nutzung von Webdiensten vorgestellt.

Kapitel 3 verschafft anschließend einen Überblick über die Motivation und Problematik multirepräsentativer Datenbanken. Es werden die bisherigen Forschungsbemühungen auf diesem Gebiet skizziert sowie die unterschiedlichen Möglichkeiten zur Modellierung einer solchen herausgearbeitet.

In dem folgenden Teil wird anschließend ein Weg aufgezeigt, beide Technologien zu verbinden, um die Vorteile einer MRDB innerhalb einer Webserviceumgebung nutzen zu können. Es wird ein Prototyp zur Verwendung multiskaliger Daten innerhalb eines Webdienstes erarbeitet.

Kapitel 4 identifiziert hier zunächst anhand von Anwendungsszenarien die Problematik bestehender Systeme und erarbeitet gleichzeitig einen Anforderungskatalog an einen Informations-Webservice. In diesem Kapitel wird eine Architektur zur Bereitstellung multiskaliger Daten über das Internet vorgestellt.

Kapitel 5 beschreibt anschließend den Aufbau einer multiskaligen Datenbank, angefangen bei dem Datenbankkonzept über den Datenbankentwurf bis hin zur Integration bestehender Testdaten sowie der Ableitung von unterschiedlich aufgelösten Geodaten durch verschiedene Generalisierungsverfahren.

Kapitel 6 stellt den Entwurf und die Implementierung einer neuen Webservice-Schnittstelle zwischen der multiskaligen Datenbank auf der einen und dem Internetnutzer auf der anderen Seite dar. Ein solcher MRDB-Service, welcher auf dem OGC-WFS aufsetzt, stellt hier die notwendigen Funktionalitäten bereit, um mit der multiskaligen Datenbasis interagieren und von dem besonderen Schema der MRDB, wie beispielsweise den Verknüpfungen zwischen den alternativen Repräsentationen, profitieren zu können. Das Kapitel erarbeitet die notwendigen Funktionalitäten eines auf einer MRDB basierenden Webdienstes und beschreibt dessen Umsetzung.

Kapitel 7 stellt einen Karten-Webservice vor, welcher über die im vorangegangenen Kapitel 6 beschriebene MRDB-Schnittstelle auf die multiskaligen Daten zugreift. Es werden hier die mit Hilfe der Anwendungsszenarien aus dem Kapitel 4.1 erarbeiteten Anforderungen an einen derartigen Service umgesetzt. Das Kapitel beschreibt Lösungen zur Visualisierung von Geoinformationen auf mobilen Geräten, welche auf den Möglichkeiten einer MRDB aufbauen. Gleichzeitig wird an dieser Stelle ein Vergleich angestellt zu der alternativen Nutzung einer Online-Generalisierungs-Komponente.

Kapitel 8 beschreibt schließlich zwei typische Nutzerplattformen: Eine SVG-basierte Anwendung für mobile Geräte, welche auf den zuvor erläuterten Kartendienst zurückgreift, sowie eine desktopgebundene GIS-Applikation, welche sich direkt der MRDB-Schnittstelle bedient.

Kapitel 9 fasst die Ergebnisse der Arbeit zusammen, diskutiert die Grenzen der vorgestellten Entwicklungen, beschreibt weitere Aspekte, welche in der vorliegenden Arbeit lediglich angedeutet wurden und eröffnet einen Ausblick auf zukünftige Möglichkeiten, welche auf den vorgestellten Ergebnissen aufsetzen.

⁷World Wide Web

2 Grundlagen: Webdienste und Geo-Webdienste

In der vorliegenden Arbeit soll eine Verbindung hergestellt werden zwischen zwei für die Nutzung von Geoinformationen gewinnbringenden Technologien, den Webservices auf der einen und der Speicherung und Integration multirepräsentativer Geodaten auf der anderen Seite.

In diesem Kapitel folgt zunächst eine grundlegende Beschreibung der ersten Komponente, den Webservices. Was sind Webservices, wie werden diese realisiert und vor allen Dingen, worin liegt der große Vorteil dieser Technologie? Nach einer allgemeinen Beschreibung folgt der Übergang auf den speziellen Bereich der Geo-Webdienste und der mobilen Webdienste. Webservices bieten auch für den Bereich der Geoinformationen entscheidende Vorteile, denn aus dieser Technologie erwachsen völlig neue Möglichkeiten der Nutzung und Verbreitung von räumlichen Daten und hiermit verbundenen Diensten. Geoinformationen bilden die Grundlage vieler Entscheidungsprozesse und anderer Services. Je einfacher es für den Anwender ist, an diese Informationen zu gelangen, desto eher werden diese entsprechend genutzt. Die Webservice Technologie bietet für Geoinformationsdienste verschiedenster Art eine Möglichkeit, auf einfache Weise Daten und Dienste für eine große Anzahl potenzieller Nutzer bereitzustellen, nämlich für alle diejenigen, welche Zugriff auf das WWW haben.

Neue Märkte erwachsen durch neue Gerätegenerationen und bessere, drahtlose Übertragungstechniken auch auf dem mobilen Sektor. Diese stellen zusätzliche Herausforderungen an die Webdienste und die verwendeten Daten. Chancen und Besonderheiten, neueste Generationen mobiler Endgeräte, LBS¹ und auf diese Situation zugeschnittene Geodaten werden hier vorgestellt. Dabei sollen jeweils die grundlegenden Praktiken und Techniken, die Architektur und Infrastruktur der verschiedenen Dienste dargelegt werden.

Für den Kontext dieser Arbeit ist neben der Zweckmäßigkeit von Geo-Webservices vor allen Dingen auch der Aspekt der Datenhaltung innerhalb eines Webservices interessant. Wie wird also die Datenhaltungskomponente in die konzeptionelle Gesamtarchitektur eines Webservice integriert?

2.1 Webservices - Eine Definition

Seit wenigen Jahren hat sich auf der Basis des Internets², beziehungsweise des WWW als ein Dienst des Internets, die Webservice-Technologie etabliert. Vorangetrieben werden diese Entwicklungen und Technologien vor alle Dingen durch das W3C³, welches im Jahre 2001 die „Webservice Activity“ gründete⁴. Das Ziel dieser Aktivität besteht in der Entwicklung von Technologien, um das volle Potenzial der Webservices auszuschöpfen.

Services stellen laut ISO 19119 eine Sammlung von Operationen dar, welche über ein Interface zugänglich sind und es einem Nutzer erlauben, einen Prozess zu seinem Vorteil aufzurufen. Sie verkörpern Komponenten, welche ihrerseits zusammengefügt werden können, um einen größeren, umfangreicheren Service aufzubauen. Man spricht hier von „Service chaining“, also dem Zusammenketten von einzelnen Diensten zu einem größeren Service. Als Beispiel sei hier Microsofts *Passport* genannt, ein Authentisierungsservice⁵. Serviceanbieter können diesen Dienst in ihre eigene Applikation integrieren ohne eine eigene Zugangskontrolle implementieren zu müssen. Die Anwender werden zur Passwortabfrage automatisch an diesen Service weitergeleitet. Mohan (2002) definiert einen Webservice wie folgt: „Webservices stellen eine neue Art von Web Applikationen dar. Sie sind eigenständige, sich selbst beschreibende, modulare Applikationen, welche über das Netz publiziert, lokalisiert und aufgerufen werden können. Webservices bieten jegliche Arten von Funktionen, von einer einfachen Anfrage bis hin zu einem komplexen Geschäftsprozess [...] Sobald ein Webservice aufgestellt ist, können andere Applikationen (und Webservices) diesen Service finden und aufrufen.“

Webdienste verkörpern also die Vision eines interoperablen und ubiquitären Systems. Sie wären allerdings nach Doyle & Reed (2001) vor wenigen Jahren noch zu ineffizient und kostenintensiv gewesen, so dass sie sich

¹Location based services

²Interconnected Networks

³World Wide Web Consortium

⁴W3C Webservice Activity: <http://www.w3.org/2002/ws/>

⁵<https://accountservices.passport.net>

zum damaligen Zeitpunkt noch nicht etablieren konnten. Erst durch die Entwicklung neuer IT⁶-Trends und der Weiterentwicklung der Infrastruktur sind immer größere und günstigere Bandbreiten und auch Speicher entstanden. Genauso hat die Verfügbarkeit dynamischerer Inhalte und der Technologie zur Unterstützung dieser diese Entwicklungen begünstigt. Doyle & Reed (2001) führen weiter aus, dass gleichzeitig die Verbreitung und Diversität der Computer, mit unterschiedlichsten Zugriffsplattformen, die Wichtigkeit einer verbindenden „Middleware“ drastisch erhöht hat, während die Kosten, also Bandbreite und Speicherplatz, weniger kritisch zu betrachten sind. „Middleware“ bezeichnet in diesem Zusammenhang anwendungsunabhängige Technologien, die zwischen den einzelnen Komponenten vermitteln, so dass die Komplexität und Heterogenität der einzelnen Applikationen und der Infrastruktur nivelliert wird. Die Stärken des Internets als Informationsanbieter, dessen Einfachheit im Zugriff und seine Allgegenwärtigkeit, lösen die Probleme bisheriger Middlewarekomponenten mit ihrer geringen Interoperabilität. Webservices bieten einen einheitlichen Zugriff auf die vielfältig gestalteten Middlewarekonzepte, wie beispielsweise *CORBA*, *Enterprise JavaBeans*, *XML/SOAP*, *COM+* oder *.NET*. Ein Webservice bietet Zugriff auf einen Dienst, welcher dann in einer anderen Form, also auf Basis einer individuellen Middleware, implementiert sein kann.

2.2 Architektur eines Webdienstes

2.2.1 Client-Server-Architektur

Die traditionelle Art der Modellierung auch von Webdiensten stellt die Client-Server-Architektur dar. Hierbei sendet immer ein Element dieser Architektur eine Anfrage an ein zweites, welches diese Anfrage bearbeitet. Der Anfragende wird dabei als *Client*, und der Ausführende (oder Diener) als *Server* bezeichnet.

Typischerweise werden Webservices dabei durch eine *Drei-Ebenen-Architektur* oder eine *n-Ebenen-Architektur* beschrieben. Die drei Ebenen werden, wie in Abbildung 2.1 abzulesen ist, von der *Präsentationsebene*, der *Applikations- oder Logikschicht* und der *Datenebene* gebildet. Der modulare Aufbau dieser Architektur ermöglicht die Trennung der einzelnen Komponenten, sowohl entwicklungs-technisch als auch physikalisch. Definierte Schnittstellen ermöglichen das Zusammenarbeiten der verschiedenen Schichten. Gleichzeitig kann jede der Schichten individuell verändert oder ausgetauscht werden.

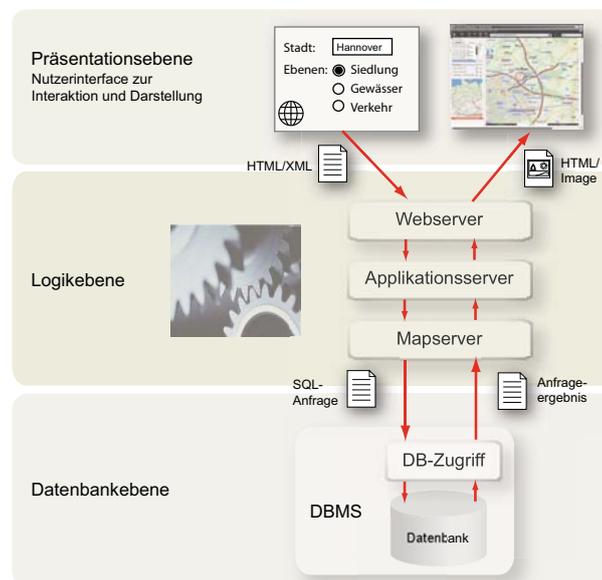


Abb. 2.1: Drei-Schichten Systemarchitektur eines Webservices

Dabei können diese drei Elemente auf drei Computersystemen verteilt sein oder auch auf einer einzigen Maschine laufen. Im ersten Fall spricht man von einer Client-Server-Applikation, im letzten Fall von einer Stand-alone Anwendung. Es muss also eine begriffliche Unterscheidung getroffen werden zwischen dem logischen Client-Server Konzept auf der einen und dem physikalischen Client-Server Konzept auf der anderen Seite (Peng & Tsou 2003, S. 98f.).

⁶Informationstechnologie

Die Präsentationsebene stellt die Möglichkeit für den Anwender dar, mit dem System zu interagieren. In den meisten Fällen ermöglicht eine grafische Nutzeroberfläche, Anfragen an den Server zu stellen, um so letztendlich Daten abzufragen und diese später bearbeiten zu können. Diese Schicht dient gleichzeitig dazu, die angefragten Informationen zu präsentieren. Es werden dabei eine Reihe unterschiedlicher Plattformen bedient, ermöglicht durch den modularen Aufbau des Systems. Der Serviceaufruf wird in Form von XML-codierten Daten auf Grundlage des HTTP⁷ an die Applikations-, respektive Logikebene übertragen. Daher werden im Regelfall Webbrowser als Nutzerschnittstelle genutzt. Man spricht in diesem Fall von *Thin-Clients*, da die Hauptarbeit zur Prozessierung in der mittleren Ebene und nicht beim Client durchgeführt wird. Neben herkömmlichen, HTML⁸-basierten Oberflächen existieren eine Reihe von Erweiterungen, wie zum Beispiel *dynamisches HTML*, welches Skriptsprachen wie *Java-Script* zur Erweiterung nutzt. Hierdurch wird es möglich, dynamische Aktionen zu aktivieren, damit der Nutzer zum Beispiel bestimmte Grafikobjekte auswählen kann oder durch *Mouse-over*-Aktionen Informationen von bestimmten Objekten abfragen kann. Des Weiteren werden clientseitige Applikationen verwendet, wie *Java Applets* oder *-Beans*, *Active-X Komponenten* oder andere Erweiterungen der Webbrowser, wie beispielsweise das *Macromedia-Flash-Plugin* oder das *Adobe-SVG-Plugin*. Neben den Vorteilen der verschiedenen Erweiterungen der Webbrowser besteht bei Verwendung dieser jedoch gleichzeitig eine Abhängigkeit hiervon. Jeder potenzielle Nutzer muss diese Erweiterungen auf seinem Gerät installieren, um einen bestimmten Service nutzen zu können.

Die Schicht der Prozesslogik (Geschäftslogik) koordiniert die Anwendungen und führt Befehle aus oder leitet diese weiter. Gleichzeitig werden Daten von der darüber und darunter liegenden Ebene angenommen beziehungsweise an diese weitergeleitet. Diese mittlere Ebene kann dabei wiederum aus mehreren Ebenen bestehen. Somit wird dann aus der Drei-Schichten- eine n-Schichten-Architektur (*n-tier*). Diese Schicht besteht aus einem Webserver und einem Applikationsserver (zum Beispiel JavaEE⁹), wobei der Applikationsserver einen Webserver enthalten kann (zum Beispiel enthält der *WebSphere* Applikationsserver der Firma *IBM* den Webserver *Tomcat*) oder der Applikationsserver ergänzt den Webserver. Die Aufgabe des Webserver besteht darin, eine HTTP-Anfrage entgegenzunehmen und eine entsprechende Antwort zu senden. Ein sogenannter *Listener* nimmt die HTTP-Anfrage an, analysiert diese und leitet sie an eine entsprechende Operation weiter. Im weiteren Verlauf wird die resultierende Nachricht dann von diesem verpackt und an die Nutzerschnittstelle zurückgesendet. Dabei kann der Webserver entweder vorbereitete Dokumente, wie HTML-Dokumente, an den Client zurücksenden oder aber die Anfrage an andere Programme weiterleiten. Diese anderen Programme werden ihrerseits von einem Applikationsserver verwaltet. Dabei werden die Applikationen speziell für den jeweiligen Applikationsserver implementiert. Der Applikationsserver dient somit als Vermittler zwischen dem Webserver und den eigentlichen Applikationen des Serviceanbieters, was im Falle der Verwendung des *Java2EE* Applikationsserver zum Beispiel die *Servlets* sind. Im Falle eines Geoservices befindet sich hinter dem Applikationsserver zum Beispiel ein Mapserver oder Feature-Server, welcher die Funktionalitäten bereitstellt, um Geodaten abfragen und bearbeiten zu können.

Die Datenebene besteht in dem meisten Fällen aus einem DBMS¹⁰, welches den Zugriff auf die Daten regelt und diese in einer bestimmten Form speichert. Weiterhin existieren hier Mechanismen zum schnellen Auffinden und Abrufen dieser. Die Abfrage aus der Logikebene erfolgt über die SQL¹¹. Zur effizienten Speicherung von Geodaten sind bestimmte Strukturen und Standards notwendig (vgl. Abschnitt 2.6). Alternativ werden die Daten nicht in Datenbanken, sondern filebasiert vorgehalten.

Client-Server-Architekturen bieten nur einen streng hierarchischen Mechanismus. Ein Client fordert Daten mittels eines Servercalls von einem bestimmten Server an. Dieser reagiert auf die Anfrage des Clients, sammelt die Daten, stellt diese zur Verfügung und sendet sie dem anfragenden Client als Antwort in einem definierten Format. In dieser strengen Hierarchie gibt es spezielle Server, die Anfragen entgegennehmen und Antworten verschicken. Clients können nur Anfragen stellen und müssen auf Antworten des Servers warten (Lackner 2004). Alternativ hierzu existieren die sogenannten Peer-to-Peer (P2P) Modelle. Lackner (2004) schreibt hierzu: „P2P-Modelle treten einerseits als Server und andererseits als Client auf. [...] Im Gegensatz zu Client-Server Anwendungen sind momentane P2P-Lösungen nur schwer administrierbar, [...] da diese entsprechend der Einsatzmöglichkeit verteilt über das Internet vorliegen und auch die Administrierung dezentral erfolgt und im Verwaltungsbereich des jeweiligen Anbieters liegt.“

⁷Hypertext Transfer Protocol

⁸Hypertext Transfer Markup Language

⁹Java Enterprise Edition

¹⁰Datenbank Managementsystem

¹¹Structured Query Language

2.2.2 Die Web Service Architecture (WSA) des W3C

Um nun die Interoperabilität zwischen den Servern und den Clients zu gewährleisten, gilt es, gewisse Spezifikationen und Standards zu beschreiben. Das W3C hat hierzu eine Empfehlung einer WSA¹² veröffentlicht (Booth et al. 2004). Das W3C betont: „Diese Publikation beschreibt ein konzeptionelles Modell sowie einen Rahmen zum Verständnis von Webservices und die Beziehungen zwischen den einzelnen Komponenten. Hierin soll keine Spezifikation zur Implementierung eines solchen Dienstes gegeben werden und verhängt auch keine Restriktion über die Art und Weise, wie Webservices miteinander kombiniert werden sollen. Die WSA beschreibt zum einen die minimalen Charakteristika, die allen Webservices gemeinsam sind und einige Charakteristika, welche von vielen, aber nicht von allen Webservices benötigt werden. Die WSA beschreibt eine Interoperabilitäts-Architektur: Sie identifiziert die globalen Elemente des Webservice-Netzwerkes, welche benötigt werden, um die Interoperabilität zwischen den einzelnen Webservices sicherzustellen.“

Das W3C definiert Webservices somit aus technischer Sicht: „Ein Webservice ist ein Softwaresystem, das entwickelt wurde, um Interaktionen über ein Netzwerk zwischen zwei Maschinen abwickeln zu können. Es unterstützt ein spezielles, maschinell nutzbares Interface (WSDL¹³). Verschiedene Systeme können unter Einhaltung der Spezifikation SOAP¹⁴-Messages an den Webservice stellen. Typischerweise werden Messages in XML-serialisierter Form mit HTTP oder anderen webspezifischen Standards übertragen“ (Booth et al. 2004).

Der Datenaustausch bei der WSA erfolgt in XML¹⁵ Notation. Durch die Wahl von XML als Transportmedium ist eine Unabhängigkeit bezüglich Plattform und Programmiersprache gegeben. XML stellt eine sowohl für den Menschen als auch für Maschinen und Software lesbare Form der Codierung von Informationen dar. Die Sprache ist erweiterbar, kann also den jeweils zu übertragenden Daten angepasst werden, bleibt jedoch trotzdem für den Empfänger lesbar. XML ist selbstbeschreibend, was bedeutet, dass innerhalb des Dokumentes auch dessen Elemente und Beziehungen beschrieben werden. Sie trennt die Inhalte von deren Darstellung. Ein weiterer Vorteil dieser Sprache besteht darin, dass die hierauf basierenden Dokumente einfach transformiert werden können (durch Verwendung von XSLT¹⁶ Dokumenten) und somit Daten, welche auf dem XML-Standard basieren, kombiniert und deren Inhalte auf einfache Weise extrahiert und auf ein anderes, XML-basiertes Format übertragen werden können. Einen für den Bereich räumlicher Informationen wichtiger XML-basierter Standard stellt die GML¹⁷ dar.

Eine einfache Webservice-Architektur besteht nun aus drei Komponenten, dem *Service-Anbieter*, dem *Service-Broker* und dem *Service-Konsumenten*, oftmals als Client bezeichnet (vgl. Abbildung 2.2). Ein Provider meldet seinen Dienst bei einem Broker an. Um einen gewünschten Dienst zu suchen, wendet sich auch der Client an den Broker, der den entsprechenden Dienst in einem Verzeichnis veröffentlicht hat. Dieser dient somit als Vermittler zwischen Client und Provider. Ein Standard zur Registrierung von Webservices bei Brokern ist UDDI¹⁸, welche ebenfalls in XML definiert ist. Der Broker sammelt die registrierten Einträge in einem Katalog. Zur Beschreibung der registrierten Dienste wird die WSDL eingesetzt, welche, ebenfalls auf XML basierend, die Operationen, die notwendigen Parameter zum Aufruf dieser sowie die Rückgabewerte eines Dienstes beschreibt. Schließlich legt das SOAP oder auch XML-RPC¹⁹ einen Standard zur Interaktion zwischen den einzelnen Services fest. SOAP greift seinerseits auf Standards wie beispielsweise XML zur Darstellung der Daten sowie Internet-Protokolle (TCP/IP) zur Übertragung der Informationen zurück.

Dabei stellt die WSA eine Webservice-spezifische Ausprägung einer sogenannten „serviceorientierten Architektur“ dar. Die drei Komponenten WSDL, UDDI und SOAP bilden zusammen ein Element dieser SOA²⁰.

Webservices realisieren also mit Hilfe standardisierter Schnittstellen die drei wichtigsten Teile im Zusammenspiel zwischen Client und Server: Das *Publizieren* (publish) und *Auffinden* (find) der Daten und Services sowie den *Datenaustausch* (bind). Erreichbar sind Webservices über einen eindeutigen URI²¹.

Zur Realisierung oben beschriebener Webservices existieren auf dem Markt unterschiedlichste Lösungen verschiedener Anbieter, die ähnliche aber doch unterschiedliche Strategien verfolgen. Die beiden am weitesten

¹²Web Service Architektur

¹³Web Services Description Language

¹⁴Simple Object Access Protocol

¹⁵Extensible Markup Language

¹⁶Extensible Stylesheet Language Transformation

¹⁷Geography Markup Language

¹⁸Universal Description, Discovery and Integration

¹⁹Remote Procedure Call

²⁰Service Oriented Architecture

²¹Uniform Resource Identifier

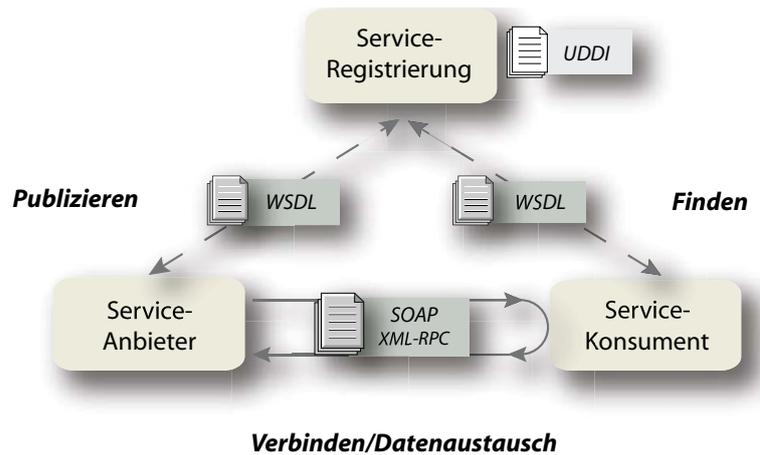


Abb. 2.2: Webservice Architektur des W3C

verbreiteten Lösungen stellen die JEE²² Technologie von *Sun Microsystems* in der aktuellen Version *JEE5* (welche die frühere Sun ONE²³ Plattform seit September 2003 ersetzt) und das *.NET* Paket der Firma *Microsoft* dar. Ziele von *.NET* sind der Einsatz in verteilten, webbasierten Diensten, mit der Unterstützung verschiedener Programmiersprachen bei der Produktentwicklung. Die Möglichkeit, den Web-Browser als Transportmittel für den Informationsaustausch unterschiedlicher Medien nutzen zu können, spielt dabei eine entscheidende Rolle für den Anwenderbereich. Die JEE beruht auf einer n-Ebenen-Architektur. Alle JEE-Applikationen bestehen zumindest aus drei Schichten, der Client-Schicht, dem JEE-Server und der Datenbankschicht. Die JEE Architektur basiert auf der Programmiersprache Java. Diese Basis soll den Ansatz, plattformübergreifenden Code zu erzeugen, garantieren.

2.2.3 Das Semantische Web

Das *Semantic Web* stellt eine Entwicklung dar, welche es ermöglicht, die im Internet verfügbaren Informationen mit für Maschinen nutzbarer Semantik zu erweitern. Es handelt sich dabei um eine vom W3C vorangetriebene Initiative²⁴. Der Austausch von Daten im WWW soll verbessert werden, denn bislang werden nur Dokumente ausgetauscht. Des Weiteren sind die Daten an die jeweiligen Applikationen gebunden und können nicht problemlos untereinander ausgetauscht werden. Das Ziel ist es, zum einen einheitliche Formate zum Austausch von Daten hervorzubringen, auf der anderen Seite aber auch eine formale Sprache, welche die Verbindung zwischen den Daten und den realen Objekten, die sie repräsentieren, beschreibt. Hierdurch soll den Anwendern von Daten und Diensten die Nutzung erleichtert werden. Reisen können automatisch inklusive Mietauto, Hotel und Flug gebucht werden oder man bestellt eine Karte für ein Konzert und das System trägt den Termin automatisch in den Kalender ein.

Erste Ansätze versuchen, die Informationen für den Menschen lesbar darzustellen, auf der anderen Seite aber auch formal für Maschinen nutzbar zu beschreiben. Maschinen sollen den Bedeutungsinhalt verstehen können, anstatt nur bestimmte Datentypen verarbeiten zu können. In diesem Zusammenhang wurden zwei Standards entwickelt, RDF²⁵ und OWL²⁶. Das RDF basiert auf der XML und beschreibt einfache semantische Zusammenhänge. RDF wird insbesondere zur Definition von Metadaten für Internet-Seiten verwendet, damit diese automatisch verarbeitet werden können, vor allen Dingen von Suchmaschinen. Die OWL dagegen bietet ein Vokabular zur Beschreibung komplexer semantischer Zusammenhänge. Es können hierdurch Ontologien erstellt und publiziert werden.

Das semantische Web wird dabei nicht bloß als Sonderform des WWW angesehen, sondern als dessen nächste Ebene (Shadbolt et al. 2006).

²²Java Platform, Enterprise Edition

²³Open Network Environment

²⁴W3C Semantic Web Initiative: <http://www.w3.org/2001/sw/>

²⁵Resource Description Framework

²⁶Web Ontology Language

2.3 Geo-Webdienste

In dem vorangegangenen Abschnitt wurde herausgestellt, dass Webservices Applikationen und Komponenten darstellen, welche über das Internet Daten austauschen, Aufgaben teilen und Prozesse automatisieren können. Webdienste eignen sich hierdurch zur Kostenreduzierung bei der Softwareintegration sowie der Verbreitung von Daten. Die Webservice Infrastruktur erweitert die Zugriffsmöglichkeiten der Nutzer auf Daten und andere Ressourcen. Diese Servicestruktur lässt sich auch auf den Bereich von Geodaten übertragen. Dem Nutzer wird die Möglichkeit geschaffen, über das Internet auf Geodaten zugreifen zu können und von verschiedenen Services zur Prozessierung der Daten zu profitieren. „Seit Ende der 90er Jahre vollzieht sich auf dem Gebiet der Geografischen Informationssysteme ein Wandel von informationsaufbereitenden zu kommunikationsunterstützenden Systemen. Kennzeichen sind die intensive Nutzung der Internettechnologie sowie die konsequente Verwendung dienstbasierter Architekturen“ (Fitzke & Greve 2002).

Die Geodaten ermöglichen die Verknüpfung verschiedenster Tatbestände über die räumliche Lage. Hierin besteht der Hauptunterschied zwischen einem Geo-Webdienst und anderen Webdiensten sowie gleichzeitig der Hauptgrund, weswegen Geodaten das Fundament für andere Webdienste bilden.

Eine Definition des Begriffes Geo-Webservices ist in dem schweizerischen Bundesgesetz über Geoinformation (GeoIG) zu finden: „Geodienste sind vernetzbare Anwendungen, welche die Nutzung von elektronischen Dienstleistungen im Bereich der Geodaten vereinfachen und Geodaten in strukturierter Form zugänglich machen.“ In den Erläuterungen zum GeoIG wird diese Definition weiter ausgeführt: „Man kann einen Dienst als die Folge einer oder mehrerer Operationen definieren, die es ermöglichen, einen Bedarf zu decken oder der Nachfrage einer Person nachzukommen. Wird dieser Dienst über das Internet zur Verfügung gestellt, spricht man von Webservice oder einfach von Service. Falls diese Dienste eine geografische Komponente aufweisen, spricht man von Geodiensten: Zum Beispiel wenn man eine Adresse auf einer Straßenkarte oder den kürzesten Weg zwischen zwei Adressen herausfinden möchte.“

Die räumliche Dimension unterstützt Menschen im Bereich der Wissenschaft, Stadtplanung, Umweltmanagement, Transportplanung und jeder anderen Anwendung in dem Bereich der Geoprozessierung (OGC 2004). Das „Räumliche“ wird zum essenziellen Element vieler Arbeitsabläufe.

Ein Hauptziel der Verbreitung von Geodiensten über das Internet besteht darin, eine Grundlage zur Nutzung von Geoinformationen zu schaffen, damit die Menschen und Organisationen ihre Entscheidungen auf einer besseren Informationsbasis treffen können (Peng & Tsou 2003, S. 651). Dabei werden diese Services sowohl von privaten als auch von öffentlichen Dienstleistern angeboten und umfassen die Bereitstellung von Daten ebenso wie von Operationen.

Die Datenkomponente setzt sich dabei wie folgt zusammen:

- *Daten* sind die Rohinformationen eines Objektes
- *Metadaten* beschreiben diese Daten
- *Namen* oder IDs werden benutzt, um Daten zu registrieren und zu finden
- *Beziehungen* beschreiben Verknüpfungen zwischen einzelnen Objekten

Die Operationskomponente bietet Operationen, wie räumliche Suchfunktionen oder Abfragen, und umfassen nach Peng & Tsou (2003, S. 641) vier Gruppen:

- *Viewer* und *Editor*: Nutzerschnittstelle, um Informationen zu betrachten, zu bearbeiten oder zu verwalten
- *Katalogdienste* sammeln Metadaten und Informationen über die Daten und beschreiben diese. Der Nutzer durchsucht zunächst diese Kataloge, bevor er zum Datenlager gelangt. Katalogdienste können z.B. umgesetzt werden mit Hilfe der WSDL
- *Datenlager* dienen der Sammlung und Organisation der Daten. Hier werden die Daten gespeichert und mit Hilfe von Indizes und Namen organisiert, um diese schnell auffinden zu können. Auf der Grundlage des Namens und des Index können diese Operationen die Quelldaten finden und aufrufen. Derartige Dienste können umgesetzt werden z.B. mit Hilfe der UDDI
- *Operatoren* sind Komponenten, welche bestimmte Funktionalitäten zur Geoprozessierung bieten. Dies sind zum Beispiel Analysetools wie Pufferanalysen, räumliche Überlagerungen oder Projektionsberechnungen. Diese können Daten transformieren oder kombinieren und generieren eine Ausgabe. Derartige Operatoren können umgesetzt werden z.B. mit Hilfe des SOAP

2.3.1 Standards für Geo-Webdienste

Im Abschnitt 2.2 wurde eine Beschreibung der Komponenten und Standards zum Aufbau eines Webservices gegeben. Geo-Webservices können basierend auf diesen Festlegungen aufgebaut werden. Soll allerdings das Zusammenspiel zwischen verschiedenen Services, welche Geodaten oder -dienste bereitstellen, funktionieren, so ist eine vertiefte Standardisierung grundlegender Geodienste notwendig.

Sollen beispielsweise im Fall einer Katastrophe die relevanten Daten idealerweise sofort nach einem Ereignis schnell und einfach gesammelt werden, um die Lage richtig beurteilen zu können, so bietet die oben genannte Technologie hier die notwendigen Voraussetzungen. In der Realität findet sich dagegen eine Reihe von Hindernissen, so dass es Stunden oder gar Tage dauert, um alle relevanten Informationen, von der topografischen Karte über Informationen der Bevölkerungsverteilung, Infrastruktur oder anderen räumlich verbreiteten, relevanten Phänomenen zusammenzutragen. In einer visionären Infrastruktur dagegen hat der Nutzer mit Hilfe seiner Hardware Zugriff auf verschiedenste Datenquellen und Prozessierungsdienste. Um diese Vision zu realisieren, arbeiten diverse Gremien an einer Lösung zur Schaffung eines interoperablen Netzes von Geoinformationsdiensten. Zu nennen sind hier insbesondere die ISO²⁷ und das OGC.

ISO

Die ISO beschreibt sich selbst und seine Tätigkeit wie folgt: „ISO ist ein globales Netzwerk, das feststellt, welche internationalen Standards in Unternehmen, Regierungen und der Gesellschaft benötigt werden, entwickelt sie in Zusammenarbeit mit den Sektoren, die diese benutzen werden, passt sie an [...] basierend auf nationalem Input und stellt diese dann bereit, um weltweit umgesetzt werden zu können.“²⁸ Für den hier beschriebenen Bereich der Geoinformationen ist die TC²⁹211 „Geographic information/Geomatics“ verantwortlich, welche unter anderem folgende Standards entwickelt hat:

- ISO 19101 Geographic information - Reference model: Dieser Standard definiert den Rahmen für Standardisierungen in dem Bereich geografischer Informationen und beschreibt die wesentlichen Prinzipien dieser. Hier wird der Bereich weiterer Standardisierungsaktivitäten abgesteckt und die Vorgehensweise beschrieben
- ISO 19107 Geographic Information - Spatial Schema: Modellierung von Raumbezugsformen und Raumbezugsgrundformen (z.B. Punkt, Linie, Fläche oder Knoten, Kante, Masche)
- ISO 19109 Geographic Information - Rules for application schema: Regeln für die Erstellung von Anwendungsschemata
- ISO 19119 Geographic Information - Services: Beschreibt Architektur-Muster für Serviceschnittstellen im Bereich von Geoservices sowie eine Taxonomie für Geoservices. Sie setzt weiterhin fest, wie eine plattformneutrale Servicespezifikation zu gestalten ist und wie konforme, plattformspezifische Servicespezifikationen daraus abgeleitet werden
- ISO 19136 Geographic Information - Geography Markup Language (GML): Eine nähere Beschreibung hierzu folgt auf der nächsten Seite

OGC

Das Open Geospatial Consortium (OGC) ist „eine gemeinnützige, internationale, auf freiwilliger Basis basierende Standardisierungs-Organisation, die die Entwicklung von Standards für den Bereich von räumlichen und ortsbezogenen Diensten leitet.“³⁰

Da das OGC und die ISO/TC211 in den gleichen Bereichen arbeiten, besteht sinnvollerweise seit der Gründung des OGC 1994 eine enge Zusammenarbeit beider Gremien, welche 1999 schließlich in einem Kooperationsvertrag mündete. Hierdurch wurden die formalen Prozesse vereinfacht, um OGC Spezifikationen als internationale ISO Standards anzuerkennen. Das OGC arbeitet dabei eher technologiebezogen, um möglichst zeitnah technische Spezifikationen zu erarbeiten, während die Standards der ISO/TC 211 abstrakt gehalten sind.

Das Hauptziel des OGC liegt darin, eine Reihe von Spezifikationen bereitzustellen, damit aus verschiedenen Geo-Services eine funktionierende Architektur aufgebaut werden kann, entsprechend der oben aufgeführten

²⁷International Organization for Standardization

²⁸<http://www.iso.org/iso/en/aboutiso/introduction/index.html>

²⁹Technical Commission

³⁰<http://www.opengeospatial.org/>

„Web Services Architecture“ (vgl. Unterabschnitt 2.2.2). Das OGC (2004) nennt verschiedene Gründe, warum bislang die Integration räumlicher Informationen verschiedener Quellen sowie räumlicher Informationen in nicht-räumliche Informationssysteme schwierig war:

- Unterschiedliche Typen von Geo-Systemen (Vektor-GIS, Raster-GIS, Aufnahmesysteme, CAD, Transportsysteme und Navigationssysteme) produzieren sehr verschiedene Datentypen
- Verschiedene Bearbeitungssysteme der Datenanbieter benutzen und produzieren interne Datenformate
- Verschiedene Datenanbieter nutzen proprietäre Softwarebibliotheken mit proprietären Interfaces, welche die Möglichkeiten für eine Interprozesskommunikation (IPC) im Netzwerk einschränken
- Unterschiedliche Datenlieferanten, auch wenn diese dasselbe System benutzen, benennen räumliche Objekte unterschiedlich
- Metadaten werden nicht in Standardschemen strukturiert

Durch die Nutzung der Eigenschaften einer WSA wird es möglich, diese technischen und semantischen Probleme zur Interoperabilität zu lösen. Daher sieht es das OGC als wichtig an, die räumliche Komponente durch Schaffung von Spezifikationen in diesen Kontext der Webservices zu integrieren.

Auf diese Art und Weise soll also die Interoperabilität von Geoservices und -anwendern sichergestellt sein. In der OGC Webservice Initiative werden Interfaces für räumliche Daten und Services entworfen, um eine funktionierende Infrastruktur zu erhalten. „Webservices bieten einen neutralen, interoperablen Rahmen zur webbasierten Ermittlung, zum Abruf, zur Integration, Analyse, Ausnutzung und Visualisierung multipler Online-Geodatenquellen, von Sensoren ermittelten Informationen und Geoprozessierungskapazitäten. OGC Webservices werden es zukünftigen Applikationen ermöglichen, nach Bedarf von über das Web verfügbaren Geoprozessierungs- und ortsbezogenen Diensten aufgerufen zu werden“ (Doyle & Reed 2001).

Grundsätzlich lassen sich die entwickelten Spezifikationen aufteilen auf die drei Bereiche *Datenservices*, *Präsentationsservices* sowie die *Verzeichnisdienste*. Verzeichnisdienste eignen sich zur Lokalisierung von Geoservices und -informationen sowie zur Beschreibung dieser Dienste. Datenservices bieten räumliche Daten sowie Operationen zur Prozessierung dieser. Präsentationsdienste bereiten Geoinformationen auf, um diese dem Nutzer in geeigneter Weise darzustellen. Zusätzlich existieren Portaldienste, welche einen einheitlichen Zugriff auf alle zuvor genannten Services bieten. Das OGC hat für jeden dieser Bereiche entsprechende Spezifikationen erarbeitet, um Serviceanbietern einen Rahmen an Standards an die Hand zu geben, mit deren Hilfe eine funktionierende Servicearchitektur sichergestellt wird. Im Folgenden soll eine für die vorliegende Arbeit relevante Auswahl dieser Spezifikationen vorgestellt werden.

Simple Feature Specification (SFS) Die SFS (OGC 1999) legt einen Standard zur Definition und Beschreibung von räumlichen Primitiven wie Punkten, Linien oder Polygonen fest.

Geography Markup Language (GML) Die GML (Cox et al. 2004) stellt eine auf XML basierende Spezifikation zur Modellierung, Speicherung und Übertragung räumlicher Daten dar. Dabei werden die räumlichen Daten aufgeteilt in einzelne Objekte (Features) mitsamt ihren Attributen, Relationen und Geometrien. Die GML bietet eine Vielzahl an Objekten zur Beschreibung der Geografie, wie Features, Koordinatenreferenzsysteme, Geometrie, Topologie, Zeit- und Maßeinheiten. GML wurde vom OGC entwickelt und wird mit der Version 3.1.1 inzwischen auch auf die Internationalen Normen der ISO 191xx - Serie zurückgeführt. Die OGC-Spezifikation für GML 3.1.1 ist identisch mit ISO 19136 und damit auch selbst eine offizielle Internationale Norm der ISO.

OGC Web Services (OWS) common Das OGC hat zunächst durch die OWS-common Spezifikation (Whiteside 2005b) eine allgemeingültige Beschreibung aller folgenden Implementierungs-Spezifikationen wie WMS³¹, WFS³² oder WCS³³ festgesetzt. Diese enthält gemeinsame Aspekte wie Parameter oder Datenstrukturen, welche in den jeweiligen Interface-Spezifikationen detaillierter festgelegt werden sollen, durch diese jedoch nicht wiederholt beschrieben werden müssen. Hierzu gehört beispielsweise die Funktion `<GetCapabilities>`, welche es ermöglicht, die Funktionalitäten und Inhalte eines Dienstes abzufragen. Der Dienst wird somit selbstbeschreibend.

Web Feature Service (WFS) Der WFS (OGC 2005c) fungiert, im Gegensatz zum weiter unten beschriebenen WMS, als Lieferant räumlicher Daten. Die Implementierung eines WFS stellt einen Webservice dar, welcher, entsprechend der oben genannten Server-Funktion, innerhalb einer Client-Server Architektur die Anfrage eines Client entgegennimmt und seinerseits auf gespeicherte Geoinformationen zurückgreift, um dem Client die gewünschten Informationen zur Verfügung zu stellen. Innerhalb dieser Architektur ist herauszustellen, dass die

³¹Web Map Services Implementation Specification

³²Web Feature Services Implementation Specification

³³Web Coverage Services Implementation Specification

Datenschicht für den Client unsichtbar bleibt und dieser lediglich mit dem WFS kommuniziert. Der WFS stellt Operationen zur Beschreibung des Dienstes sowie zur gezielten Abfrage der Daten bereit. So ermöglicht die Operation `<GetCapabilities>` die Abfrage eines die Metadaten des Dienstes beschreibendes XML-Dokument. `<DescribeFeatureType>` liefert ein Dokument, welches einen spezifischen Objekttypen der Datenbasis, dessen Schema und Attribute beschreibt. Über `<GetFeature>` kann der Client dann bestimmte Objekte abfragen und erhält als Antwort auf seine Anfrage ein GML-Dokument, welches die geforderten Geodaten enthält. Die Abbildung 2.3 verdeutlicht folgenden, typischen Kommunikationsverlauf zwischen einem Client und dem WFS:

1. Eine Nutzerapplikation fordert zunächst ein Dokument zur Beschreibung der Leistungen des WFS. Dieses Dokument enthält eine Beschreibung aller Operationen, welche vom WFS unterstützt werden sowie eine Liste aller Objekttypen (Featuretypes), die dieser bereitstellen kann
2. Eine Nutzerapplikation stellt (optional) eine Anfrage an den WFS zur Definition einer oder mehrerer Objekt- oder Elementtypen, die dieser bereitstellt
3. Basierend auf dieser Definition der Objekttypen generiert die Anwendung eine Anfrage oder Transaktion entsprechend der OGC WFS-Spezifikationen und schickt diese Anfrage an den Webserver
4. Der WFS wird aufgerufen, um die Anfrage zu lesen und zu bedienen
5. Die resultierenden Daten werden in Form eines GML-Dokuments an den Nutzer gesendet. Hat der WFS die Bearbeitung der Anfrage abgeschlossen, wird zusätzlich ein Statusreport generiert und zum Client gesendet

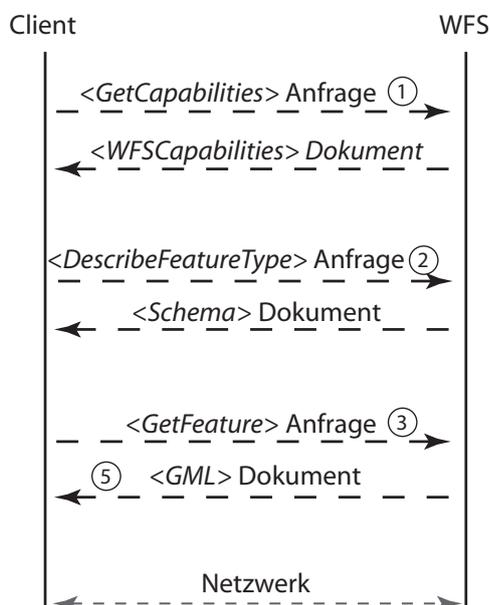


Abb. 2.3: Protokoll-Diagramm eines WFS (nach OGC 2005c)

Web Map Service (WMS) Im Gegensatz zum WFS liefert der WMS (OGC 2006) auf Anfrage ein digitales Bild einer Karte im Vektor- (SVG³⁴) oder Rasterformat (PNG³⁵, GIF³⁶, TIFF³⁷ oder JPEG³⁸). Hier wird das Verhalten eines Services spezifiziert, welcher Karten dynamisch aus Geoinformationen generiert und bereitstellt. Dabei werden Operationen zur Abfrage von Beschreibungen der Karten, zur Abfrage der Karten selbst sowie der Charakterisierung von Objekten, die in der Karte dargestellt sind, festgesetzt. Somit steht hierdurch eine Festlegung bereit, welche Webnutzern den Zugriff auf von Mapservern generierte Karten ermöglicht. Der Anwender kann dabei den Kartenausschnitt, -inhalt und das Koordinaten-Referenzsystem festlegen. Gleichzeitig können durch einen Transparenzmodus mehrere Karten kombiniert werden. Der Nutzer gibt zudem die Größe des resultierenden Bildes vor und kann durch sogenannte Stylesheets, also vordefinierte Muster, das Aussehen der Objekte und somit das Kartenbild beeinflussen. Im Jahre 2004 wurde von der ISO ein internationaler Standard (ISO 19128) basierend auf der WMS Spezifikation des OGC verabschiedet.

³⁴Scalable Vector Graphics

³⁵Portable Network Graphics

³⁶Graphics Interchange Format

³⁷Tagged Image File Format

³⁸Joint Photographic Expert Group Format

Web Coverage Service (WCS) Die WCS Specification (OGC 2003b) ist ähnlich dem WFS und dem WMS zu sehen. Laut OGC (2003b) besteht die Aufgabe des Web Coverage Service in der Beschreibung und Anforderung von mehrdimensionalen, zusammenhängenden Rasterdaten, die die Erdoberfläche beschreiben. Dieser ermöglicht den Zugriff auf georeferenzierte Bilddaten wie PNG, GIF, JPEG sowie GeoTIFF.

Web Gazetteer Service (WGS) Die WGS Specification (Atkinson 2001) erweitert den WFS durch eine Schnittstellendefinitionen zu einem Gazetteer, d.h. einem Index geografischer Namen einschließlich der Lage.

Web Registry Server (WRS) und OpenGIS Catalogue Service Der WRS (Nebert & Whiteside 2005) dient als Service-Broker zur Ermittlung und Registrierung von Geoservices, ähnlich den Funktionalitäten des UDDI.

Web Processing Service (WPS) Der WPS (Schut & Whiteside 2005) spezifiziert eine Schnittstelle zu einem Prozessierungsservice. Ein WPS kann seinem Clients jegliche GIS Funktionalitäten über ein Netzwerk anbieten. Das Interface bietet Mechanismen zur Lokalisierung der zu prozessierenden Daten, löst die Prozessierung aus und regelt die Ausgabe der Ergebnisse, so dass diese vom Nutzer abgerufen werden können.

Neben diesen existieren eine Reihe weiterer Spezifikationen des OGC, welche alle dem Ziel dienen, einen integrativen Rahmen zur Gestaltung von Webservices für räumliche Daten zu bilden.

2.3.2 Die OGC Web Services Architektur

Die oben beschriebenen Servicespezifikationen sind allesamt Teile eines Konzeptes des OGC zum Aufbau einer allgemeingültigen Architektur für Geo-Webdienste. Diese Architektur wird vom OGC als OWS³⁹-Architektur betitelt (Lieberman 2003). Das Konzept der OWS dient als Rahmen zum Aufbau eines Netzwerkes aus Geoprozessierungs-Applikationen oder um Geoprozessierungs-Funktionalitäten in andere Informationsanwendungen zu integrieren. Abbildung 2.4 zeigt die Form der OWS, wie das OGC sie verstanden haben möchte. Die OWS-Architektur bietet Entwicklern und Anbietern von Webdiensten die Möglichkeit, eigene Servicekomponenten zu entwickeln, welche wiederum auf die OGC Webservicekomponenten zurückgreifen können. Diese werden dann publiziert und von anderen Nutzern über das Internet durch Metadaten lokalisiert und aufgerufen. Dabei sind alle Dienste im System eigenständig, selbstbeschreibend und modular aufgebaut. Der Anwender muss sich keine Gedanken darüber machen, wie die einzelnen Komponenten implementiert sind, er muss sie nur noch benutzen. Der Dienst kann dabei entweder über einen Desktop-PC oder aber über mobile Geräte aufgerufen werden, da das gesamte System plattformneutral aufgebaut ist.

Diese Architektur basiert auf der in dem Unterabschnitt 2.2.2 beschriebenen, allgemeingültigen serviceorientierten Architektur (SOA), deren Grundlage Dienstanbieter und -konsumenten innerhalb eines verteilten Systemes bilden. Interaktionen zwischen den einzelnen Komponenten finden in Form von Serviceanfragen, -antworten und Fehlerbehandlungen statt (Lieberman 2003). Die OGC Webservice Strategie nutzt dagegen nicht zwingend die im Unterabschnitt 2.2.2 beschriebenen Komponenten einer WSA, wie SOAP oder WSDL. In einem aktuellen Diskussionsbeitrag des OGC (Sonnet 2004) wird deswegen die Unterstützung der offenen Standards UDDI, WSDL, SOAP innerhalb der OGC-Webservice-Spezifikationen gefordert und spezifiziert.

Dabei müssen eine Reihe unterschiedlicher Services zusammenarbeiten, um letztendlich ein sinnvolles System zum „Publizieren, Finden und Einbinden“ zu erhalten. Daher muss die grundlegende Architektur der OWS Daten-, Registrierungs-, Präsentations- und Prozessierungsdienste, Austauschformate sowie die Nutzerapplikation berücksichtigen. Jeder Dienst stellt eine Sammlung von Operationen dar, welche durch eine Schnittstelle verfügbar sind. Der Nutzer ruft den jeweiligen Dienst auf und übergibt diesem bestimmte Parameter, worauf dieser ein entsprechendes Resultat generiert, welches dem Anwender oder der Applikation bereitgestellt wird. Die OWS Initiative entwickelt hierzu Schnittstellen für räumliche Daten und Services sowie zur Definition von Metadaten, wie sie im vorangegangenen Abschnitt vorgestellt wurden, damit diese Architektur in einem verteilten System zur Prozessierung von Geodaten einzusetzen ist (OGC 2004). Das System des OWS bietet eine Sammlung von Interfaces und Beschreibungssprachen, um die funktionalen Elemente eines Dienstes, wie er in Abbildung 2.4 dargestellt ist, zu realisieren. Dieses System besteht aus Anwendern, dem eigentlichen Netzwerk sowie einem Bereich von Geo-Diensten und Datenanbietern.

Damit die Kommunikation zwischen den einzelnen Komponenten funktioniert, müssen die Schnittstellen zwischen diesen spezifiziert werden. Die Implementierung der einzelnen Server und Clients dagegen ist dem Entwickler überlassen und ist in keiner Weise reglementiert, so lange der Computer, auf dem der Service läuft, mit dem Internet verbunden ist. Die Servicearchitektur dagegen ist hard- und softwareneutral.

³⁹OGC Web Services

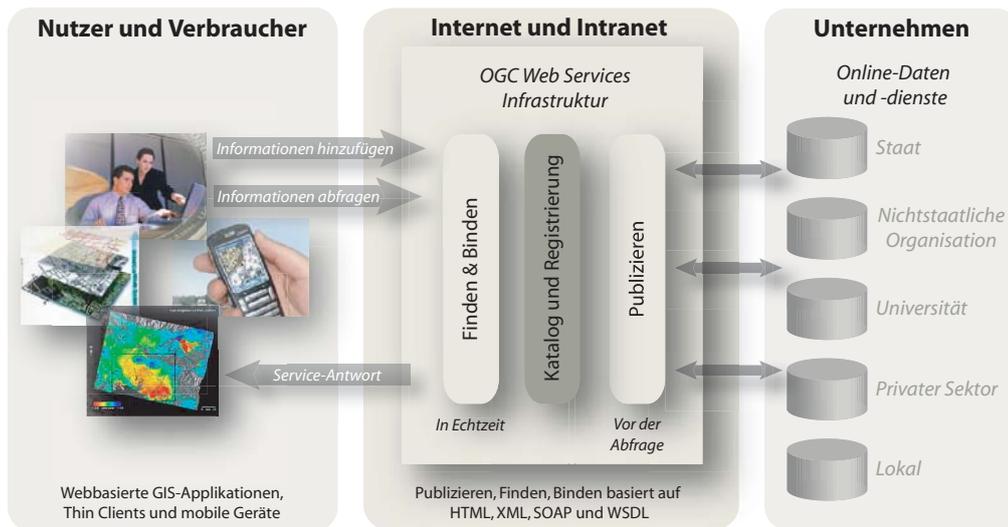


Abb. 2.4: OGC Webservices Architektur (nach OGC 2004)

Dabei sind nach Whiteside (2005b) folgende Eigenschaften signifikant für den OWS:

- Servicekomponenten sind in verschiedenen Ebenen organisiert
- Dienste werden oftmals verkettet
- Service-Schnittstellen beruhen auf offenen Standards und sind relativ einfach gehalten
- Die Service-Kommunikation beruht auf offenen Internet-Standards (z.B. HTML, XML)
- Server- und Client-Implementierungen sind unabhängig und somit hard- und softwareneutral

Whiteside (2005b) liefert eine ausführliche Beschreibung dieser Eigenschaften, die im Folgenden zusammenfassend dargelegt werden sollen.

Service-Ebenen

Services beziehungsweise Service-Instanzen können in vier Ebenen aufgeteilt werden (vgl. Abbildung 2.5):

- Anwender (Client)
- Applikationsdienste
- Prozessierungsdienste
- Informationsmanagement-Dienste

Die Aufteilung der Ebenen ist dabei weniger streng, so dass nicht notwendigerweise alle Ebenen implementiert sein müssen. Einzelne Ebenen können ohne Weiteres übersprungen werden. Entsprechend ihrem Titel enthält jede Ebene spezifische Dienste. Mit Ausnahme der Anwenderebene bietet jede Schicht ihre Dienste für die Services der übrigen Ebenen. Des Weiteren können Services andere Services derselben Ebene nutzen.

Informationsmanagement-Service-Ebene: Innerhalb dieser Ebene befinden sich Dienste zur Speicherung der Daten sowie zur Regelung des Zugriffs auf diese. Dabei verwaltet jede Serviceinstanz in der Regel mehrere Datensätze. Darüber hinaus können Metadaten gespeichert und zur Suche herangezogen werden. Die Zugriffsfunktionalitäten ermöglichen die Abfrage einer anwenderspezifischen Untermenge der Daten. Einige Beispiele für Services innerhalb dieser Ebene finden sich in der Tabelle 2.3.2.

Prozessierungs-Service-Ebene: Diese Ebene enthält Dienste zur Prozessierung der Daten. Diese werden von Anwendern oder Diensten der Applikations-Ebene aufgerufen und können ihrerseits auf andere Dienste derselben Ebenen sowie der Informationsmanagement-Ebene zurückgreifen. Tabelle 2.2 listet beispielhaft einige für diese Ebene spezifische Services auf.

Applikations-Service-Ebene: Die Applikationsebene enthält Dienste, welche die Nutzerebene und vor allen Dingen die sogenannten „thin clients“ unterstützen sollen. Die Services können also von der Client-Anwendung direkt genutzt werden, anstatt dass diese von der Applikation des Nutzers bereitgestellt werden muss. Die

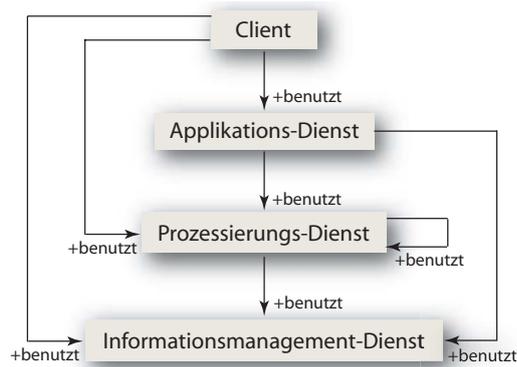


Abb. 2.5: Service-Ebenen der OWS Architektur (nach Whiteside 2005a)

<i>Name des Dienstes</i>	<i>Beschreibung des Dienstes</i>
Web Map Service (WMS)	Generiert aus einem oder mehreren vom Anwender ausgewählten geografischen Datensätzen eine räumlich referenzierte Karte, basierend auf einem vom Anwender definierten Rechteck
Web Feature Service (WFS)	Liefert Objekte (Features) auf Grund der vom Nutzer definierten Auswahlkriterien
Web Coverage Service (WCS)	Liefert eine vom Anwender definierte Untermenge eines von diesem definierten Datensatzes zur Beschreibung der Erdoberfläche
Gazetteer Service	Liefert die Lagegeometrie für vom Anwender spezifizierte geografische Namen

Tab. 2.1: Auswahl einiger spezifischer Dienste der Informationsmanagement-Ebene

<i>Name des Dienstes</i>	<i>Beschreibung des Dienstes</i>
Web Processing Service (WPS)	Bietet GIS-Funktionalitäten und prozessiert Daten, die über das Netzwerk gesendet werden oder über einen Server erreichbar sind (z.B. Generalisierungsservice)
Web Terrain Service (WTS)	Generiert anwenderspezifische, perspektivische Ansichten geografischer Daten und liefert gerenderte Bilder in Bild- oder Grafikformaten
Web Coordinate Transformation Service (WCTS)	Transformiert die Koordinaten von Objektdaten von einem Referenzsystem in ein anderes
Geocoder Service	Dienst, um ortsbezogene Texte mit Positionskoordinaten anzureichern
Routingsservice	Bestimmt die optimale Route zwischen zwei Punkten, basierend auf Eingabeparametern

Tab. 2.2: Auswahl einiger spezifischer Dienste der Prozessierungs-Ebene

Dienste in dieser Ebene werden somit vom Client genutzt und greifen ihrerseits auf Dienste der Prozessierungs-, Informationsmanagement- und auch der Applikationsebene zurück. Beispiele für Applikationsdienste sind in der Tabelle 2.3 aufgelistet.

<i>Name des Dienstes</i>	<i>Beschreibung des Dienstes</i>
Web portal services	Dienste, welche es einem Anwender erlauben, mit verschiedenen Applikationsdiensten für unterschiedliche Datentypen und Zwecke zu interagieren
WMS application services	Dienste, welche es einem Nutzer erlauben, mit einem WMS zu interagieren, um entsprechende Daten zu finden und auszuprägen; z.B. über ein GUI ⁴⁰
Gazetteer application services	Dienste, welche es einem Nutzer erlauben, mit einem geografischen Namensverzeichnis zu interagieren, z.B. über Suchmasken

Tab. 2.3: Auswahl einiger spezifischer Dienste der Applikations-Ebene

Client-Ebene: Clients sind Softwarepakete, zu denen ein menschlicher Anwender Zugang besitzt. Diese werden entsprechend ihrem Umfang meist kategorisiert in „thin“ (zum Beispiel Webbrowser), „thick“ (zum Beispiel umfassende GIS) oder „chubby“ (als Zwischenkategorie) Clients.

Verkettung der Dienste

Einige Dienste können erst dann sinnvoll genutzt werden, wenn diese mit anderen Services zusammenarbeiten. Somit unterstützt auch die OWS die Verkettung einzelner Dienste, wobei diese nicht nur linear, sondern vielmehr auch in Form einzelner Netzwerke miteinander verknüpft werden können. Grundsätzlich verwendet ein Dienst die Ergebnisse seines Vorgängers in der Kette als Input für seinen eigenen Service.

Neben der Verkettung ist auch eine Anhäufung von Diensten denkbar. Dies ist der Fall, wenn zwei oder mehr Dienste sich auf denselben Inhalt beziehen. So können zum Beispiel ein WFS und ein WMS eines Anbieters auf denselben Datensatz zurückgreifen (Lieberman 2003).

Schließlich ist auch eine Verschmelzung einzelner Dienste denkbar. So können Schnittstellen, welche von mehreren Services getrennt implementiert wurden, nun in einem einzigen Dienst vereint werden. Die Registrierungseinheit eines Dienstes mit seiner *GetCapabilities*-Funktionalität zur Abfrage der Fähigkeiten einzelner Dienste würde in dem neuen Service beispielsweise nur einmal implementiert werden (Lieberman 2003).

Insgesamt geht es bei der Verkettung von Diensten um nicht mehr als die Schnittstellen der einzelnen Dienste. Es ist lediglich wichtig zu wissen, wie die Fähigkeiten eines Dienstes, dessen Inhalte sowie die Metadaten abgefragt werden.

Service-Schnittstellen

Das OGC (Lieberman 2003) definiert Services und Schnittstellen wie folgt:

- Service: Bestimmter Teil der Funktionalität, welcher durch eine Instanz über Schnittstellen angeboten wird
- Schnittstelle: Benannter Satz von Operationen, welcher das Verhalten einer Instanz charakterisiert
- Operation: Spezifikation einer Transformation oder Anfrage, die von einem hierzu aufgerufenen Objekt ausgeführt wird. Sie hat einen Namen und eine Liste von Eingabe- und Ausgabeparametern

Die im OWS verwendeten Interfaces basieren auf offenen Standard-Schnittstellen, oftmals Service-Schnittstellen des OGC, und sind einfach gehalten. Dabei verwenden die benannten Dienste jeweils nur ein Interface. Die OWS Schnittstellen teilen sich gemeinsame Elemente an Stellen, an denen es praktisch erscheint, so dass diese nur einmal implementiert werden müssen. So besitzen zum Beispiel alle Services eine *GetCapabilities*-Operation zur Abfrage der Metadaten. Tabelle 2.4 zeigt auszugsweise einige Formate und Sprachen zum Datenaustausch. Dabei ist der Zugriff auf einen Geoservice mit jedem internetfähigen Gerät möglich. Somit haben Desktop-Computer, aber auch mobile Geräte wie PDA⁴¹ oder Mobiltelefone die Möglichkeit, eine große Anzahl von Geoinformationsquellen und -diensten nutzen zu können.

⁴¹Personal Digital Assistant

<i>Name des Dienstes</i>	<i>Beschreibung des Dienstes</i>
Filter Encoding (FE)	Kodiert Anfragen zu Objekten oder anderen Daten, welche bestimmte Bedingungen erfüllen, wie z.B. Attributwerte
Styled Layer Descriptor (SLD)	Kodiert die vom Anwender spezifizierte kartografische Ausprägung von Objekten und Oberflächen (Bildern), wie z.B. Linienfarben und Stärken
Geography Markup Language (GML)	Auf bestimmten XML-Schemata basierende Sprache zum Austausch von Geometrien
Web Map Context	Kodiert den Inhalt einer WMS Darstellung

Tab. 2.4: Auswahl einiger spezifischer Kodierungsformate und Sprachen zum Datenaustausch

Kommunikation zwischen den Diensten und Komponenten

Bei der oben beschriebenen Verkettung von Diensten oder beim Aufruf eines Dienstes durch einen Client werden Daten, wie Abfrageparameter und die resultierenden Datensätze, über das Internet ausgetauscht. Zum Datenaustausch über das Internet existieren verschiedene Dienste und Protokolle. Die OWS nutzt hierzu Standard-WWW-Protokolle wie *HTTP GET*, *HTTP POST* sowie *SOAP*. Einzelne Operationen bestimmter Server werden durch eindeutige URLs⁴² verortet. Schließlich werden MIME⁴³ Typen benutzt, um Datenübertragungsformate zu identifizieren. Die übertragenen Daten selbst sind in den meisten Fällen XML-kodiert, wobei deren Inhalte und Formate in XML-Schemata (XSD⁴⁴) definiert sind.

Den Rahmen für Geo-Webservices bilden somit die vom OGC entwickelten Spezifikationen und Referenzimplementierungen, wie sie im vorangegangenen Abschnitt beschrieben sind. Die OWS ermöglichen Geodiensten über das Internet per XML und HTML miteinander zu kommunizieren.

2.3.3 Infrastrukturen und Anwendungsbereiche von Geo-Webdiensten

Geodaten oder weiter gefasst Geoinformationssysteme (GIS) werden mehr und mehr zum Kern computergestützter Anwendungen. GIS können helfen, Informationen und Daten aus den verschiedensten Quellen über ihre Lage miteinander in Beziehung zu setzen. Dies wird weitaus effektiver realisiert durch eine technische Umgebung mit einem freien und vor allen Dingen nahtlosen Zugriff auf die vorhandenen Daten (Davis & Alves 2005). Die derzeitige Praxis erfordert jedoch, verschiedenste Probleme im Austausch der Daten zu lösen, wie Speicherformate, Datenqualitäten, verschiedene Inhalte, kartografische Projektionen oder Datenstrukturen (Rajabifard & Williamson 2001). Teege (2001) nennt drei Haupthindernisse für einen funktionierenden Geodatenmarkt:

- Dem potenziellen Datennutzer fehlt die Übersicht über das Datenangebot
- Möchte der Datennutzer auf Geodaten unterschiedlicher Quellen zugreifen [...], so stößt er wegen der Heterogenität der Daten auf erhebliche Schwierigkeiten
- Hat der Nutzer einmal die Daten von einem bestimmten Anbieter bezogen, bleibt meist ein großer Aufwand, die Daten aktuell zu halten [...]

Dabei ist es wichtig, dass vorhandene Daten mehrfach genutzt werden können, denn die Datenerfassung ist in den meisten Fällen eine schwierige und vor allen Dingen kostenintensive Prozedur. Als Lösung dienen oftmals Werkzeuge zur Übersetzung verschiedener Datenformate zum Austausch der Daten. Ein erster Schritt aus diesem Dilemma sind daher einheitliche Datenformate. Oft verwendete, proprietäre Datenformate haben sich in vielen Fällen als Quasistandards etabliert, wie beispielsweise das *Shapefile*-Format (.shp) der Firma *ESRI*. Diese Ansätze erfassen lediglich syntaktische Belange, nicht jedoch die semantischen Probleme. Außerdem sind solche Datenformate ungeeignet für den Online-Zugriff, da entsprechende Zugriffs-Schnittstellen, Mechanismen für eine parallele Bearbeitung oder auch Such- und Manipulationsoperationen fehlen. Dies führt zu einem Bedarf an einem offline Import-Export-Kreislauf und zu mehrfachen Kopien der Daten, was wiederum Synchronisationsprobleme nach sich zieht (Davis & Alves 2005).

Die derzeitige Praxis bei der Verwendung von Geodaten sieht eine offline-Nutzung der Daten vor. Der Anwender erhält eine Kopie des Datenbestandes eines Anbieters und speichert diese Momentaufnahme in seinem System. In

⁴²Uniform Resource Locators

⁴³Multipurpose Internet Mail Extensions

⁴⁴XML Schema Definition

regelmäßigen Abständen spielt er neue Datensätze hinzu oder erneuert alte Bestände. Eine Alternative, welche in Zukunft praktikabel sein wird, ist der Online-Zugriff auf die Datenbestände eines oder mehrerer Anbieter über das Internet. Dies wäre gleichzeitig ein Schritt in Richtung der Mehrfachnutzung von Geodaten ohne eine Redundanz in den Datenbeständen zu erhalten. Es existieren bereits verschiedene Ansätze zur Realisierung dieses Konzeptes. Dabei helfen die in den vorangegangenen Abschnitten beschriebenen Spezifikationen zum Datenaustausch über die verschiedenen Geo-Webdienste. Es gilt nun, diese Dienste innerhalb einer entsprechenden Infrastruktur einzusetzen. Dabei sind einfache, aber auch komplexe, allumfassende System realisierbar. Der einfachste Fall wäre z.B. die Schaffung von Online-Zugriffspunkten auf die Daten der einzelnen Anbieter, realisiert durch den jeweiligen Betreiber. Die verschiedenen Möglichkeiten zur Nutzung verfügbarer Geodatenbestände und Geodienste mit Hilfe von Webservices sollen im Folgenden aufgezeigt werden.

Geodaten Warehouse

Einen Schritt weiter als der bloße Zugriff auf einen bestimmten Datensatz über eine spezifizierte Schnittstelle (z.B. WMS oder WFS) geht hier das Konzept der sogenannten *Warehouses*⁴⁵. Das *Warehouse* integriert die Datenbestände mehrerer Anbieter in einem einheitlichen System (siehe Abbildung 2.6). Hier entsteht also der Vorteil, dass der Zugriff auf verschiedene Daten gleichzeitig möglich ist. Somit können die Daten auch für Vergleiche oder Analysen herangezogen werden. Die Schwierigkeit liegt vor allen Dingen in der Integration und Aktualisierung größerer Datenmengen. Das hier aufgezeigte Problem findet seine Lösung auf dem Gebiet der MRDB (vgl. Kapitel 3), denn genau diese Fragestellungen zur Integration mehrerer Datenbestände werden hier bearbeitet und sollen im Rahmen der vorliegenden Arbeit vertieft werden.

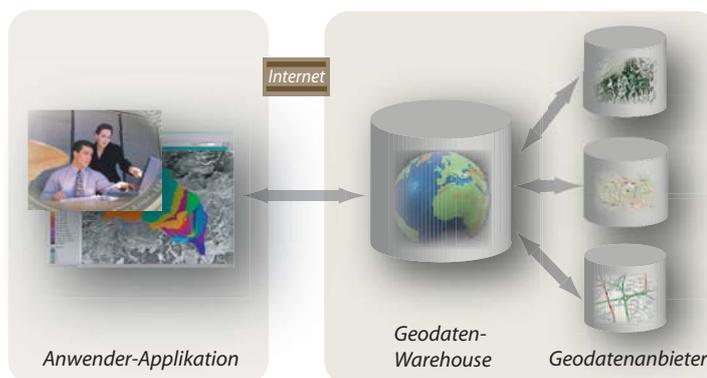


Abb. 2.6: Typische Datawarehouse Architektur (nach Teege 2001)

Geodaten Clearinghouse

Ein weiterer Schritt hin zu verteilten Geodaten wurde durch die Entwicklung von sogenannten *Clearinghouses*⁴⁶ für räumliche Daten getätigt. Diese internetbasierten Komponenten sollen den Zugriff auf räumliche Daten erleichtern mit Hilfe einer zentralen Seite, auf der Daten aus verschiedensten Quellen gefunden werden können (siehe Abbildung 2.7). Sie bieten darüber hinaus ergänzende Leistungen zum Suchen, Verbreiten, Betrachten, Bestellen und Übertragen von Daten (Crompvoets et al. 2004) und verfügen außerdem über Metadaten, welche die eigentlichen Geodaten beschreiben. Clearinghouses beschreiben die Daten selbst sowie deren Nutzungsbedingungen und Bezugsmöglichkeiten. Andere, neuere Definitionen stellen den Servicegedanken in den Vordergrund und weisen dadurch starke Ähnlichkeiten zu Webportalen (s.u.) auf. Smits et al. (2002) definieren ein Clearinghouse synonym zu Katalogdiensten: „Clearinghouses sind Services, welche Nutzern von Anwendungssoftware helfen, Informationen zu finden, welche irgendwo in einem verteilten System existieren.“ Sie halten also nicht die Daten selbst vor, sondern verweisen auf Quellen zum Abruf dieser. Clearinghouses decken den Servicebereich der Datensuche ab und werden oftmals synonym verwendet mit „Catalogue Services“ oder „Spatial Data Directories“.

⁴⁵engl. warehouse = (hier) Lagerhalle; nicht Warenhaus, wie in vielen Publikationen zu finden (Teege 2001)

⁴⁶Unter clearinghouse (deutsch: „Clearingstelle“) versteht man aufgearbeitete Zusammenstellungen verschiedener (Internet-) Quellen, die für ein bestimmtes Fachgebiet oder für mehrere Gebiete als Suchhilfen zusammengestellt werden (Beifuss et al. 2001)

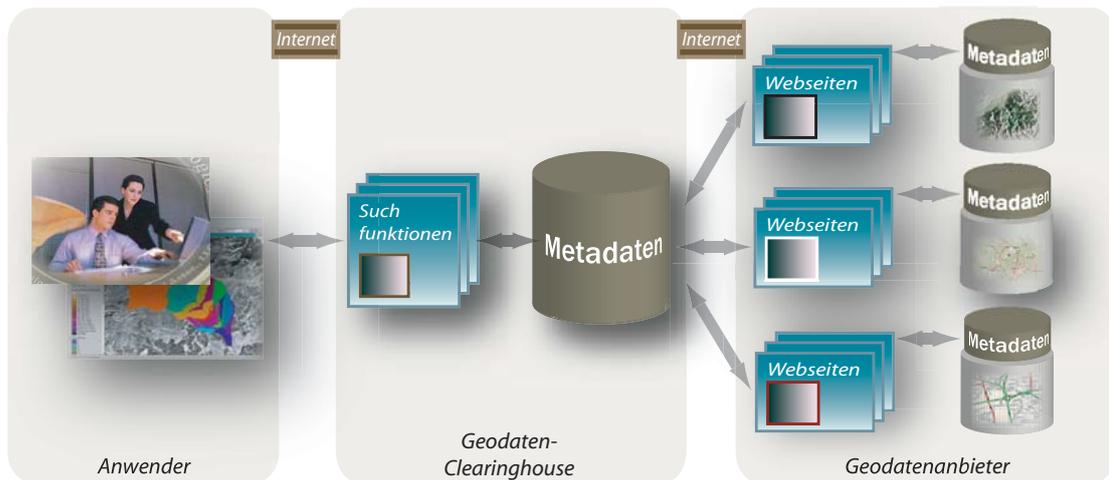


Abb. 2.7: Typische Data-Clearinghouse Architektur (nach Teege 2001)

Clearinghouses muten zunächst per Definition ähnlich dem weiter unten beschriebenen Geoportal an. Charakteristisch ist hier jedoch, im Gegensatz zu den Geoportalen, dass der Metadatenserver keine Nutzungsschnittstelle zu den Geodaten anbietet. Somit hilft ein Metadatenserver nur bei der Suche nach verteilten Datenangeboten.

Geoportale

Ein Portal⁴⁷ bezeichnet in der Architektur ein durch besondere Gestaltung hervorgehobene Eingangstür eines wichtigen Gebäudes. Ein Internetportal stellt synonym hierzu einen Eingang oder Zugang zu einem bestimmten Themenbereich im Internet dar. Teege (2001) definiert ein Internetportal als „Einstiegsportal für das Internet, bei dem ein User all die Informationen und Services zur Verfügung gestellt bekommt, die er benötigt [...]“. Das Ziel ist es, dem Anwender stets die passenden Informationen zur richtigen Zeit, im richtigen Kontext und auch am richtigen Ort zu präsentieren.“ Weiterhin betont Teege die Möglichkeit der Personalisierung, also dem speziellen Zuschnitt der Informationen, Nutzerschnittstellen, Visualisierungen etc. auf eine bestimmte Person, als eine typische Eigenschaft. Geoportale übertragen diese Eigenschaften auf den Bereich von Geodaten; Webseiten also, welche einen Eintrittspunkt zu geografischen Inhalten im Netz darstellen (Tait 2005). Sie ermöglichen eine Abfrage von Metadaten für Daten und Services und bieten gleichzeitig einen Link zu den gewünschten Daten (siehe Abbildung 2.8). Des Weiteren unterstützen sie eine kommerzielle Verbreitung, indem sie auch den Verkauf beziehungsweise Erwerb von Geoinformationen erlauben. Aus technischer Sicht sind Geoportale Webseiten, welche auf eine Datenbank zurückgreifen, die Metainformationen über geografische Daten und Services beinhaltet. Die Services sind ihrerseits als Webservices (vgl. Abschnitt 2.2) implementiert.

Die aktuelle unterscheidet sich von der ersten Generation der Geoportale durch zwei neue technische Möglichkeiten. Zum einen ist es nun möglich, sowohl auf Metadaten, welche den Dienst beschreiben, als auch auf den Service selbst (Kartendienst, Datendownload, Geocodierung, Routing, etc.) zugreifen zu können. Zum anderen können diese Dienste von einem konventionellen Desktop-GIS als auch von einem sogenannten „Thin Client Browser“ aus aufgerufen und genutzt werden (Maguire & Longley 2005). Dabei ist es möglich, verschiedenste Services zu kombinieren.

Es existieren derzeit eine Reihe von Geoportalen, wie das im folgenden Abschnitt erwähnte „Geospatial One-Stop“ der USA oder das „European Geo-Portal“ als Teil des INSPIRE⁴⁸ Projektes (siehe Abbildung 2.9). Dieses enthält Funktionen zum Publizieren von Informationen, zum Suchen, einen Kartendienst, einen Zugriff auf die GDI sowie serverseitige Applikationen, wobei sich der letztgenannte Bereich derzeit (2006) noch im Aufbau befindet.

Geodateninfrastruktur - GDI

Schon in den 80er Jahren entstand, zunächst vor allen Dingen in den USA und Kanada, das Bewusstsein nationaler Vermessungsbehörden und Geodatenanbieter für einen umfassenderen Zugriff auf Geoinformationen.

⁴⁷lat. porta = Tür, Pforte

⁴⁸Infrastructure for Spatial Information in Europe

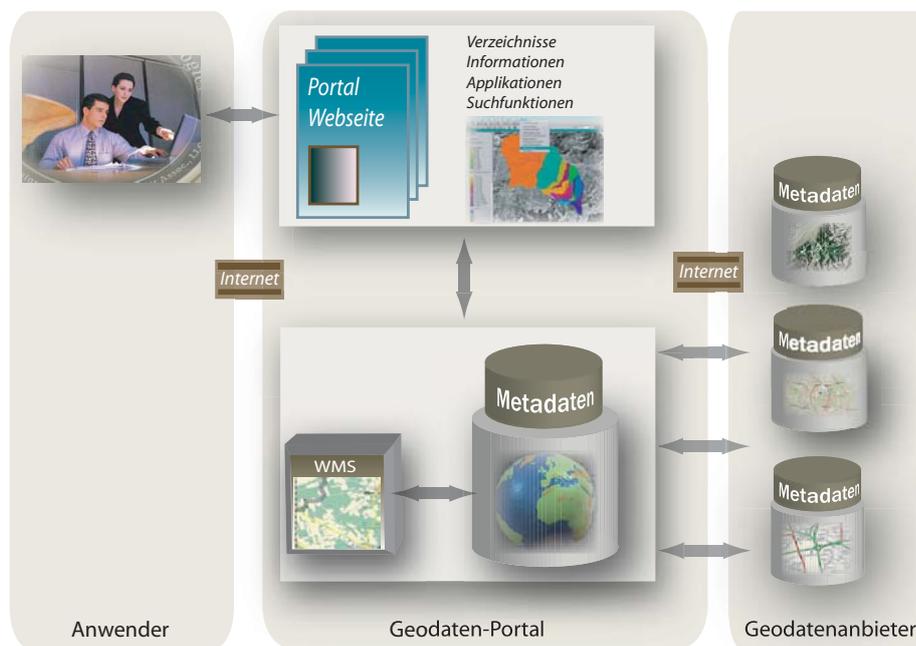


Abb. 2.8: Typische Architektur eines Geoportals (nach Teege 2001)



Abb. 2.9: EU-Geoportal INSPIRE⁴⁹: Es finden sich Publikationsmöglichkeiten, Kartendienste, Suchfunktionen, Applikationen und andere Informationen über das Portal

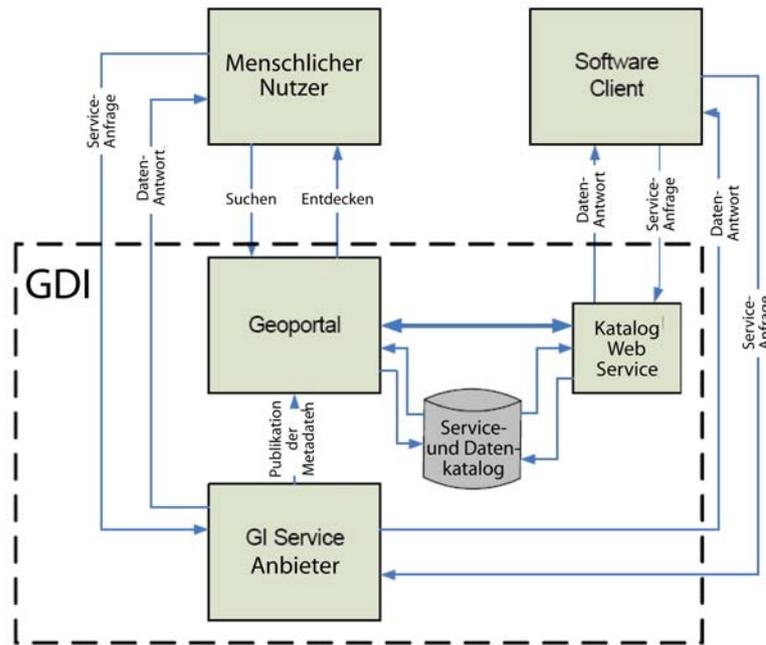


Abb. 2.10: Geoportale und GDI (nach Davis & Alves 2005)

1993 wurde der Begriff SDI⁵⁰ von dem Nationalen US Forschungskonzil geprägt, damals unter anderem in dem Zusammenhang eines standardisierten Zugriffs auf Geoinformationen. Das FDGC⁵¹ definiert SDI als „die Summe aus Technologie, Methodik, Standards, Personal sowie die damit verknüpften Aktivitäten, um räumliche Daten in allen Bereichen der Regierung, dem privaten und nicht-kommerziellen Sektor sowie der Wissenschaft zu beschaffen, prozessieren, verteilen, nutzen, unterhalten und erhalten“ (National Reserach Council 1993). Auch in Deutschland hat man den Wert der Geodaten und ihre Notwendigkeit für die Verwaltung, Wirtschaft und Wissenschaft erkannt und versucht durch eine geeignete GDI⁵² der Mehrfacherfassung von Geodaten entgegenzutreten. Der Bund hat hierzu den IMAGI⁵³ gegründet, dessen Ziel in der Schaffung einer geeigneten GDI für Deutschland zu sehen ist. Auf europäischer Ebene verfolgt das Projekt INSPIRE ähnliche Ziele. Folgende Probleme sollen durch die Umsetzung einer europäischen GDI im INSPIRE-Projekt gelöst werden (vgl. Annoni et al. 2004):

- Lücken in den räumlichen Datenbeständen durch unvollständige oder fehlende Daten
- Mangel an Dokumentationen einschließlich Metadaten, was die Wiederverwendung der Daten einschränkt
- Inkompatibilitäten räumlicher Datensätze, wodurch die Möglichkeit zur Kombination verschiedener Datensätze beschränkt wird
- Inkompatibilität der GIS, wodurch es schwierig wird, räumliche Daten zu finden, auf diese zuzugreifen und wiederzuverwenden
- Das Teilen und Wiederverwenden der Daten wird weiterhin behindert durch kulturelle, institutionelle, finanzielle und rechtliche Schranken

Dabei ist die GDI mehr als ein Geoportale (s.o.). Nach Davis & Alves (2005) besteht eine GDI aus der Konfluenz verschiedener Datenanbieter, von denen jeder einen Zugriff auf seine Daten über einen Webservice ermöglicht. Davis schreibt weiter, dass der Anwender, um seinen Ansprüchen entsprechende Daten zu finden, Service-Metadaten durchsucht, welche Aussagen über die verfügbaren Geodaten und Services beinhalten. Die Suche erfolgt dabei interaktiv über ein Geoportale mit Hilfe von Suchfunktionen oder anderen interaktiven Werkzeugen, wie beispielsweise einem Web Catalogue Service. Demzufolge ist das Geoportale als Teil einer GDI zu verstehen (Davis & Alves 2005) (vgl. Abbildung 2.10). Geoportale bieten Such-, E-Commerce- und Administrationsfunktionalitäten und bilden somit das Herz einer GDI (Maguire & Longley 2005). Das Geoportale bedeutet einen Zugriffspunkt auf eine GDI.

⁵⁰Spatial Data Infrastructure

⁵¹US Federal Geographic Data Committee

⁵²Geodateninfrastruktur

⁵³Interministerieller Ausschuss für Geoinformationen

In den USA wurde im Jahre 2002 das aktuelle Konzept des NSDI⁵⁴ überarbeitet und an aktuelle Technologien und Bedürfnisse angepasst. Es entstand das GOS⁵⁵, welches Zugriff auf Geoinformationen über ein webbasiertes Portal ermöglicht. Das Konzept des GDI-DE sieht einen dreistufigen Aufbau mit der Entwicklung eines Metadateninformationssystemes (GeoMIS.Bund), der Harmonisierung der Geodatenbestände und dem Aufbau eines Geodatenportales (GeoPortal.Bund) vor (Reindl 2005).

Internet GIS

Auch die Entwicklung der Geoinformationssysteme (GIS) wurde beeinflusst durch die Möglichkeit, Dienste und Informationen über das vernetzte System des Internets verfügbar zu machen. Ein Internet GIS nutzt über das Internet verfügbare Geodatenquellen sowie Werkzeuge zur Datenprozessierung. Auch hier wird das Client-Server-Prinzip verwendet: Das GIS greift auf einen entfernt liegenden Service zu, welcher einen gewünschten Prozess ausführt oder Daten bereitstellt. Alternativ wird auch von verteilten GIS oder online-GIS gesprochen.

Dabei lassen sich die Anwendungen nach Peng & Tsou (2003, S.25 ff.) unterteilen in die Bereiche *Datenaustausch*, *Geoinformations-Verbreitung* sowie *Online-Datenprozessierung*. Daten können nicht nur über das Internet ausgetauscht werden (z.B. über das FTP⁵⁶), sondern der Zugriff erfolgt direkt aus der Applikation heraus auf den Datenservice. Hierdurch können zusätzlich Suchfunktionen verwendet und die Daten direkt in der Anwendung genutzt werden. So bietet das Produkt *ArcGIS-Online* die Möglichkeit, aus dem GIS heraus auf Daten eines entfernten Servers zuzugreifen und für die eigenen Zwecke zu verwenden. Auch andere GIS ermöglichen über WMS-Schnittstellen bereitgestellte Daten in die Applikation zu integrieren. Die Ergebnisse einer GIS-Analyse können nicht mehr nur ausgedruckt und verteilt, sondern über das Internet online publiziert werden. Auf diese Weise können zum Beispiel schnellstmöglich aktuelle Wetterinformationen oder die Verkehrssituation verbreitet werden. So bietet das Verkehrsamt der Stadt Seattle eine Kartendarstellung der aktuellen Verkehrsdichte⁵⁷, welches aus aktuellen Verkehrsdaten gespeist wird.

Des Weiteren wird es möglich, die Datenprozessierung an anderer Stelle von einem Service durchführen zu lassen. Das GIS bietet eine Verknüpfung zu diesem Service, so dass der Nutzer den Eindruck erhält, das GIS selbst beinhalte diese Funktionalität. Burghardt et al. (2005) beschreiben einen Online-Generalisierungsdienst, welcher als Funktion in ein GIS integriert wird. Der entsprechende Datensatz wird an den Service gesendet, dort generalisiert und das Ergebnis zurückgeliefert und in dem GIS dargestellt. Dabei greift dieser Service auf die WPS⁵⁸-Spezifikation des OGC zurück, welche darauf ausgelegt ist, Anwendern über das Netzwerk GIS-Funktionalitäten anzubieten.

Produktkategorien der Geo-Webdienste aus Kundensicht

Ein Geo-Webdienst muss nicht allumfassend sein, wie ein Geoportal oder eine GDI, denn Geo-Webdienste sind Teile einer solchen und können für ganz spezifische Anwendungen konzipiert sein. Es existieren eine Reihe unterschiedlicher Kategorisierungen der Geo-Webdienste, welche sich nicht gegenseitig ausschließen sollen, sondern auf verschiedene Sichtweisen zurückzuführen sind. In den vorangegangenen Abschnitten sind Klassifizierungen aus technischer Sicht zu finden. Im Folgenden sollen die Webdienste und deren Nutzen aus Sicht des Anwenders vorgestellt werden.

Eine Studie der *MICUS Management Consulting GmbH* zum Aufbau einer Geodateninfrastruktur in der Schweiz (Fornfeld & Oefinger 2005) bildet hierzu folgende Produktkategorien der Geo-Webdienste aus Kundensicht:

- *Auskunft*: Ergebnis einer Anfrage, das keine technische Weiterverarbeitung ermöglicht
 - Datenbereitstellung: Nur Darstellung am Bildschirm (Viewer) oder Druckfunktion
 - Zweck: Gezielte Auskunft
 - Darstellung: Oftmals aufbereitete Form
 - Beispiel: Map Services, Routingservices; z.B. Auskunftssystem „BORIS.nrw“⁵⁹ zur Darstellung der Bodenrichtwerte. Der Kunde gibt eine bestimmte Adresse ein und erhält über den Viewer Auskunft über die Grundstückspreise in Form einer Kartendarstellung

⁵⁴National Spatial Data Infrastructure

⁵⁵Geospatial One-Stop

⁵⁶File Transfer Protocol

⁵⁷<http://www.wsdot.wa.gov/traffic/seattle/>

⁵⁸Web Processing Service

⁵⁹<http://www.boris.nrw.de>

- *Daten*: Datenbezug im Wesentlichen durch die Definition eines räumlichen Bereiches
 - Datenbereitstellung: Kunde erhält Daten online, nicht auf Datenträger. Daten werden in das System des Kunden integriert
 - Zweck: Eigenanalysen, -planung
 - Darstellung: Rohdaten oder aufbereitet
 - Beispiel: Der Geodatenshop des Geozentrums Hannover⁶⁰ stellt Raster- und Vektorkarten zum Download gegen Gebühr bereit
- *Lösungen* (bezieht sich auf die Problemlösung beim Kunden): Nach spezifischer Kundenanfrage erstelltes, in der Regel aus mehreren Komponenten bestehendes Geodatenprodukt
 - Datenbereitstellung: Daten werden in das System des Kunden integriert, Software verbleibt beim Anbieter oder wird auf den Systemen des Kunden installiert
 - Zweck: Abhängig von der Aufgabe des Kunden
 - Darstellung: In der Regel lösungsorientiert aufbereitet
 - Beispiel: Internet GIS, das Produkt *The Button* der Firma *Terramapserver* ermöglicht z.B. den Bezug von Geoinformationen direkt aus dem firmeneigenen CAD/GIS

2.4 Mobile Webdienste

„No matter how you slice it or analyze it, the world is becoming mobile. Suddenly, wireless is an entirely new medium for commerce. Location is a key piece, because that mobile device moves with you.“ (Mark Cattini, MapInfo Präsident und CEO⁶¹, in einem Interview im April 2005).

Das Internet ist in den letzten zehn Jahren enorm gewachsen, sowohl die Inhalte, als auch deren Anwender, welche diese Angebote täglich nutzen. Zwei Drittel aller Deutschen (66 Prozent) ab 18 Jahren hatten im II. Quartal 2006 einen Zugang zum Internet (Forschungsgruppe Wahlen⁶²). Im Juli 2006 existierten weltweit etwa 440 Millionen Domainbetreiber, die rasante Entwicklung lässt sich in der Abbildung 2.11 ablesen.

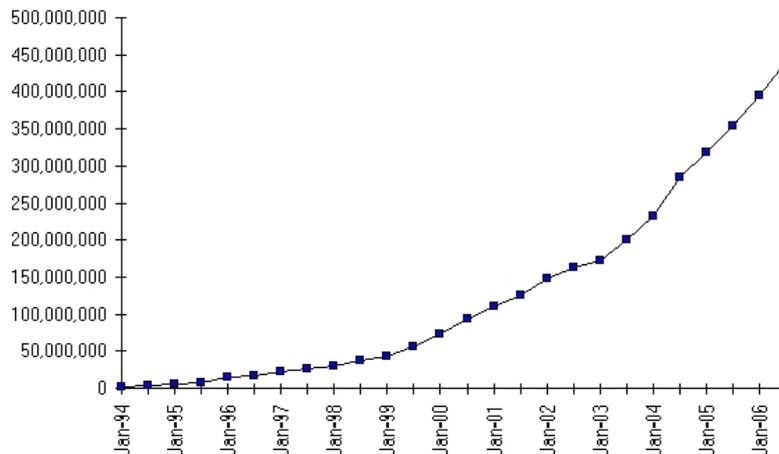


Abb. 2.11: Entwicklung der Internet-Hosts weltweit (Quelle: Internet Software Consortium <http://www.isc.org>)

Gleichzeitig sind mobile Geräte überall verfügbar, kaum jemand bewegt sich heute ohne sein Mobiltelefon. Durch die Kombination beider Phänomene entstehen neue Möglichkeiten, den Nutzer mit Informationen und anderen hilfreichen Diensten zu versorgen. Es sind heute schon mehr Mobiltelefone als PCs an das Internet angeschlossen (Beaulieu 2002, S.3). Abbildung 2.12 zeigt die Wertschöpfungskette mobiler Dienste und warum diese ein wirtschaftlich interessantes Anwendungsfeld für viele Geschäftsbereiche darstellen. Für den Bereich der Geoinformation sind hier vor allen Dingen die Applikationsentwicklung, Datenbereitstellung und Datenzusammenführung interessant.

⁶⁰<http://www.geoshop-hannover.de>

⁶¹Chief Executive Officer

⁶²Forschungsgruppe Wahlen: <http://www.forschungsgruppe.de>



Abb. 2.12: Wertschöpfungskette mobiler Dienste (nach Müller-Veerse 1999)

In den vorangegangenen Abschnitten wurden Architekturmodelle für Webdienste diskutiert, in denen der Nutzer von seinem Arbeitsplatz-Rechner direkten Zugriff auf das Internet hat. Daneben wurden auch mobile Nutzer als mögliche Clients beschrieben; die Frage bleibt jedoch: Wie werden die Daten und Webdienste auch für mobile Nutzer verfügbar? Vor allen Dingen ist hier interessant zu wissen, ob relevante Unterschiede bestehen zwischen einem mobilen und einem nicht mobilen Nutzer. Definitionsgemäß beschreiben mobile Webdienste den Zugriff auf Webdienste mit Hilfe mobiler Geräte, wie Mobiltelefonen, Smartphones oder PDAs.

Hinzu kommt die Möglichkeit, mit Hilfe verschiedener Techniken die Position dieser Geräte und damit auch des Anwenders zu bestimmen. Den größten Teil der Konsumenten mobiler Dienste stellen dabei Außendienstmitarbeiter und vor allen Dingen Nutzer ortsbezogener Dienste dar. Der Außendienstmitarbeiter kann in seinem mobilen GIS festgestellte Veränderungen gleich editieren und in die Datenbank übertragen. Ortsbezogene Dienste ermöglichen ihren Kunden die Bestimmung des Standortes und helfen ihnen bei der Suche bestimmter Lokalitäten.

Dabei könnte die Standard-Webservicetechnologie auch direkt auf einem mobilen Geräte angewendet werden. Eine Shopping-Anwendung auf einem Mobiltelefon beispielsweise könnte über eine SOAP-Anweisung eine Operation eines Webservices auslösen, um etwas zu kaufen und Zahlungsinformationen zu übermitteln (Farley & Capp 2005). Farley schreibt weiter, dass hierdurch jedoch nicht die Möglichkeiten ausgeschöpft werden, die ein mobiler Webservice in sich birgt. Dieser unterscheidet sich von einem traditionellen Webservice durch folgende Faktoren:

- Die Portabilität des Gerätes
- Die Verbindung des Gerätes mit einer bestimmten Nutzeridentität: GSM Mobiltelefone enthalten die SIM⁶³-Card, welche dem Gerät die Identität des Teilnehmers mitteilt und dieses somit einem bestimmten Nutzer oder Rechnungs-Adressaten zugeordnet werden kann
- Die mögliche und auch notwendige Personalisierung der Anwendung für den Nutzer: Es müssen Informationen den persönlichen Anforderungen angepasst und vor allen Dingen die Anzahl der Optionen für den mobilen Nutzer reduziert werden (auf Grund der limitierten Darstellungsfläche sowie dem geringen Interaktionswunsch des Nutzers z.B. wg. schlechtem Wetter, Lärm, Bewegung des Nutzers etc.)
- Die Einschränkungen bedingt durch die Limitierungen des Gerätes, wie das beschränkte Nutzerinterface, die Größe sowie geringe Leistung und geringere Bandbreiten (vgl. Unterabschnitt 2.4.1)
- Die Möglichkeit der Positionsbestimmung: In Verbindung mit den ersten beiden Punkten kann die Position des Gerätes in den meisten Fällen mit der Position des Eigentümers (SIM-Card Inhabers) gleichgesetzt werden

Derzeitige Applikationen nutzen dabei weniger den auf den verschiedenen Geräten vorhandenen Micro-Browser, sondern bieten für jede Gerätekategorie eine spezielle Version, welche der Anwender herunterladen und installieren kann. So existiert beispielsweise der Kartendienst der Firma *Google*⁶⁴ auch in einer Java2ME Version für mobile Geräte⁶⁵. Diese Applikation wird mit dem mobilen Gerät bei der ersten Benutzung heruntergeladen und installiert. Der Nutzer kann nun für seine aktuelle Position einen Kartenausschnitt abfragen (siehe Abbildung 2.13). Dies zeigt, wie flexibel Webservices einsetzbar sind, denn es ist möglich, beliebige Nutzer-Applikationen in das System zu integrieren.

⁶³Subscriber Identification Module

⁶⁴<http://maps.google.com>

⁶⁵<http://www.google.com/gmm/index.html>



Abb. 2.13: Kartendienst der Firma Google für mobile Geräte (Quelle Google: <http://www.google.com/gmm/index.html>)

Dabei sind folgende Komponenten Teil der mobilen Webdienste (Steiniger et al. 2006):

- *Mobile Geräte:* Der Anwender benötigt ein Gerät, mit dem er auf die einzelnen Services zugreifen und die resultierende Antwort betrachten oder anhören kann. Eine Bedingung für diese Gerätekategorie ist seine Transportabilität
- *Drahtloses Kommunikationsnetzwerk:* Um mit dem Serviceanbieter kommunizieren zu können, wird ein drahtloses Netzwerk zur Datenübertragung benötigt, welches überall verfügbar ist
- *Positionierungsdienste:* Die Positionsbestimmung ist nicht essentiell für die mobile Nutzung von Webservices, eröffnet aber völlig neue Servicemöglichkeiten, die sich in dem Zweig der LBS wiederfinden
- *Serviceanbieter:* Diese bieten eine Reihe von Diensten, wie die Positionierung, Routing oder Suchdienste, zugeschnitten auf die Position und die aktuellen Interessen des Nutzers
- *Datenanbieter:* Serviceanbieter unterhalten einen Teil der Informationen, die sie über ihren Dienst anbieten, selbst. Gleichzeitig greifen diese jedoch auf Daten Dritter, wie beispielsweise Geobasisdaten, zurück

Die einzelnen Punkte sollen in den folgenden Abschnitten näher ausgeführt werden.

2.4.1 Mobile Geräte

Die Ausbreitung der oben aufgeführten mobilen Services wurde weitgehend vorangetrieben durch die fortschreitende Entwicklung und Verbreitung mobiler Geräte. Dabei hat sich viel verändert, seit im Jahre 1987 die ersten tragbaren Mobiltelefone der Firma *Siemens* im deutschen C-Netz genutzt wurden und auch seit *John Sculley* von der Firma *Apple* im Jahre 1993 erstmalig den Begriff „Personal Digital Assistant“ benutzte, damals im Zusammenhang mit einer neuen Produktkategorie namens *Newton*, dem ersten PDA also. Auch wenn *Apple* diese Produktkategorie inzwischen eingestellt hat, die mobilen Geräte haben ihre Klienten gefunden und sich etabliert. Was die Entwicklung bis heute hervorgebracht hat, kann jeder Deutsche selbst beobachten, wenn er oder sie in seine Tasche greift⁶⁶.

Dabei sind Mobiltelefone nur ein mobiles Medium neben Laptop, PDA und Smartphone (Abbildung 2.14). Laptops und Tablet-Computer zählen dabei auch zu den mobilen Geräten, da diese auf Grund von Drahtlosnetzwerken und ihrem reduzierten Gewicht und der Größe nicht an einen bestimmten Platz gebunden sind. Trotzdem sollen diese im weiteren Verlauf nicht weiter als mobile Geräte betrachtet werden, da sie in allen anderen Belangen einem Desktop PC entsprechen. PDAs und Mobiltelefone sind dagegen mobile Computer, welche sich stark von dem bekannten Desktop PC bezüglich der Rechenleistung, den Formfaktoren sowie den ihnen zugeordneten Funktionen unterscheiden.

Personal Digital Assistants

Ein PDA (siehe Abbildung 2.14, Mitte) ist ein tragbarer Computer, welcher einfache Funktionalitäten, wie Kalender, Kontakte, Notizblock, aber auch Webbrowser, E-Mail-Client, PDF-Reader und Media Player in sich trägt. Diese Geräte sind klein genug, um in der Hosentasche getragen zu werden. Als Eingabeschnittstelle dienen entweder sogenannte Touchscreens, welche mit einem Stift bedient werden, oder in einigen Fällen auch

⁶⁶Es gibt zur Zeit (August 2006) nach Aussage des BITKOM⁶⁷ 82,8 Mio. Mobilfunkanschlüsse in Deutschland, also statistisch gesehen mehr als einen Anschluss pro Einwohner



Abb. 2.14: Mobile Geräte: Smartphone (links), PDA (Mitte), Mobiltelefon (rechts)

eine kleine Tastatur. Die neueste Generation dieser Geräte ist mit WiFi⁶⁸ Funktionalitäten, Bluetooth oder GPRS⁶⁹ ausgestattet. Weiterhin ist in einigen Geräten ein GPS-Empfänger integriert. Wurde diese Geräteform zunächst von der Firma *Palm* (und deswegen als *Palm Handhelds* bezeichnet) und dem Betriebssystem *Palm OS* dominiert, so werden zur Zeit die meisten PDA mit dem Betriebssystem *Windows Mobile* (auch bekannt unter dem Namen *Pocket PC*) der Firma *Microsoft* ausgestattet. Das Betriebssystem wird für PDAs und für Smartphones vertrieben.

Mobiltelefone und Smartphones

Mobiltelefone unterscheiden sich von den oben beschriebenen PDAs vor allen Dingen auf Grund ihrer gewünschten Funktionalität, sie sollen in erster Linie zum Telefonieren und in weiterer Entwicklung der Netzwerkstandards (siehe Unterabschnitt 2.4.2) auch zum Datenaustausch genutzt werden. Daher beherrschen Mobiltelefone heute Dienste wie Sprachübertragung (telefonieren), WAP⁷⁰, GPRS oder ähnliche Zugänge zum Internet sowie SMS⁷¹ und MMS⁷². Die Funktionalitäten der Mobiltelefone wurden dabei jeweils geprägt von den verfügbaren Netzen (von 1G bis 3G).

Neueste Entwicklungen lassen Mobiltelefone und PDAs zusammenwachsen zu sogenannten Smartphones (Abbildung 2.14, links). Diese vereinen also die Vorteile eines PDA mit seinen Office-Funktionalitäten mit denen des Mobiltelefons und bieten gleichzeitig die Möglichkeit des Datenaustausches im Internet über verschiedene drahtlose Kommunikationsprotokolle. Smartphones werden zunehmend dominant auf dem Markt, während die Kaufzahlen von einfachen PDAs, auch als „unconnected handheld“ bezeichnet (Anderson 2005), rückläufig sind. Entsprechende Zahlen der Marktforschungsgruppe Canalys (2005) beschreiben ein Wachstum der Smartphones um 170% im ersten Quartal 2005. Der Vorteil des PDA gegenüber dem Mobiltelefon oder Smartphone liegt in seiner Ähnlichkeit zum PC. Inhalte und Anwendungen des PDA können nach Belieben des Nutzers neu konfiguriert oder installiert werden. Im Gegensatz dazu bieten Mobiltelefone zwar eine Möglichkeit zur Personalisierung, die Softwarekomponenten sind dagegen fest vom Hersteller oder Mobilfunkbetreiber vorgegeben und können nicht verändert werden. Einige dieser Geräte sind jedoch Java-fähig, so dass es möglich ist, Java-ME⁷³-Applikationen aufzuspielen.

Eigenschaften mobiler Geräte

Charakteristika aller mobilen Geräte sind ihre besonderen Formen, technischen und ergonomischen Beschränkungen sowie die Fähigkeit, drahtlos mit Netzwerken zu kommunizieren. Neben dem enormen Vorteil, immer und überall präsent zu sein sowie der Fähigkeit, ihre aktuelle Position zu bestimmen, sind diese auf Grund ihrer geringen Größe und Mobilität gehandikapt. Ihre Rechenleistung und auch Bedienmöglichkeiten sind beschränkt, die Displays sind klein und die Verbindung zu den Netzwerken wird oftmals gestört und besitzt darüber hinaus eine (heute noch) geringe Bandbreite. Mobile Geräte sind batteriebetrieben, so dass die gesamte Hardware

⁶⁸Wireless Fidelity

⁶⁹General Packet Radio Service

⁷⁰Wireless Application Protocol

⁷¹Short Message Service

⁷²Multimedia Messaging Service

⁷³Micro Edition

auf eine möglichst energieschonende Performance ausgelegt ist. Weiterhin besitzen mobile Geräte eine geringe Speicherkapazität, sowohl zur Datenspeicherung als auch zum Betrieb laufender Programme.

Ein Blick in die Zukunft deckt folgende Trends mobiler Geräte auf: Wie oben schon erwähnt, werden PDA und Mobiltelefon zusammenwachsen. *Gartner*, Gründer des gleichnamigen Marktforschungsunternehmens *Gartner-Consultants*, prädizierte 2005: „*The era of the smartphone is upon us*“. Der mobile Nutzer hat somit Zugriff auf alle verfügbaren Netzwerke und besitzt gleichzeitig entsprechende Applikationen und die Möglichkeit der Ortsbestimmung zum Beispiel über GPS. Limitierender Faktor bei der Entwicklung der Rechenleistung ist die Leistungsfähigkeit der Batterien. Brennstoffzellen sollen hier den Ausweg aus dem Dilemma bedeuten. Die Displays dieser Geräte bestehen zur Zeit aus TFT⁷⁴ oder OLED⁷⁵ mit einer Auflösung von maximal 640x480 Pixeln (VGA⁷⁶). Zukünftige Entwicklungen sehen reflektierende, flexible Bildschirme, die biegsam sind und somit gerollt werden können (Anderson 2005), auch paperlike-displays genannt (siehe Abbildung 2.15). Weitere Forschungen berichten von Informationsdarstellungen, die direkt auf das menschliche Auge projiziert werden und somit den Bildschirm überflüssig machen (Anderson 2005).



Abb. 2.15: Paperlike Displays (Beispiel der Firma PolymerVision, Quelle: <http://www.polymervision.nl/>)

Die Prozessoren werden nach Ankündigungen der Firma *Intel* bei den PDAs mit einem XScale Prozessor mit Namen *Monahans* in diesem Jahr (2007) die Grenze von 1GHz übersteigen, bei den Smartphones erreicht seit Ende 2006 der Doppelkern-Prozessor *Merom* (Goodwins 2005) Geschwindigkeiten von mehr als 600 MHz bei einem Stromverbrauch von 45 Watt. Nach Aussagen der Firma *Intel*⁷⁷ (Anderson 2005) ist das Ziel der Entwicklungen, Leistungen von Desktop-Computern zu erreichen bei einem Stromverbrauch von maximal 0,5 Watt. Zukünftige Geräte werden außerdem mit Festplatten ausgestattet sein, welche Speicherkapazitäten von bis zu 4 GB bieten.

2.4.2 Anbindung mobiler Geräte - Drahtlose Netzwerke

Es existieren unterschiedliche Wege, drahtlos Daten über das Internet auszutauschen.

Wireless Personal Area Networks - WPAN

Der WPAN⁷⁸ Standard wird vom IEEE⁷⁹ entwickelt⁸⁰. Diese Art des Netzwerks folgt der Idee, einzelne Komponenten wie PCs, PDAs, Peripheriegeräte, Digitalkamera, Mobiltelefon und andere elektronische Geräte drahtlos miteinander zu vernetzen, damit diese miteinander kommunizieren können. Da dieses Netz lediglich ein Kabel ersetzen sollte, wurden hier Reichweiten von 0,2 m bis zu 50 m realisiert, kürzer also als die Reichweite des WLAN⁸¹. Dieses Netz soll lediglich die persönliche Umgebung abdecken. Durch die geringe Reichweite dieser Netze werden Störungen durch andere Netze zum größten Teil vermieden. Hier werden Datenübertragungsraten von 1 Mbit/s (Bluetooth) bis 4 Mbit/s (FIR-IrDA) realisiert.

⁷⁴Thin Film Transistor

⁷⁵Organic Light-Emitting Diode

⁷⁶Video Graphics Array

⁷⁷<http://www.intel.com>

⁷⁸Wireless Personal Area Network

⁷⁹Institute of Electrical and Electronics Engineers

⁸⁰<http://ieee802.org/15/index.html>

⁸¹Wireless Local Area Network

Wireless Local Area Networks - WLAN

Dem drahtgebundenen Fall am ähnlichsten ist die Möglichkeit, eine dem WiFi Standard entsprechende Verbindung zwischen dem mobilen Geräte und einer Basisstation (einem Access-Point), welcher seinerseits mit dem Internet verbunden ist, aufzubauen (Abbildung 2.16). Diese Verbindung kann Entfernungen bis zu 100 m drahtlos überbrücken. WiFi-zertifizierte Komponenten unterstützen die auf der IEEE-802.11-Familie aufbauenden Standards zur Kommunikation zwischen einem zugreifenden Gerät und einer Basisstation. Sind sowohl das mobile Gerät als auch der Access-Point WiFi zertifiziert, so ist eine technisch problemlose Verbindung zu erwarten. Die IEEE-802.11-Familie reicht in ihrer Entwicklung von 802.11a über den heute am häufigsten verbreiteten Standard IEEE-802.11b mit einer theoretischen Übertragungsgeschwindigkeit von 11 MBit/s. Gleichzeitig etabliert sich mehr und mehr der IEEE-802.11g mit theoretischen Geschwindigkeiten von bis zu 54 MBit/Sekunde. Dieser Standard wird von der kommenden Generation mobiler Geräte unterstützt werden. Die Zukunft sieht das IEEE-802.11n mit Geschwindigkeiten von bis zu 250 MBit/s für das Jahr 2007 voraus. Ein weiterer in naher Zukunft verfügbarer Standard wird das WiMax⁸² sein, ein Standard basierend auf dem IEEE 802.16. Dieser ermöglicht tragbaren, mobilen Geräten einen direkten Zugriff auf das Internet über größere Entfernungen von mehreren Kilometern bis zur Basisstation. WiMax ermöglicht Übertragungsgeschwindigkeiten von bis zu 75 MBit/s (Anderson 2005).

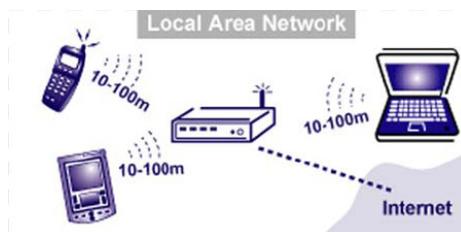


Abb. 2.16: Drahtloses LAN zur Anbindung mobiler Geräte an das Internet (aus Steiniger et al. 2006)

Wireless Metropolitan Area Networks - WMAN

Das WMAN⁸³, der IEEE-802.16 Standard, kombiniert mehrere WLAN-Netze zu einem größerem, einen Stadtteil oder ein Stadtgebiet abdeckendes Netz aus mehreren Access-Points. Der Anbieter eines WMAN kombiniert einzelne WLAN Zugriffsknoten zu einem Verbund und ermöglicht seinen Nutzern somit den Zugang zum Internet. Diese Art der Internetanbindung ist vor allen Dingen in den Gebieten zu finden, in denen keine drahtgebundene Anbindung, wie DSL⁸⁴, verfügbar ist.

Wireless Widearea Networks - WWAN

Die alternative und vor allen Dingen fast überall verfügbare Form der Datenübertragung beruht auf dem Telekommunikationsnetz. Drahtlose Telekommunikation besteht heutzutage aus sogenannten zellularen Netzen, aus einem Netzwerk von Funkzellen also. Dabei stellt eine Zelle den Bereich dar, in dem eine Verbindung zu einer bestimmten Basisstation oder einem Transceiver⁸⁵ möglich ist. Diese Basisstationen sind untereinander vernetzt (vgl. Abbildung 2.17), wobei der Mobilfunknutzer in der Regel mit *einer* Basisstation kommuniziert (Peng & Tsou 2003, S.457). Die Technologie dieser Netzwerke hat sich dabei ständig weiterentwickelt. Die erste Generation (1G) dieser war analog und konnte nur zur Übertragung der Stimme, also zum Telefonieren genutzt werden. In der zweiten Generation (2G) wurden die Stimmen beim Telefonieren in digitale Signale übersetzt. Dadurch wurde die Sprachqualität erheblich verbessert. Gleichzeitig war es so möglich, auch Daten zu übertragen, allerdings mit einer sehr langsamen Geschwindigkeit von weniger als 10 Kbps⁸⁶. Der meist verbreitet 2G-Standard ist das GSM⁸⁷. Auf dem Weg zur dritten Generation (3G) wurden die Netze zunächst für höhere Übertragungsgeschwindigkeiten der Daten aktualisiert, die etwa der eines Modems entsprechen; man spricht hier von 2^{1/2}G Netzen. In diesem Zusammenhang ist vor allen Dingen eine Weiterentwicklung des GSM, das

⁸²Worldwide Interoperability for Microwave Access

⁸³Wireless Metropolitan Area Network

⁸⁴Digital Subscriber Line

⁸⁵Kunstwort aus **Transmitter + Receiver**, einem Sender und Empfänger also, oftmals auch als Mobilfunksender bezeichnet. Teilweise werden auch Sektorantennen eingesetzt, die jeweils nur einen Sektor von 120 Grad abdecken.

⁸⁶Kilobits per second

⁸⁷Global System for Mobile Communications

GPRS⁸⁸ zu nennen. Der große Vorteil des GPRS ist allerdings weniger der Geschwindigkeitszugewinn auf 14,4 kbps, sondern vielmehr die Technologie, Daten paketorientiert zu übertragen. Die Bandbreite der Datenleitung wird außerdem nur dann aktiviert, wenn Daten übertragen werden. In der übrigen Zeit besteht eine dauerhafte Verbindung bei einer Rate von maximal 0,1 kbps, was dem Nutzer vor allen Dingen Gebühren spart.

Die dritte Generation erhöht dann abermals die Geschwindigkeit der Datenübertragung auf bis zu 2 MBit/s. Hier spielt vor allen Dingen das UMTS eine tragende Rolle. 3G ist der heute und bis etwa 2010 gültige Standard, bevor dieser dann von der vierten Generation (4G) abgelöst werden wird. In der vierten Generation werden Übertragungsgeschwindigkeiten von bis zu 100 Mbps erwartet, was dem heutigen LAN entspricht (Peng & Tsou 2003, S.458).

Die Entwicklungen, sei es WiMax, UMTS oder 4G, tendieren eindeutig in die Richtung, auch mobilen Nutzern, über Hotspots oder Datennetze, eine für Datenübertragungen attraktive Alternative zum Desktop PC zu bieten. Hindernisse in der heutigen Zeit stellen vor allen Dingen die im Vergleich zum LAN⁸⁹ hohen Preise und geringeren Bandbreiten des mobilen Internetzugriffs dar. Die Herausforderung für mobile Geoservices liegt daher zur Zeit noch in der Reduzierung der Datenmenge, um dem Nutzer Zeit und Kosten zu sparen. In naher Zukunft werden sich jedoch die Preismodelle den neuen Möglichkeiten anpassen, so dass keine Einschränkungen von Seiten der Datennetze bezüglich der zu übertragenden Datenmengen mehr zu berücksichtigen sind.

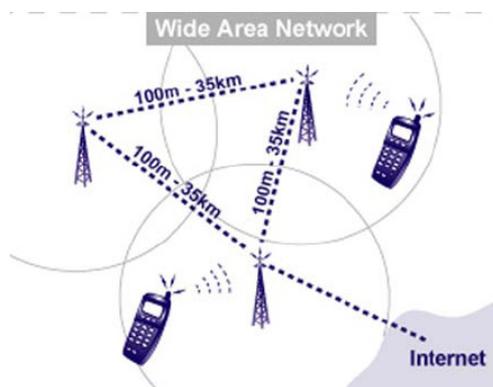


Abb. 2.17: Drahtloses WAN zur Anbindung mobiler Geräte an das Internet (aus Steiniger et al. 2006)

Zusammenfassung

Es lässt sich also zusammenfassend feststellen, dass neben den Desktop-Computern auch mobile Geräte ernstzunehmende Clients für Webservices darstellen. Dabei sind auf der einen Seite die heute noch bestehenden Limitierungen der Hardware und Netzwerke zu berücksichtigen. Die Palette mobiler Geräte ist sehr vielfältig. Dies kann die Entwicklung von Applikationen für mobile Nutzer erschweren, denn die Anwendung muss entweder auf das schwächste Glied ausgerichtet sein oder der Service unterscheidet zwischen den unterschiedlichen Plattformen und bietet für jede Gerätekategorie verschieden ausgestaltete Applikationen. Es müssen außerdem diverse Standards wie GPRS, CDMA⁹⁰ oder AMPS⁹¹ unterstützt werden. Inhalte, welche für Standard-Internetbrowser bestimmt sind, können von mobilen Geräten nicht ohne Weiteres interpretiert werden. Hier sind Internetstandards wie die WML⁹² anzuwenden. WML ist eine an die HTML angelehnte Seitenbeschreibungssprache, welche Teil des WAP ist und zur Darstellung veränderlicher Inhalte auf Mobiltelefonen entwickelt wurde.

Drahtlosnetzwerke haben darüber hinaus geringere Bandbreiten, längere Wartezeiten und einen geringeren Grad an Zuverlässigkeit verglichen mit drahtgebundenen Desktop-Lösungen. Die aktuellen Entwicklungen zeigen jedoch, dass diese Limitierungen reduziert werden oder aber alternative Lösungen zum Ausgleich dieser entwickelt werden, so dass die Probleme, mit denen die Entwickler mobiler Anwendungen heute zu kämpfen haben, in Zukunft weniger relevant sein werden.

⁸⁸General Packet Radio Service

⁸⁹Local Area Network

⁹⁰Code Division Multiple Access

⁹¹Advanced Mobile Phone Service

⁹²Wireless Markup Language

2.4.3 Ortsbezogene Dienste

Ortsbezogene Dienste oder auch *location based services (LBS)* bezeichnen im Allgemeinen das Konzept eines Dienstes, welcher insbesondere die aktuelle Position des Nutzers mit in Betracht zieht und somit einen Mehrwert für den Dienstanbieter und auch den Nutzer bedeutet. Die Serviceanbieter können dabei die Möglichkeit zur Positionsbestimmung der mobilen Geräte für ihre Dienste nutzen. Hierzu sind grundsätzlich drei Verfahren zu unterscheiden, die satelliten-, netzwerk- und funkzellenbasierten Methoden. Die netzwerk- und funkzellenbasierten Methoden können ohne zusätzliche Infrastruktur realisiert werden und erlauben Genauigkeiten der Positionsbestimmung bis zu 50 Metern, abhängig von der Größe der jeweiligen Netzwerk- oder Funkzelle. Zur Nutzung satellitengestützter Ortungssysteme werden zusätzliche in dem mobilen Gerät implantierte Empfänger benötigt. Diese Verfahren lassen eine Positionsbestimmung mit einer Genauigkeit von etwa 10 Metern zu (Bauer 2003). Allerdings gilt diese Aussage mit gewissen Einschränkungen, da hier zur Positionsbestimmung immer mindestens vier Satelliten sichtbar sein müssen. Je mehr Satelliten sichtbar sind, desto zuverlässiger wird die Positionsbestimmung. In engen Häuserschluchten, im Wald oder in den Bergen kann es passieren, dass durch Abschattungen die Verfügbarkeit von GPS⁹³, GLONASS⁹⁴ oder GALILEO nicht gegeben ist.

Auch wenn die Idee des ortsbezogenen Dienstes nicht neu ist (Ortsbestimmung für das Flottenmanagement existierten schon in den 80er Jahren), so ist eine Kommerzialisierung der LBS erst durch den technologischen Fortschritt der letzten Jahre, wie die Weiterentwicklung der mobilen Geräte, Verbesserung der drahtlosen Netzwerke sowie die Möglichkeiten der Positionsbestimmung und durch die Trends der Industrie interessant und möglich geworden. Hierdurch sind neue Anwendungsfelder entstanden, wie Notdienste, Flottenmanagement, Navigation oder ortsabhängige Informationsdienste.

In seiner einfachsten Form stellt der LBS einen Positionierungsdienst dar, beantwortet dem Nutzer also die Frage: „Wo bin ich?“. Die Position allein liefert dem Anwender allerdings nur wenig Informationen. Sinnvollerweise wird der Standort mit einer Karte kombiniert; dieser Service wird dann „Kartenservice“ genannt. Die resultierende Karte kann mit hilfreichen Informationen angereichert werden, wie beispielsweise der Position von Sehenswürdigkeiten (Points of Interest) für den Touristen; hier also die Frage: „Was ist in der Nähe?“. Schließlich können weitere Informationen über Dienstleistungsbetriebe folgen, die dem Client in seiner aktuellen Situation hilfreich wären (Prinzip der gelben Seiten). Ortsbezogene Dienste bieten dabei nicht nur Informationen *über* die Umgebung, sondern auch Dienste *in* der Umgebung. Zu nennen wäre zum Beispiel ein Taxiruf zur aktuellen Position oder auch der automatische Fahrkartenkauf. Des Weiteren stellen Notfall- und Sicherheitsdienste eine wichtige Anwendergruppe dar. Auch hier zeigt sich, dass der entscheidende Faktor nicht die aktive Nutzung durch den Client ist, sondern dass die aktuelle Position vom Service genutzt wird (Virrantaus et al. 2001). Der Nutzer benötigt darüber hinaus Anweisungen, wie er oder sie zur Sehenswürdigkeit oder zum nächsten Restaurant gelangt: „Wie gelange ich von hier nach B?“. Diese Information liefert ihm der Routingsservice.

Barnes (2003) kategorisiert die potenziellen Anwendungsbereiche ortsbezogener Dienste in folgende Bereiche:

- *Sicherheit*: Die Möglichkeit, zu jedem Zeitpunkt die Position einer Person bestimmen zu können, bietet ein hohes Maß an Sicherheit in Notfallsituationen. Sie hilft Personen, welche selbst nicht mehr wissen, wo sie sich befinden oder aber nicht mehr in der Lage sind, ihre Position bekannt zu geben, wenn sie beispielsweise schwer verletzt sind. Somit ist es möglich, Personen in Not schneller Hilfe zukommen zu lassen
- *Navigation*: Navigationsservices bestimmen eine Route von der bestimmten Position zu einem Zielort und helfen dem Anwender gleichzeitig durch verbale oder visuelle Anweisungen, dieser Route bis zum Ziel zu folgen
- *Tracking*: Die interessanteste Komponente eines Tracking-Services ist die aktuelle Position. Es werden Fahrzeuge innerhalb eines Flottenmanagement-Systems verfolgt, genauso wie die aktuelle Position bestimmter Güter oder auch Personen. Die Nutzer sind nicht an der eigenen Position, sondern an dem aktuellen Standort von Freunden oder den eigenen Kindern interessiert. Anwendungen wie „Friend-Finder“ oder „Childwatch“ sind nur einige Beispiele dieser Servicekategorie
- *Transaktionen*: Die automatische Bezahlung ohne notwendigen Ticketkauf beruht auch auf der Bestimmung der aktuellen Position eines Nutzers. Betritt er beispielsweise einen Zug oder ein Museum, so wird ihm automatisch das Eintrittsgeld in Rechnung gestellt
- *Information*: Die Vielfalt ortsbezogener Informationen ist scheinbar unbegrenzt. Der Dienstanbieter berücksichtigt in seiner Information die aktuelle Position, aber auch Vorlieben des Nutzers und andere kontextbezogene Faktoren (siehe Abschnitt 2.5). Ein Beispiel zeigt das EU-Projekt *Web-Park*, welches dem

⁹³Global Positioning System

⁹⁴Globalnaya Navigatsionnaya Sputnikovaya Sistema

Besucher eines Nationalparks Informationen über Tier- und Pflanzenarten in der näheren Umgebung des Standortes übermittelt (Edwardes et al. 2003)

Eine weitere Definition unterscheidet zwischen „Push Services“ und „Pull Services“. Pull-Dienste zeichnen sich innerhalb einer Client-Server-Architektur durch die aktive Anforderung eines Dienstes von Seiten des Clients aus (3GPP 2002). Dagegen werden Push-Dienste an den Nutzer herangetragen, ohne dass dieser einen solchen Service explizit angefordert hat. Diese Dienste werden durch andere Ereignisse ausgelöst, wie zum Beispiel durch eine bestimmte Uhrzeit oder wenn der Anwender eine bestimmte Position erreicht oder passiert. Dabei lässt sich dieser für bestimmte Services registrieren und erhält bei Eintreten eines bestimmten Ereignisses die gewünschten Informationen. Auch die Werbung macht sich diese push-Dienste zunutze durch sogenanntes „m-advertising“, also Werbung für mobile Nutzer, die sich beispielsweise in der Nähe eines bestimmten Geschäftes befinden (Dimpfel & Heep 2004). Dabei haben sich diese Push-Services bislang nicht durchsetzen können, da der Aufwand, diese zu betreiben, unverhältnismäßig hoch ist. Es muss ständig die Position des Nutzers überprüft werden oder das Netzwerk muss regelmäßig die Nummern der Mobilfunkgeräte überprüfen, welche an einem bestimmten Geschäft vorbeigehen (Spiekermann 2004, S. 14).

Location based services zeichnen sich also durch die Einbeziehung des Nutzerstandortes sowie seiner Umgebung in das Serviceangebot aus. Hierdurch wird es möglich, den Anwender mit zielgerichteten Informationen zu versorgen. Informationen über den Anwender, wie in diesem Fall die Position, hilft dem System, sich optimal an die aktuellen Bedürfnisse anzupassen. Auch für den Bereich der Visualisierung der Geoinformationen in einer Karte ist dies sinnvoll, da somit der Schritt von der allgemeingültigen zu einer personalisierten oder adaptierten Karte vollzogen wird.

Das OGC hat speziell für dieses Anwendungsfeld die Open Location Service (OpenLS) Spezifikation (OGC 2005b) entwickelt. Ziel dieser Dokumentation ist die Entwicklung von Schnittstellen-Beschreibungen zur Nutzung räumlicher Informationen innerhalb einer drahtlosen Internetumgebung. Die entwickelten Spezifikationen sollen die Interoperabilität ortsbezogener Dienste unterstützen.

2.5 Adaption der Inhalte

Webdienste und insbesondere Geo-Webdienste ermöglichen eine individuelle Bearbeitung der Anfragen und können somit jeden Ergebnisdatsatz auf die aktuelle Situation und die Bedürfnisse des Client zuschneiden. Verknüpft man mit dem Begriff des Clients nun eine natürliche Person, einen Kunden, so besteht durch die individuelle Bearbeitung der Anfrage die Chance, eine persönliche Antwort auf eine Anfrage zu generieren, welche irrelevante Informationen weglässt und nur die für den Kunden interessanten Daten an diesen weitergibt. Die Firma *Amazon*⁹⁵ hat dieses Konzept in ihrem Verkaufs-Webportal als einer der ersten Anbieter in großem Stil umgesetzt. Der Kunde hat die Möglichkeit, auf alle Produkte zuzugreifen, erhält gleichzeitig jedoch, bedingt durch sein persönliches Profil, seine bisherigen Kauf tätigkeiten sowie den Käufen anderer Kunden eine Auswahl an Produkten, die ihn interessieren könnten. In den meisten Fällen wird die Möglichkeit der Individualisierung der Services zu kommerziellen Zwecken genutzt: Dem Kunden soll das für ihn richtige Produkt angepriesen werden, zu einem Zeitpunkt, an dem er dieses Produkt benötigen könnte und auch nur dann, wenn er sich an einem bestimmten Ort (zum Beispiel in der Nähe eines Restaurants) befindet.

Doch auch für den Bereich der Geoservices besteht hier nun eine Möglichkeit, geografische Inhalte nicht mehr allgemeingültig für eine große Anzahl potenzieller Kunden aufbereiten zu müssen, wie es für gedruckte Kartenwerke der Fall ist. Vielmehr kann das individuelle Profil des Nutzers Berücksichtigung finden, Inhalte können adaptiert werden.

Reichenbacher (2003) betitelt in seiner Arbeit vier Bereiche zur Adaption in der Kartografie und somit den Geo-Webdiensten:

- *Information*: Der Informationsinhalt wird an die aktuelle Situation, den Nutzer, seine Aktivität sowie das von ihm benutzte System angepasst
- *Nutzerinterface*: Das Nutzerinterface wird der aktuellen Situation, dem Nutzer, seiner Aktivität, dem von ihm benutzten System, den physikalischen Konditionen und der Art der Fortbewegung angepasst
- *Präsentation*: Die Visualisierung der Information wird der aktuellen Situation, dem Nutzer, seiner Aktivität, der Umgebung, dem von ihm benutzten System, den physikalischen Konditionen und der Art der Fortbewegung angepasst

⁹⁵www.amazon.de

- *Technologie*: Die Informationskodierung wird den spezifischen Eigenschaften des Gerätes (Bildschirmgröße und -auflösung, Speicher, Rechenleistung, etc.) oder der Übertragungsart (z. B. Bandbreite) angepasst

Dabei bleibt festzuhalten, dass die Anpassung bei dem mobilen Nutzer noch weiter geht, als bei dem Anwender zu Hause oder im Büro. Die mobile Nutzung bietet mehr Parameter zur Individualisierung, wie die aktuelle Position und die Umgebung oder die physikalischen Bedingungen. So lassen sich hier, je nach Wetterlage, Tages- oder Jahreszeit andere Informationen und angepasste Darstellungen realisieren. Gleichzeitig ist auch der Bedarf der Personalisierung hier viel höher, denn ein mobiler Nutzer ist vielen Limitierungen unterworfen. Die Umgebung, in der er sich befindet, ist in den meisten Fällen weniger komfortabel, Wettereinflüsse, Umgebungslärm und andere Tätigkeiten, wie gleichzeitiges Laufen oder Autofahren, erschweren die Bedienung und die Informationsaufnahme (Nivala & Sarjakoski 2003). Daher ist es wichtig, die relevanten Informationen möglichst schnell zu übermitteln (Reichenbacher 2003).

In dem EU-Projekt *GiMoDig*⁹⁶ (Sarjakoski et al. 2002) erhalten mobile Nutzer die Möglichkeit, über ein Nutzerinterface Karten abzufragen, welche ihrem persönlichen Profil angepasst wurden. Der Anwender kann die Sprache, sein Alter und seine aktuelle Tätigkeit, wie zum Beispiel wandern oder Rad fahren, angeben. Zusätzlich werden die aktuelle Tages- und Jahreszeit von dem Service in Betracht gezogen (Abbildung 2.18). Hält der



Abb. 2.18: *GiMoDig*-Projekt: Adaption der Darstellung an die aktuelle Situation des Nutzers (aus Nivala & Sarjakoski 2004)

Serviceanbieter zusätzlich ein Nutzerprofil vor, so stehen ihm nun Profil- und aktuelle Umgebungsinformationen zur Verfügung, um seinen Service optimieren zu können.

Außerdem bedarf es auf Grund des kleinen Displays, der eingeschränkten Bedienmöglichkeiten und den limitierten Rechenleistungen des mobilen Gerätes (vgl. Unterabschnitt 2.4.1) einer Anpassung und Einschränkung der darzustellenden Inhalte.

Somit besteht ein Bedarf, aber gleichzeitig auch die Chance, zur Adaption der Serviceinhalte und die Darstellung dieser. Individuelle Parameter zur Darstellung geografischer Inhalte umfassen den Maßstab der Karte und damit verbunden den Grad der Generalisierung und den Grad grafischer Details, die Ausprägung der Karte, beeinflusst durch die Wahl der grafischen Variablen nach Bertin (1983) (Form, Farbe, Füllung, Richtung, Muster, Größe) und hierin eingeschlossen die Wahl der Symbole, Schriftarten etc. Inhaltliche Aspekte dagegen treffen eine Auswahl der darzustellenden Objektklassen und deren Attribute sowie eine Beschränkung des zu beschreibenden Gebietes.

Voraussetzung zur Umsetzung der bis hierher beschriebenen Konzepte ist dabei natürlich die Realisierbarkeit. Soll beispielsweise eine Karte unter Verwendung der oben genannten Parameter an die aktuellen Bedürfnisse der Anwendung angepasst werden, so setzt dies voraus, dass das System in der Lage ist, diese Anfrage innerhalb kürzester Zeit zu bearbeiten. Probleme ergeben sich hier beispielsweise in der Generalisierung der Datensätze, da eine Aufbereitung hochauflösender Geobasisdaten für einen bestimmten Maßstab nicht innerhalb kürzester Zeit und vollständig automatisch zu leisten ist. Auch die übrigen Ansätze zur Adaption der Services sind hier kritisch zu betrachten. Somit ist eine Zwischenlösung praktikabel, die die Ergebnisse in notwendigem Umfang vorprozessiert und gleichzeitig an den Stellen, wo dies möglich ist, eine individuelle Anpassung vornimmt, beispielsweise bei der Auswahl der Inhalte.

⁹⁶Geospatial info-mobility service by realtime data-integration and generalisation

Während die Abfrage und Darstellung von Geoinformationen auf einem Desktop PC durch die genannten Möglichkeiten weitestgehend abgedeckt ist, werden für mobile Geräte weitere Möglichkeiten gesucht. Auch hier wird zunächst versucht, alle überflüssigen Inhalte vorweg zu eliminieren, da auf Grund der kleinen Darstellungsfläche der Kommunikationskanal zwischen dem Kartenautor und dem Kartenleser stark limitiert ist. Daneben existieren Ideen, den Maßstab und den Detailreichtum innerhalb einer Darstellung zu variieren. Lichtner (1979) vergrößert in seiner Kartendarstellung innerstädtische Bereiche. Ähnliche Ansätze finden sich auch bei Harrie et al. (2002), welche diese Technik zur Visualisierung von Geoinformationen auf mobilen Geräten einsetzen. Für die Darstellung wird für einen bestimmten Umkreis um eine Position ein größerer Maßstab verwendet als für die übrige Kartenfläche, wodurch eine Art Lupeneffekt entsteht (vgl. Abbildung 2.19). Um in dem verkleinerten

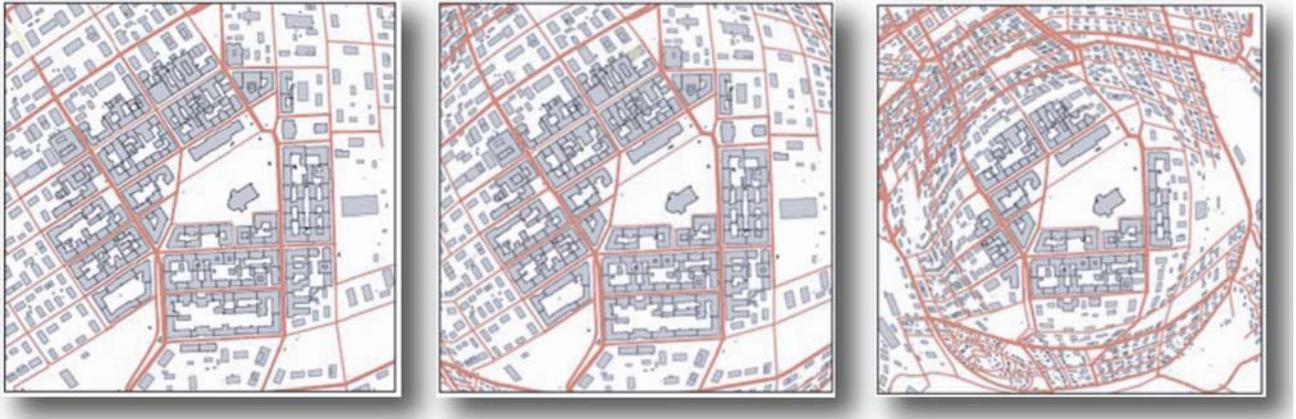


Abb. 2.19: Veränderung des Maßstabes innerhalb einer Karte durch Verzerrung der Darstellung (Fischaug) (aus Harrie et al. 2002)

Bereich am Rand der Karte auch den entsprechenden Mindestgrößen und -abständen gerecht zu werden, sollten diese gleichzeitig stärker generalisiert werden. Hier bietet sich die Verwendung vorgeneralisierter Daten innerhalb einer MRDB an.

Weiterhin werden während des Verkleinerns (oder Vergrößerns) eines Kartenausschnittes unterschiedliche Repräsentationen benötigt. Darstellungen sollten während dieser Vorgänge inhaltlich und grafisch entsprechend dem aktuellen Maßstab neu aufbereitet werden. Sester & Brenner (2004) beschreiben hierzu einen Ansatz zur kontinuierlichen Generalisierung während des Vergrößerns oder Verkleinerns. Es werden die jeweiligen Generalisierungsschritte, wie das Einfügen, Löschen oder Verschieben von Stützpunkten, für jedes Objekt gespeichert und bei Bedarf ausgeführt. Hierdurch ist es möglich, die Objekte kontinuierlich der Darstellung anzupassen. Gleichzeitig können zunächst für eine schnelle Darstellung weniger Details übermittelt werden und nach und nach mehr Feinheiten, respektive mehr Stützpunkte, eingefügt werden. Ein pragmatischer Ansatz zur Anpassung der Darstellung ist die Verwendung vorgeneralisierter Datensätze, welche idealerweise in einer MRDB vorgehalten werden.

2.6 Datenbanktechnische Aspekte eines Webservices

Ein Location Based Service für mobile Nutzer, eine einfache „Gelbe-Seiten“ Anfrage für Internetnutzer oder ein komplexer Routingservice bedingt immer die Bereitstellung von Inhalten aus einem eigenen Datenspeicher oder durch Integration von Inhalten Dritter. Daher kommt dem Serviceinhalt als Grundlage jeden Dienstes eine besondere Rolle zu. Das Schlagwort „*Content is king*“ findet sich in vielen Publikationen (vgl. Peng & Tsou (2003), Middleton (2002)). Norio Ohga, früherer CEO und Vorsitzender bei der Firma Sony, sagt: „*Without content, the network is nothing*“ (Schlender 2000).

Der Anbieter dieser Inhalte muss diese zunächst aus verschiedenen Quellen zusammenführen und seinerseits diese Informationen in sein System integrieren und dort unterhalten. Durch die Modularität des Systems ist es möglich, unterschiedliche Inhalte mit unterschiedlichen Services zu kombinieren. Die Informationen können in verschiedenen Diensten wiederverwendet werden, was den hohen Aufwand der Datensammlung relativiert und somit einen wichtigen ökonomischen Faktor für einen Serviceanbieter darstellt. Daher bedeutet die Datenintegration einen wichtigen Aspekt für zukünftige (mobile) Services (Hage et al. 2003).

Abbildung 2.20 zeigt die Wertschöpfungskette für Geoinformationen. Diese beginnt bei den Geoinformationsquellen, welche in eine interoperable Umgebung eintreten und Geoprozessierungsketten durchlaufen, die ein

mit Informationen angereichertes Zwischenprodukt erzeugen. Dieses Zwischenprodukt kann durch Vereinigung, Integration oder Kommentierung der Daten, durch Analyse und Modellierung entstehen. Schließlich wird das Endprodukt generiert, welches Geoinformationen enthält oder aus diesen abgeleitet wurde. Typischerweise finden in diesem Schritt die Visualisierung, Analyse, Übertragung und Verbreitung der Informationen statt (OGC 2003a). Es genügt also nicht, dem Anwender die Daten in ihrem Rohzustand zu präsentieren. Vielmehr sind mehrere Zwischenschritte zur Aufbereitung der Daten notwendig, damit diese schließlich den hohen Wert für den Nutzer erhalten.

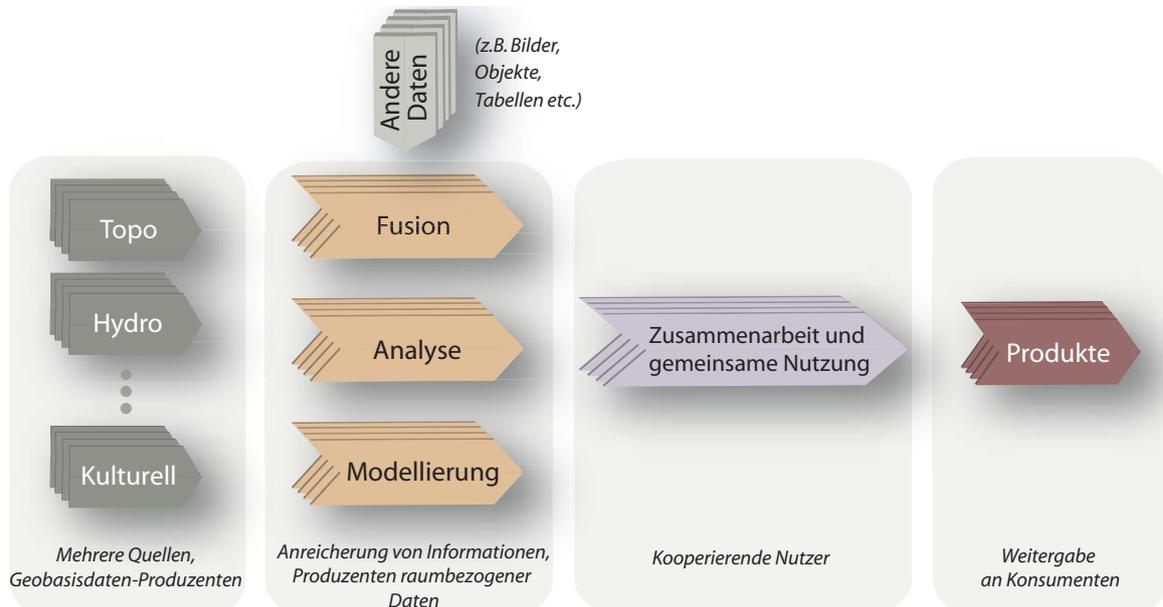


Abb. 2.20: Wertschöpfungskette Geoinformationen (aus OGC 2003a)

Der Aufbau und die Implementierung einer Datenkomponente für Webservices folgt dabei grundsätzlich den allgemeinen Regeln eines DBMS (vgl. Abschnitt 5.1). Trotzdem gelten für derartige Dienste und insbesondere für mobile Webservices eigene Regeln, die im Folgenden herausgestellt werden sollen. Der Webservice beziehungsweise die Applikation, welche die Funktionalität dieses Dienstes realisiert, fragt im Bedarfsfall Daten aus der Datenbank ab, wobei das DBMS seinerseits auf verschiedene Datenquellen zugreifen und diese integrieren kann. Alternativ wird ein Service durch Dienste Dritter mit Informationen angereichert.

Die Datenkomponente innerhalb eines Webdienstes wird üblicherweise durch einen Datenserver realisiert. Ein Datenserver besteht aus einer Rechner-Plattform, welche Datenbanken verwaltet. Diese beinhaltet Computer, Betriebssysteme, Netzwerke, DBMS, GIS und andere Hard- und Software, um Daten einzugeben, zu speichern, zu verwalten, abzufragen und auszugeben (Cowen et al. 1995). Ein Datenserver enthält also unter anderem die Daten und liefert diese durch Abfrage über eine Schnittstelle an einen Client. Die Nutzeranwendung kann diese Daten empfangen und Datenbankeinträge modifizieren. Die Abfrage der Daten erfolgt meist über eine SQL Schnittstelle. SQL wurde erstmals 1986 als ISO Standard festgelegt und in unregelmäßigen Abständen weiterentwickelt von SQL-89 über SQL-92 (SQL2) bis hin zu SQL-99 (SQL3) und schließlich SQL-2003. Aktuelle DBMS realisieren SQL-92 und zum Teil SQL-99, wohingegen SQL-2003 Funktionalitäten zukünftigen Anwendungen vorbehalten bleiben. Datenserver, welche die SQL als Abfrageschnittstelle anbieten, werden auch als SQL-Server bezeichnet. Die Applikationen der Webservices fragen die Informationen aus dem integrierten DBMS ab, während dieses relationale oder XML basierte Daten zurückliefert.

Die Besonderheit eines Webservices liegt in der Möglichkeit, von jedem System aus auf alle verfügbaren Services zugreifen zu können. Datenbanken sind in der Regel zentral gelagert und verwaltet und meist auf bestimmte Anwendungen und Nutzergruppen zugeschnitten. Oftmals werden Datenbanken innerhalb eines geschlossenen Netzwerkes einer bestimmten Institution eingesetzt. Zukünftige Konzepte sehen dagegen eine globale Sichtweise der Datenspeicherung vor, ähnlich dem Konzept der Webservices (Pfoser et al. 2003). Dies bedeutet für den Nutzer einen unendlich großen Pool an Informationen, auf den er zugreifen kann. Die Anwendung beziehungsweise der Nutzer ist nicht mehr beschränkt auf die auf eine spezielle Applikation zugeschnittene Datenbank. Vielmehr ermöglicht die Webservice-Architektur den Zugriff auf global verteilte Informationsquellen. Der Nutzer eines Webservices greift indirekt auf die an den Webservice angeschlossene Datenbank zurück. Services können also als eine Art vorbelegte Datenbankabfrage gedeutet werden. Des Weiteren erleichtert die Servicearchitektur die

Informationssuche. Da der Anwender oftmals nicht genau weiß, was er möchte, kann er auf Vorschläge des Systems reagieren. Kann eine Anfrage nicht exakt beantwortet werden, so wird diese, durch leichte, gelenkte Veränderungen der Anfrage, zu einem Ergebnis führen, welches der Anfrage in etwa entspricht. Tritt der Fall ein, dass ein Service eine Anfrage nicht beantworten kann, so besteht die Möglichkeit, diesen Fehler zu speichern und hieraus einen neuen Service aufzubauen (Pfoser et al. 2003). Derartige Systeme beschreiben den Wunsch einer zukünftigen Informationslandschaft, welche auf den bis hierher beschriebenen Technologien der verteilten Webdienste und Datenbanken aufsetzen kann.

Ein Webservice kann dabei natürlich nicht nur auf angeschlossene Datenbanken, sondern seinerseits auf andere Services zurückgreifen. Somit entsteht ein Netzwerk aus Webservices und Datenbanken. Die Architektur entspricht dabei einem föderalen System, welches sich durch autarke Komponenten (in diesem Fall Datenbanken oder Services) und der Integration in ein übergeordnetes System auszeichnet (vgl. hierzu Unterabschnitt 3.4.2).

Die Aufgabe des DBMS innerhalb einer solchen Architektur besteht in der Integration aller unabhängig voneinander vorgehaltenen Informationen in ein konsistentes System. Es genügt nicht, die vorhandenen Daten einfach in einem System zusammenzufassen und zu kombinieren. Die Datenintegration stellt dabei einen nicht trivialen Prozess dar. Allein auf dem Gebiet der Integration heterogener raumbezogener Daten finden sich zahlreiche Herausforderungen und entsprechende Publikationen (vgl. Abel et al. 1998, Devogele 2002, Devogele et al. 1998, Gösseln & Sester 2003, Parent & Spaccapietra 1998). Es geht hier um die Zuordnung homologer Objekte sowie der Beseitigung räumlicher Konflikte unter Berücksichtigung der lokalen Nachbarschaftsverhältnisse (Kampshoff 2005). Weitere Details zur Integration räumlicher Datenbestände finden sich im Abschnitt 3.4.

2.6.1 Datenbankinhalte

Gemäß den vorangegangenen Aussagen ist es möglich, über einen Webservice auf jede denkbare Datenquelle mit entsprechenden Inhalten zugreifen zu können. Besondere Anforderungen an Datenbankinhalte stellen hier vor allen Dingen mobile Dienste beziehungsweise die hierauf zugeschnittenen LBS.

Mobile Nutzer fragen typischerweise nach dem eigenen Standort oder der Lage und Entfernung zu einem bestimmten Ort von Interesse. Komplexere Anfragen lassen sich die Route dorthin berechnen. Viele Nutzer fragen nach einer Liste verfügbarer Ziele, entsprechend ihrer Präferenzen, vor allen Dingen jedoch entsprechend ihrer derzeitigen Position. So wird nach dem nächsten Hotel, der nächsten Tankstelle oder Bushaltestelle gesucht. Entsprechend der OpenLS Spezifikation des OGC (OGC 2005b) enthält ein typischer Geoservice für mobile Nutzer unter anderem Daten über das Straßennetzwerk, Navigationsinformationen, Karten, Richtungsinformationen, Adressen und Verkehrsdaten. Dieser werden durch Informationen Dritter angereichert.

Die Daten, welche in einem Webservice für mobile Clients relevant sind, können dabei in zwei Gruppen unterteilt werden. Zum einen sind geografische Informationen grundlegend für die meisten mobilen Anwendungen. Mit diesen geografischen Daten werden dann die eigentlichen Inhalte der Anwendung, auch als Geschäftsdaten bezeichnet, angereichert und verknüpft.

Geodaten Geografische Informationen geben Auskunft über die Umgebung, in der sich der Nutzer bewegt. Zur Navigation werden zum Beispiel Straßendaten benötigt, da der Nutzer sich nur entlang dieser bewegen kann. Routenplanungsservices verwenden grafbasierte Informationen sowie zusätzliche Attribute, wie Abbiegevorschriften oder aktuelle Informationen, wie die derzeitige Verkehrssituationen auf den Streckenabschnitten. Straßendaten (z.B. GDF-Daten⁹⁷) dienen zur Bestimmung von Entfernungen, um das nächstgelegene Restaurant zu finden. Die Lage dieses Restaurants gehört dabei zu den Geschäftsdaten, liegt in diesem Fall als Punktinformation vor und ist mit einem bestimmten Straßensegment auf Grund seiner Position verknüpft. Geografische Informationen dienen der Orientierung und können zusätzlich mit Geschäftsdaten verknüpft werden, wie beispielsweise der Funktion einzelner Objekte. Die geografische Infrastruktur ist grundlegend für mobile Services, denn auch wenn die applikationsspezifischen Inhalte vom Nutzer als die wichtigen Informationen angesehen wird, so werden diese erst über die geografischen Informationen fassbar. Dabei wird die geografische Information größtenteils in Form von Karten übermittelt.

Klassischerweise sind Geodaten Teil eines GIS. Dieses sammelt, verwaltet, verarbeitet, analysiert und präsentiert geografische Daten (Bill 2006). Die traditionellen Komponenten eines GIS sollten daher nicht außer Acht gelassen werden bei der Verwendung von Geodaten innerhalb eines Webservices. Bestehende Systeme verfügen über

⁹⁷Geographic Data Format: Ein von der Fahrzeugnavigations-Industrie entwickeltes, europäisches Standard-Dateiaustauschformat für vektorbasierte Straßenkarten (ISO/DIS 14825)

besondere Strukturen und Modelle zur Speicherung räumlicher Daten (z.B. *Oracle Spatial*, *IBM Spatial Extender* oder *PostGIS*, vgl. Unterabschnitt 5.2.3). Dabei sind neue Strategien notwendig, um Geodaten auf dieselbe einfache Weise übertragen und bearbeiten zu können, wie nicht-räumliche Daten. Standard-Entwicklungsumgebungen, wie zum Beispiel Java, müssen um grafische und räumliche Komponenten ergänzt werden (Virrantaus et al. 2001). Datenbanken sollten in der Lage sein, räumliche Objekte als solche zu speichern und zu verwalten. Darüber hinaus sind Analysefunktionen notwendig. Datenbanken müssen die Eigenschaften räumlicher Daten als Objekt und auch dessen räumlichen Bezug widerspiegeln. Abfragen richten sich nicht nur nach bestimmten Attributen, sondern vor allen Dingen nach geometrischen Eigenschaften, wie der Lage im Raum oder der Flächengröße. Diese Funktionalitäten sind in dem DBMS selbst und in den abfragenden Applikationen anzusiedeln.

Geschäftsdaten Geschäftsdaten dagegen sind serviceabhängige Informationen. Mobile Services geben Auskunft über „Points of Interest“, also der Lage von Orten, die für den Kunden von Interesse sind. Eine Idee der inhaltlichen Vielfalt eines mobilen Dienstes gibt Abschnitt 2.4. Interessant an dieser Stelle ist vor allen Dingen die Kombination von Geschäfts- und Geodaten. Geschäftsdaten besitzen in den meisten Fällen einen räumlichen Bezug und sollten daher mit den Geodaten in Beziehung gesetzt werden.

Die Geschäftsdaten mit Geobezug können dabei in zwei Kategorien unterteilt werden, den Punkt- und den Intervalldaten (Hage et al. 2003). Punktdaten umfassen Informationen, welche eine bestimmte Lage, aber keine entsprechende Ausdehnung aufweisen. Hierzu gehören beispielsweise Tankstellen, Museen, Hotels, Bushaltestellen oder Aussichtspunkte. Intervalldaten dagegen besitzen eine gewisse Ausdehnung, wie Geschwindigkeitsbegrenzungen oder Streckenabschnitte mit schöner Aussicht. Diese Geschäftsdaten sind dabei auf unterschiedlichste Weise mit den Geodaten verknüpft. Das Museum besitzt entweder eine Adresse, welche wiederum mit Straßen- und Gebäudedaten abgeglichen werden kann oder Koordinaten. Der Informationsumfang der Geschäftsdaten ist nahezu unbegrenzt. So kann ein Museum neben seiner Adresse auch Informationen über aktuelle Ausstellungen und Öffnungszeiten bereitstellen, das Hotel die Anzahl der aktuell verfügbaren Betten in bestimmten Kategorien, den Speiseplan des heutigen Abends oder Bilder der Anlage.

Dabei ist oftmals zu beobachten, dass Geoinformationen und Geschäftsdaten auf Grund ihrer unterschiedlichen Natur getrennt voneinander vorgehalten werden. Sinnvoll dagegen ist die Integration aller verfügbaren Informationen in einem einheitlichen System.

2.6.2 Qualität und Aktualisierung

Ein Qualitätsmodell wird festgelegt von der ISO9000 - ISO9004. Dabei beinhaltet dieses obligatorische Angaben über die Datenquelle, eine optionale Beschreibung der geplanten Verwendung der Daten und spezifische Qualitätsmerkmale.

Zu der Gruppe der primären Qualitätsmerkmale gehören nach Stanek (1994) und Caspary (1993) folgende Elemente:

- Positionsgenauigkeit
- Thematische Genauigkeit
- Aktualität
- Konsistenz
- Vollständigkeit

Demzufolge wird der eine oder andere Webservice nicht nur auf Grund seiner Quantität und Art der Inhalte favorisiert, sondern vor allen Dingen die Qualität der Daten stellt ein entscheidendes Kriterium zur Wahl eines Dienstes dar. Dabei haben Erfassungs- und Auswertemethoden Einfluss auf die Genauigkeit der Informationen. Dagegen liegen die Aktualität, Konsistenz und Vollständigkeit in der Verantwortung des Datenbankmanagements.

Gerade auf Grund der dynamischen Charakteristik, sowohl der Geoinformationen als auch der in vielen Fällen noch schnelllebigeren Geschäftsdaten wie Kinoprogramm, Verkehrsdaten oder anderen PoI-Informationen, sind CD-ROM Lösungen, welche eine Momentaufnahme darstellen, schnell veraltet und für den Nutzer uninteressant. Die Alternative, über eine Online-Datenverbindung entsprechende, aktuelle Informationen aus der Datenbank abfragen zu können, ist für die meisten Anwendungen wesentlich attraktiver.

Nicht aktuelle Informationen sind bei entsprechender Dynamik dieser auch gleichzeitig unrichtig. Somit ist es wichtig, eine angemessene Aktualität der Daten sicherzustellen. Die erforderliche absolute Aktualität, also die Differenz zwischen dem Erhebungsdatum und dem aktuellen Datum, ist dabei abhängig vom Datentyp. Idealerweise

wird eine Veränderung in dem Moment eingespielt, in dem diese in der Realität auftritt oder sogar vorher. Dabei gibt es kurzlebige Ereignisse wie Verkehrsstaus oder Unfälle, die möglichst zeitnah eingespielt oder vorhergesagt werden sollten. Diese werden oftmals von Dritten, welche von dem Serviceanbieter eine Datenbankautorisierung erhalten haben, bereitgestellt. Auch semantische Änderungen, wie geänderte Öffnungszeiten, sollten möglichst schnell verfügbar sein und in kurzen Zeitabständen überprüft werden.

Die Aktualisierung und Überprüfung von Geodaten dagegen ist zeit- und vor allen Dingen kostenintensiver und wird daher im Regelfall in größeren Abständen durchgeführt. So sieht das Konzept des ATKIS⁹⁸ eine Spitzenaktualität der Daten vor, was konkret eine Aktualisierung nach drei, sechs und zwölf Monaten, je nach Objektart und Attribut, bedeutet (Harbeck 2000).

Ein Datenbanksystem sollte dabei so aufgebaut sein, dass Veränderungen nur an einer Stelle vorzunehmen sind. Gleichzeitig können Veränderungen, die Auswirkungen auf andere Datenbankelemente haben, automatisch auf diese übertragen werden. Dabei genügt es nicht, die alten Datenbestände durch die neuen zu ersetzen. Vielmehr besteht der Bedarf einer Datenintegration, was bedeutet, dass die Daten aneinander angepasst werden und Unstimmigkeiten beseitigt werden müssen. So stammen die Grunddatenbestände meist aus analogen Datenquellen, welche digitalisiert und in ein einheitliches System transformiert wurden. Die Aktualisierungen dagegen entstehen aus genaueren, geodätischen oder photogrammetrischen, Aufnahmen. Diese neuen Daten müssen geometrisch in die bestehenden Nachbarschaften eingepasst werden (Kampshoff 2005, S.3). Gleichzeitig muss sichergestellt werden, dass geometrische Zusatzinformationen, welche aus Quellen Dritter stammen und zusammen mit den Geobasisdaten präsentiert werden, auch nach der Aktualisierung eine topologisch korrekte Lage besitzen. Eine Tankstelle sollte auch nach der Aktualisierung eines Straßensegmentes auf der richtigen Straßenseite zu finden sein.

Daneben verarbeiten viele mobile Webservices auch kontinuierliche Veränderungen in ihrem System. So wird in verschiedensten Applikationen die Position von Objekten verfolgt. Diese Position wird von dem Dienst erfasst und verarbeitet. Die Herausforderung besteht darin, relevante Positionsveränderungen zu erkennen. Bewegungen können dabei durch Messungenauigkeiten vorgetäuscht werden oder stellen eine nicht relevante Veränderung dar. Die Genauigkeit und Frequenz der Positionsbestimmung sind daher abhängig von der aktuellen Anwendung und dem aktuellen Nutzer respektive seiner Geschwindigkeit. Soll ein Nutzer zu einem bestimmten Ziel navigiert werden, so ist es notwendig, eine Positionsveränderung von wenigen Metern zu erfassen und zu erkennen. Besteht die Dienstleistung dagegen lediglich in der Anzeige aktueller Wetterinformationen, so genügt es, nur Positionsveränderungen von mehreren Kilometern zu erkennen.

Auf Grund der Dynamik mobiler Nutzer veralten diese Informationen nicht nur in den Datenbanken sondern auch auf dem Gerät des Nutzers. Daten, welche an der einen Stelle relevant waren, sind an der neuen Position schon nicht mehr zu gebrauchen und müssen aktualisiert werden. Verwendet der Nutzer beispielsweise zur Orientierung eine Karte, auf der seine aktuelle Position angezeigt wird, so wird durch eine Veränderung der Position ein neuer Kartenausschnitt notwendig. Gerade auf Grund der geringen Kapazitäten mobiler Geräte (vgl. Unterabschnitt 2.4.1) kann nur ein Kartenausschnitt bestimmter Größe im Speicher des Gerätes vorgehalten werden. Die Karte wird daher in Zellen unterteilt und je nach Position und Bewegungsrichtung die entsprechende Zelle in den Cache des mobilen Gerätes geladen (Andersen et al. 2003).

2.7 Zusammenfassung

Webservices bieten die Möglichkeit, ein Netz von Daten und Diensten aufzubauen, auf das jeder Computerbesitzer zugreifen kann, vorausgesetzt er verfügt über einen Internetzugang. Webdienste sind unabhängig von der aufrufenden Plattform und auch unabhängig von der Implementierung, dank definierter Schnittstellen. Webservices stellen somit eine für die Zukunft wichtige Strategie der IuK unter dem Stichwort „global computing“ dar. Sie erlauben einen plattformunabhängigen Zugriff auf die Daten und Dienste. Beides wird global nutzbar und somit aus ökonomischer Sicht weitaus effizienter, da einmal erfasste Daten und aufgebaute Prozesse mehrfach genutzt und vermarktet werden können. Anwender ersparen sich auf der anderen Seite eine aufwändige Datenerfassung sowie Implementierung von Applikationen. Weiterhin besteht die Möglichkeit der Verkettung der Dienste, so dass jeder Anwender diese je nach Bedarf kombinieren kann.

Auch die Geoinformatik bedient sich dieser serviceorientierten Methoden. Daten und Dienste werden über Austausch-Schnittstellen verfügbar. Dabei hat die Entwicklung spezieller Interfaces (des OGC und der ISO) für den Austausch und die Abfrage von Geoinformationen stark dazu beigetragen, dass diese Technologie immer

⁹⁸ Amtlich Topographisch-Kartographisches Informationssystem

mehr Anklang bei Datenanbietern und Nutzern findet. Die Bereitstellung zahlreicher Webmap-Services sowie die vielfältigen Möglichkeiten der LBS für den mobilen Nutzer sind hier nur zwei Beispiele. Diese neuen Technologien bereiten den Weg für neue Geschäftsfelder der Daten- und Serviceanbieter sowie neuartige Nutzungsmöglichkeiten für den Anwender. Professionelle Nutzer erhalten auf einfachste Weise Zugriff auf benötigte Daten und Dienste, ortsbezogene Dienste bieten darüber hinaus für den Privatanbieter nützliche Services, welche den Alltag erleichtern.

Geodaten bilden dabei oftmals die Grundlage dieser Dienste. Der Nutzer greift indirekt über die Applikationsebene auf die angeschlossenen Datenbanken zurück. Somit ist es möglich, vielen Anwendern die bestehenden Datenbankinhalte zur Verfügung zu stellen. Zusätzlich kann der Service selbst auf andere Services und somit auf andere Datenquellen zugreifen. Werden nun verschiedenste Datenquellen in einem Serviceangebot vereint, so müssen diese in einem konsistenten System integriert werden. Dabei bietet die Integration mehrerer Datenquellen in einer MRDB enorme Vorteile, welche auch innerhalb der Webservices genutzt werden können. Gleichzeitig werden Methoden zur Verarbeitung dieser Daten benötigt. Viele bestehende Elemente der Geoinformatik können hierzu verwendet und auf diesen Bereich übertragen werden. GIS-Funktionalitäten können genutzt werden, um ortsbezogene Problemstellungen zu lösen und die Frage nach dem nächsten Restaurant beantworten oder die Route und Entfernung von A nach B berechnen. Auch die Techniken zur Speicherung von Geodaten in räumlichen Datenbanken können innerhalb einer Webservice-Architektur weiter verwendet werden, wenngleich deren Inhalte an die neuen Möglichkeiten angepasst werden müssen.

Darüber hinaus gilt es jedoch auch, neue Methoden zu entwickeln, welche diesem neuartigen Gebiet Rechnung tragen. So muss die geografische Information zufriedenstellend präsentiert werden. Dazu genügt es nicht, die Prinzipien der traditionellen Kartografie unverändert zu übernehmen. Vielmehr sind neue Konzepte zur Darstellung der Daten beispielsweise für die kleinen Displays der mobilen Geräte gefragt. Den Limitierungen mobiler Geräte muss durch neue Formen der Informationsübermittlung entgegengewirkt werden. Der Trend geht zur Adaption der Darstellung, unwichtige Inhalte werden weggelassen, andere betont. Hierin besteht gleichzeitig eine neue Chance. Die Inhalte können individuell den Erfordernissen der einzelnen Nutzer angepasst werden, zugleich muss der Prozess der Informationsaufbereitung drastisch beschleunigt werden im Vergleich zur Herstellung gedruckter Kartenwerke. Der Anwender erwartet eine Antwort auf seine Anfrage innerhalb weniger Sekunden.

3 Grundlagen: MRDB

Eine „Multiple Representation Database“ (MRDB) oder übersetzt „Multiple Repräsentationsdatenbank“ bezeichnet im Allgemeinen die Zusammenführung unterschiedlicher Repräsentationen von Phänomenen in einem Datenbank-Managementsystem (DBMS). Dieses Kapitel befasst sich ausführlich mit den Gründen zur Einführung einer solchen Datenstruktur, den hiermit verbundenen Problemen sowie verschiedenen Ansätzen zur Lösung dieser. Zunächst erfolgt eine Definition der MRDB sowie die Übertragung dieses Begriffes auf den Bereich der Kartografie. Anschließend werden zwei Projekte vorgestellt, welche sich ausführlich mit der Thematik der MRDB auseinandergesetzt haben. Eine Initiative des NCGIA¹ stellt dabei einen Startpunkt dieser Forschungsrichtung dar und definiert die ursprünglichen Probleme und Forschungsfragen. Das vor wenigen Jahren abgeschlossene EU-Projekt *MurMur*² spiegelt die heute bestehenden Möglichkeiten und Probleme wider, wenn es darum geht, eine MRDB aufzubauen und zu verwenden.

Im Abschnitt 3.3 folgt ein Überblick über die Anwendungsfelder, welche durch die Nutzung einer MRDB ermöglicht oder zumindest erleichtert werden. Eine MRDB schafft Vorteile bei der Datenhaltung, aber auch neue Möglichkeiten im Bereich der Datenanalyse. Sie kann notwendige Generalisierungsvorgänge unterstützen oder gar ersetzen, was vor allen Dingen für den Bereich der Visualisierung von Geodaten interessant und wichtig ist.

Der Abschnitt 3.4 beschäftigt sich mit den Schritten zur Realisierung einer MRDB durch Integration heterogener Datensätze. Zunächst gilt es, ein entsprechendes Datenbankschema zu schaffen, in das die vorhandenen oder abzuleitenden Repräsentationen integriert werden können. Die Besonderheit besteht darin, dass hier ein bestimmtes Objekt oder Phänomen durch *mehrere* Instanzen repräsentiert werden kann. Es gilt eine Beschreibung zu finden, die allen zu integrierenden Datensätzen genügt und gleichzeitig die Zusammenhänge zwischen unterschiedlichen Repräsentation modelliert. In einem weiteren Schritt erfolgt die Schemaintegration, also die Integration der einzelnen Schemata in das übergeordnete Schema, sowohl auf semantischer als auch auf der Objektebene. Darüber hinaus gilt es zur Erhaltung der Konsistenz des Systems gewisse Mechanismen aufzustellen, um Veränderungen einzelner Repräsentationen auf die übrigen Repräsentationen zu übertragen.

Schließlich genügt es nicht, ein integriertes System aufzubauen, welches multiple Repräsentationen beinhaltet. Es muss dem Anwender durch entsprechende Zugriffsmöglichkeiten die Gelegenheit gegeben werden, die Vorteile eines solchen Systems für seine Zwecke auszuschöpfen. Daher werden im Abschnitt 3.5 notwendige Funktionalitäten beschrieben, welche in das DBMS oder eine entsprechende Nutzerschnittstelle zu integrieren sind.

3.1 Definition und Probleme

Werden Objekte oder Phänomene der Umwelt in einer Datenbank abgebildet, so spiegelt sich in fast allen Fällen nicht etwa die Realität wider, sondern die Sichtweise und Intention desjenigen, der die Daten erfasst und gespeichert hat. Die Mehrfacherfassung der Daten ist auf unterschiedliche Gründe zurückzuführen. Neben der Unkenntnis über die Existenz bereits erfasster Daten auf Grund von Koordinationsproblemen folgt dies vor allen Dingen aus der Heterogenität der semantischen sowie der technischen Aspekte. Die semantische Diversität fußt auf den unterschiedlichen Anforderungen der Applikationen, technische Unterschiede sind in den einzelnen Systemen des Datenmanagements zu finden (Balley et al. 2004). Das oberste Ziel ist daher darin zu sehen, aus den komplementären und sich überlappenden Datensätzen einen konsistenten Informationssatz für ein bestimmtes Gebiet zu gewinnen. Das Stichwort an dieser Stelle heißt Interoperabilität. Jeder Datensatz beinhaltet eine spezifische Repräsentation der realen Welt, beeinflusst durch die explizite Sichtweise und die Interessen der angeschlossenen Applikationen. Parent (2000) charakterisiert mögliche Unterschiede in den Eigenschaften einer Repräsentation:

- Unterschiedliche Informationen werden erfasst und gespeichert: Die relevanten Objekte sind abhängig von der jeweiligen Anwendung

¹National Center for Geographic Information

²Multiple representations multiple resolutions

- Die Informationen werden auf unterschiedliche Weise beschrieben: Dasselbe Objekt mag für verschiedenste Anwendungen interessant sein, trotzdem sind jeweils unterschiedliche Eigenschaften von Relevanz
- Die Informationen können unterschiedlich organisiert sein (im Hinblick auf die Datenstruktur): Phänomene können als eigenständiges Objekt, aber auch einfach nur als Attribut eines übergeordneten Objektes gesehen werden
- Die Information kann auf unterschiedliche Weisen codiert werden: Koordinaten eines Punktes können in dem einen oder anderen Koordinatensystem gehalten werden
- Informationen können auf unterschiedliche Weise repräsentiert werden: Geodaten können beispielsweise auf unterschiedliche Arten signaturiert werden

Allgemein resultieren unterschiedliche Repräsentationen aus einer bestimmten Sichtweise (view) des Datenbankautors und auch des Datennutzers. Jeder Anwender und jede Anwendung selektiert und betrachtet Phänomene der Realität auf eine bestimmte Art. Die Sichtweise beeinflusst jeweils die geeigneten Datenstrukturen und -werte als auch den Umfang und den Grad an Details. Nach Balley et al. (2004) sollte neben dem allgemein und auf alle Bereiche anzuwendenden Begriff der „Sichtweise“ der Begriff der „Auflösung“ gesondert behandelt werden sollte. Somit ergeben sich zwei Dimensionen:

- Die *Sichtweise* kennzeichnet die Art und Weise, wie eine Applikation auf die Daten schaut: Diese filtert die interessanten Daten und beschreibt, wie diese dargestellt und gespeichert werden
- Die *Auflösung* bestimmt den gewählten Detaillierungsgrad, um nur die relevanten Teile eines Phänomens in der Datenbank zu erhalten (Stell & Worboys 1998). Geografische Auflösung wird hier als minimale Größe eines Objektes definiert, welches in der Datenbank gespeichert wird. Die semantische Auflösung bestimmte den Grad semantischer Details in den Daten

In der Kartografie sind unterschiedliche Repräsentationen eng verknüpft mit unterschiedlichen Maßstäben, in denen die Geodaten repräsentiert werden. Die Kartenreihen der Landesvermessungen beispielsweise werden in verschiedenen Maßstäben aufgelegt, damit möglichst alle Anforderungsbereiche abgedeckt sind. Um diese unterschiedlichen topografischen Kartenwerke herstellen zu können, werden in den Datenbanken für jeden Maßstabsbereich eigene Datensätze und -modelle vorgehalten (z.B. ATKIS DLM³25, DLM50, DLM250 und DLM1000 für die deutsche Landesvermessung). Ist der Begriff Maßstab weitestgehend treffend, wenn es um Karten geht, so ist dieser weniger angemessen, wenn Geodaten gemeint sind, welche in Datenbanken vorliegen. Die Definition des Maßstabes als Verhältnis zwischen dem Abbild und der Realität greift hier nicht mehr. Devogele et al. (1996) beschreiben daher eine *multiskalige* Datenbank als eine geografische Datenbank, welche es uns erlaubt, dasselbe Phänomen der realen Welt in unterschiedlichem Grad an Präzision, Genauigkeit und Auflösung zu repräsentieren. Dabei definieren Goodchild (1991) und Müller et al. (1995) die Begriffe wie folgt:

- Präzision: Grad an Details bei der Auswertung einer Messung
- Genauigkeit: Zusammenhang zwischen einer Messung und der Realität, welche diese zu repräsentieren beabsichtigt
- Auflösung: Das kleinste Element, welches dargestellt werden kann

Kilpeläinen (1997) beschreibt in ihrer Arbeit unter anderem die Möglichkeit, verschiedene *Auflösungsstufen* in einer Datenbank zu unterhalten und diese bei Veränderungen inkrementell zu aktualisieren. Die Idee beruht auf dem Prinzip, die Repräsentation mit der höchsten Auflösung nachzuführen, um anschließend die nachgeordneten Auflösungsstufen automatisch anpassen zu können. In diesem Zusammenhang besteht eine MRDB demzufolge aus Stufen unterschiedlicher Auflösung, wie sie zur Darstellung verschiedener Kartenmaßstäbe Verwendung finden. *Kilpeläinen* definiert eine MRDB durch folgende Eigenschaften (siehe Abbildung 3.1):

- Die Repräsentations-Level bestehen aus verschiedenen Repräsentationen desselben Objektes, was bedeutet, dass die geometrische Repräsentation sich von einer Ebene zur nächsten ändern kann
- Die Repräsentationen der verschiedenen Level besitzen Verknüpfungen untereinander

Sowohl die verschiedenen Repräsentationen, welche in der Datenbank gespeichert werden, als auch die Repräsentationen, die über bestimmte Regeln aus den bestehenden abgeleitet werden können, sind Teil des Konzeptes einer MRDB. In diesem Zusammenhang wird der Begriff *konzeptionellen Auflösungsstufen* dem der verschiedenen *Maßstäbe* vorgezogen, da die Daten hier unterschiedlichen Applikationen dienen sollen und nicht nur vordefinierten Maßstäben. Eine MRDB soll also verschiedene Auflösungen für unterschiedliche Anwendungen bereitstellen, unabhängig davon, ob diese Daten real in der Datenbank gespeichert sind oder nur konzeptionelle

³Digitales Landschaftsmodell

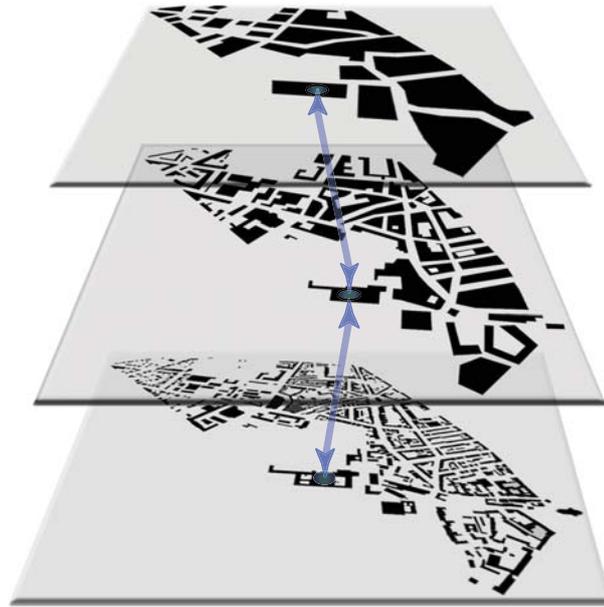


Abb. 3.1: Definition einer MRDB: Verschiedene Repräsentationen und Verknüpfung korrespondierender Objekte

Regeln vorhanden sind, um diese aus bestehenden Daten abzuleiten. Des Weiteren bestehen Relationen zwischen den einzelnen Objekten oder Objektgruppen (vgl. Abbildung 3.1). Hier müssen nach *Kilpeläinen* die *Beziehungen* (relationships) zwischen den Objekten derselben Auflösungsstufe, so zum Beispiel die topologischen Beziehungen, von den *Verbindungen* (connectivities) zwischen den Objekten in den unterschiedlichen Auflösungsstufen unterschieden werden.

Die Variation der Auflösung hat dabei Einfluss auf die Charakteristika der räumlichen Objekte (Parent et al. 2000):

- Vereinfachung der Form bei geringer werdender Auflösung und zusätzlich unterschiedliche Geometrietypen; aus einer Fläche wird beispielsweise ein Punktobjekt oder eine Linie
- Veränderung der semantischen Auflösung
- Veränderung der Existenz der Objekte: Unterschiedliche Abstraktionsebenen bedeuten eine Koexistenz zwischen einer Gruppe von Objekten und deren Aggregationen

3.2 Vorausgegangene Arbeiten auf dem Gebiet der MRDB

In dem folgenden Abschnitt soll zunächst ein Überblick über bisherige Forschungstätigkeiten auf dem Gebiet der MRDB erfolgen. Daher werden zunächst zwei Forschungsprojekte näher beschrieben, welche die umfassendsten und weitreichendsten Arbeiten hierzu hervorgebracht haben. Dies ist zum einen eine Forschungsinitiative des NCGIA, die quasi die Grundlage aller Forschungsarbeiten darstellt sowie das EU-Projekt *MurMur*, das jüngste, großangelegte Projekt mit dieser Thematik. Weitere Arbeiten, welche sich jeweils mit Teilproblemen der MRDB beschäftigen, werden in den jeweiligen Unterkapiteln beschrieben.

3.2.1 Eine Forschungsinitiative des NCGIA

Eine Initiative des NCGIA unter dem Titel „Multiple Representations“ stellt den Startpunkt für das genannte Forschungsgebiet dar. Die Arbeitsgruppe beschäftigte sich in dem Zeitraum von 1988 bis 1990 erstmalig gezielt mit diesem Thema. Ziel war es unter anderem, die zu dem damaligen Zeitpunkt unter vielen Kategorien verstreut zu findenden Forschungen unter dem Titel „Multiple Representation“ zusammenzufassen und so die Forschungsarbeiten auf diesem Gebiet zu zentrieren und gleichzeitig einen Überblick zu erhalten über den Forschungsstand. Der Begriff umfasste in diesem Zusammenhang „die Veränderungen der geometrischen sowie topologischen Strukturen digitaler Objekte, bedingt durch die Variation der Auflösung der Objekte zum Zwecke der Speicherung, Analyse und Visualisierung“ (Buttenfield & Delotto 1989). Es sollten Probleme

aufgedeckt und Forschungsschwerpunkte definiert werden, um weltweit Forscher für die benannten Sachverhalte zu sensibilisieren und diese zu motivieren, diese Probleme zu lösen. Die Initiative selbst beschäftigte sich nicht mit der Implementierung einer MRDB, sondern konzentrierte sich auf theoretische, konzeptionelle Arbeiten.

Zu Beginn der Initiative wurden mögliche Probleme durch Einführung multirepräsentativer Datenschemata aufgedeckt. Buttenfield & Delotto (1989) fassen die Ergebnisse dieser Diskussionen in ihrem Bericht zusammen.

Die Speicherung unterschiedlicher Auflösungen desselben Objektes wurde zunächst als Behinderung effizienter Operationen von GIS sowie der Qualität kartografischer Produkte angesehen, da Methoden fehlen, um verschiedene Repräsentationen zu speichern und zu pflegen. Diese Probleme spiegeln sich in den Bereichen der Datensammlung, der Ableitung von Karten unterschiedlicher Maßstäbe aus einer einzelnen Datenquelle sowie den Analyse- und Abfragemöglichkeiten wider.

Konsistenzprobleme entstehen durch die Verwendung unterschiedlicher Datenquellen, aus denen die Geodaten extrahiert werden. Schon in der Stufe der höchsten Auflösung treten Geodaten unterschiedlicher Herkunft, wie Luftbildern oder anderen Quellen der Fernerkundung, Ergebnissen von Vermessungsarbeiten, existierenden Karten etc. und somit auch unterschiedlicher Auflösung und Qualität auf. Das Konzept einer MRDB dagegen erwartet in jeder Repräsentationsebene einen konsistenten Datensatz. Als größte Schwierigkeit offenbart sich die Erhaltung der Konsistenz gerade während einer Datenaktualisierung. Daher sollten in Zukunft Spezifikationen sorgfältig entworfen und klar dokumentiert werden, um Inkonsistenzen zu minimieren.

Mehrfachrepräsentationen bedingen außerdem einen Verwaltungs-Overhead durch mehrere Versionen eines Datensatzes in der Datenbank. Die Kosten für ein Datenbank-Update werden vervielfacht, wenn keine Methoden zur automatischen Weitergabe von Aktualisierungsinformationen entwickelt werden. Das höhere Speichervolumen wurde hier als wenig kritisch angesehen. Objektbeschreibungen in den verschiedenen Auflösungsstufen sowie deren Verknüpfungen müssen formalisiert werden, um Veränderungen im Datenbestand auf andere Ebenen automatisch übertragen zu können.

Ein weitere, grundlegende Frage, welche sich jeder Nutzer einer MRDB stellen muss: In welchen Auflösungen sollten die Repräsentationen vorgehalten werden? Es wäre unmöglich oder zumindest unendlich kostenintensiv, die Informationen für alle möglichen Maßstäbe aufzubereiten und vorzuhalten. Wie groß ist der Maßstabsbereich, den ein bestimmter Datensatz versorgen kann? Dieser ist sicherlich abhängig von dem jeweiligen Anwendungsgebiet. Cecconi (2003) beantwortet in seiner Arbeit diese Frage zum Beispiel für einen Kartendienst, welcher bestehende Repräsentationen und online-Generalisierungsprozesse kombiniert und auf diese Weise einen Maßstabsbereich von 1:25k bis 1:200k abdeckt. In diesem Fall wird eine neue Repräsentation benötigt, sobald das Generalisierungsmodul außer Stande ist, einen noch kleineren Maßstab zu produzieren.

Schließlich wurde der Einfluss multipler Repräsentationen auf den Nutzer und die Nutzerschnittstelle angesprochen. Wie kann ein für eine bestimmte Situation oder Aufgabe angemessener Maßstab bestimmt werden? Und wie kann vermieden werden, dass Nutzer Operationen auf der Datenbank durchführen, welche zu Fehlinterpretationen führen, indem beispielsweise unterschiedliche Repräsentationen als gleichwertig angesehen und synonym verwendet werden? Weiterhin sollten Informationen über die Datenqualität in das Datenmodell integriert werden. Diese Fragen werden teilweise im weiteren Verlauf dieser Arbeit aufgegriffen und es soll versucht werden, diese zu beantworten.

Eine Evaluierung bestehender Systeme und Arbeiten deckte primären Forschungsbedarf auf den Teilgebieten der Datenbanktechnik, der Generalisierung sowie der räumlichen Modellierung auf. Diese drei Gebiete stellen dabei nicht die einzigen Forschungsthemen dar, besaßen jedoch zu dem damaligen Zeitpunkt die höchste Priorität. Auf dem Gebiet der Datenbanken wurden Modelle benötigt, welche einen effizienten Zugriff sowie die Implementierung der Verknüpfungen zwischen den einzelnen Repräsentationen realisieren. Weiterhin waren Regeln zu entwickeln, um die Konsistenz in der Datenbank zu erhalten. Die Kartengeneralisierung sollte zukünftig weiter formalisiert werden, um diesen Prozess automatisieren zu können.

Hieraus folgten zunächst grundlegende Arbeiten und Untersuchungen zur Maßstabsabhängigkeit der Geometrien (Buttenfield & Delotto 1989), dem konzeptionellen Vorgehen in der Generalisierung (Mark 1989), hierarchischen Datenstrukturen (Bruegger & Frank 1989), der Formalisierung von Verknüpfungen in Datenbanken (Bruegger & Kuhn 1991) sowie dem konzeptionellen Rahmen für geografisches Wissen (McMaster 1991). Weitere hieraus resultierende Forschungsbereiche, von denen einige im weiteren Verlauf dieser Arbeit vertieft dargestellt werden sollen, umfassen die Modellierung und Abfrage von Datenbanken mit multiplen Auflösungsstufen (Rigaux & Scholl 1994), Daten- und Wissensmodellierung für die Generalisierung (Ruas 1995), objektorientierte Modelle zur Verarbeitung multipler Repräsentationen (Kidner & Jones 1994), die Entwicklung von Strukturen, um die Repräsentationen miteinander zu verknüpfen (Devogele et al. 1996), Datenbankstrukturen für multiskalige GIS (Jones et al. 1996) sowie die Erhaltung der Konsistenz zwischen multiplen Repräsentationen räumlicher Daten (Egenhofer et al. 1994, Tryfona & Egenhofer 1997, Kilpeläinen 1997).

3.2.2 Das EU-Projekt MurMur

Ein weiteres Forschungsprojekt auf diesem Gebiet stellt das EU-Projekt *MurMur* dar, welches in den Jahren 2000 bis 2002 aktiv gewesen ist. Innerhalb dieses Projektes wurde das Ziel verfolgt, die Funktionalitäten aktueller Software zur Datenverarbeitung (DBMS oder GIS) zu erweitern, um Multirepräsentations-Schemata, genauer gesagt Mehrfachrepräsentationen unterschiedlicher geometrischer und zeitlicher Auflösung, auf flexible Weise zu unterstützen (Spaccapietra et al. 1999). Bestehende Datenbankmodelle wurden angepasst an die erweiterten Anforderungen multirepräsentativer Daten und ein GIS wurde mit neuen Funktionalitäten ausgestattet. Die Arbeit in diesem Projekt konzentrierte sich insbesondere auf die Modellierung multipler geografischer Daten. Diese Motivation resultiert daraus, dass die Diversität der Nutzerprofile in Anwendungen von Geodaten weitaus breiter ist als in konventionellen Datenbankanwendungen, welche in den meisten Fällen auf einen kleinen, spezifischen Nutzerkreis zugeschnitten sind.

Eine wichtige Erkenntnis ergab sich in dem Projekt durch eine a priori durchgeführte Analyse und Bewertung existierender GIS und DBMS hinsichtlich der Unterstützung multipler Repräsentationen. Dabei stellte sich heraus, dass die bestehenden Systeme nur wenig Unterstützung im Umgang mit Mehrfach-Repräsentationen bieten. Sowohl die Datenbanksysteme als auch die GI-Systeme überlassen diese Verantwortung übergeordneten Anwenderprogrammen (Balley et al. 2004).

Ziel war es daher, ein neues Datenmodell zu entwickeln, welches auf ein vorhandenes Modell aufsetzt und dieses um die Fähigkeit einer multiplen Darstellung erweitert. Dabei sollte sich das zu entwickelnde Konzept möglichst nah an den Anforderungen der zukünftigen Nutzer orientieren. Die speziellen Fähigkeiten richten sich nach den Bedürfnissen zweier Nutzergruppen aus den Bereichen „Kartografie“ und des „Risikomanagements“. Das erste Szenario beschreibt die Anwendung aus der Sicht eines Kartografen, angesiedelt bei einer nationalen Landesvermessung. In dem beschriebenen Fall handelt es sich um die französische Landesvermessungsbehörde IGN⁴. Die Landesvermessung hält Geodaten in verschiedenen Maßstäben vor, wobei jeder Datensatz unabhängig von den anderen Datensätzen erfasst und verwaltet wird. Um den Aufwand zur Datenerfassung und -haltung zu minimieren, sollten zukünftig alle Datensätze in einem System unterhalten werden, welches die Daten weniger maßstabsorientiert als vielmehr objektorientiert betrachtet. Der Nutzer, in diesem Fall der Kartograf, digitalisiert die neu erfassten Informationen für den größten Maßstab und führt diese einer einheitlichen Datenbank zu. Hier werden die Daten auf Konsistenz überprüft und anschließend die Änderungen automatisch den kleineren Maßstäben zugeführt. Zu einem späteren Zeitpunkt können hieraus kartografische Produkte abgeleitet werden, die auf diese Daten zugreifen. Der zweite Anwendungsfall befasste sich mit dem Risikomanagement: Es existieren verschiedene Karten des selben Gebietes, die für das Risikomanagement relevant sind, wie Risikokarten, welche alle Arten von Ereignissen aufzeigen, wie Bergrutschungen, Flutereignisse, Lawinen, Steinschläge etc. Daneben werden Gebietskarten verwendet, welche die Erdoberfläche unterteilen in Gebiete, die nicht, solche, die nur unter bestimmten Voraussetzungen sowie Gebiete, die jederzeit bebaut werden dürfen. Als dritte Informationsquelle sind die Landnutzungskarten zu nennen. Diese und andere Karten repräsentieren demzufolge dasselbe Gebiet der Erdoberfläche, jeweils mit unterschiedlicher Sichtweise und Priorität. Da das Risikomanagement im Idealfall auf alle Informationen gleichzeitig zugreifen will, sollten die vorhandenen Karten in ein einheitliches System integriert werden. Unterschiede bestehen hier im Maßstab, der Genauigkeit und Auflösung sowie dem Zeitpunkt der Erfassung. Ein zu entwickelndes System sollte in der Lage sein, diese Unterschiede zu verarbeiten und zu harmonisieren.

Das Projekt entwickelte einen theoretischen Rahmen zur Nutzung und Verarbeitung von Vektordaten unterschiedlicher Auflösungsstufen sowie eine prototypische Implementierung. Es wurden Lösungen in den Bereichen der Datenmodellierung, des Datenmatchings sowie der Datennutzung gezeigt (vgl. Balley et al. 2004). Wie können also verschiedene Repräsentationen in einem Vektor-GIS modelliert werden? Wie können unterschiedliche Repräsentationen miteinander verschnitten werden, damit das DBMS weiß, dass diese dasselbe geografische Phänomen repräsentieren? Diese beiden Punkte können auch durch den Begriff der Datenintegration, also der Zusammenfassung verschiedener Datenquellen in einem System, beschrieben werden. Und schließlich die Frage: Welchen Nutzen bringen Systeme, die multiple Repräsentationen unterstützen, dem Anwender?

In der vorliegenden Arbeit stellen sich ähnliche Fragen, hier allerdings vor dem Hintergrund einer Webanwendung, so dass die Entwicklungen auf den theoretischen Ergebnissen dieses Projektes aufbauen sollen. Vor allen Dingen die Vorschläge zur Datenmodellierung sowie den grundlegenden Funktionalitäten einer derartigen Systems sind für die vorliegende Arbeit interessant und werden in Abschnitt 3.4 sowie Abschnitt 3.5 detaillierter vorgestellt.

⁴Institut Geographique National

3.3 Nutzen einer MRDB

Multiple Repräsentationen sind laut Bruegger & Frank (1989) ein Schlüsselproblem in heutigen und in zukünftigen GIS. Stehen Mehrfachrepräsentationen und vor allen Dingen Methoden zur konsistenten und effizienten Modellierung, Speicherung und Abfrage dieser bereit, so erwachsen hieraus Vorteile gegenüber monorepräsentativen Datenbanken, welche auf verschiedenen Gebieten genutzt werden können. Bei der Datenhaltung können durch die Verknüpfung aller Datensätze effizientere Aktualisierungsprozesse sowie eine Maximierung der Datenbank-Integrität erreicht werden. In einem GIS bedeuten alternative Repräsentationen neue Analysemöglichkeiten. Und nicht zuletzt können hierdurch optimale Visualisierungen der Daten ohne aufwändige Generalisierungsprozesse realisiert werden. In dem folgenden Abschnitt sollen einige Anwendungsbeispiele aufgezeigt werden, in denen eine multirepräsentative Datenbank in das System integriert wurde. Die Ursachen zur Einführung einer MRDB und die daraus resultierenden Anwendungsbereiche sind dabei eng miteinander verknüpft.

3.3.1 Darstellung von Geodaten in unterschiedlichen Maßstäben

Multiple Repräsentationen sind ein natürliches Phänomen bedingt durch unterschiedliche Anforderungen und Sichtweisen der Datenanbieter und -nutzer. In der Kartografie werden auf Grund unterschiedlicher Anforderungen an Maßstab und Auflösung in den kartografischen Produkten verschiedene Repräsentationen der darzustellenden Objekte benötigt. Daher bestand auch die ursprüngliche Anforderung darin, diese maßstabsabhängigen Strukturen, Unterschiede sowie Übergänge zu modellieren und zu verstehen.

Um die Konsistenz zu erhalten und gleichzeitig Redundanz zu vermeiden, existiert im Idealfall in einer kartografischen Datenbank nur ein hochauflösender Datensatz, aus dem alle notwendigen Repräsentationen automatisch abgeleitet werden können. Somit ist die Konsistenz der Daten automatisch sichergestellt (Buttenfield & Delotto 1989). Der Vorteil liegt in diesem Fall in einer redundanzfreien Datenbasis mit geringem Speichervolumen. Auf dem Weg dahin müssen jedoch zunächst die Eigenschaften und Zusammenhänge der unterschiedlichen Repräsentationen verstanden werden. Auch wenn zahlreiche Forschungsbemühungen zur Echtzeit-Generalisierung existieren (vgl. van Oosterom & Schenkelaars 1995, Jones et al. 2000, Lehto & Kilpeläinen 2001, Cecconi 2003), so ist dieses Problem nach Harrie et al. (2002) und anderen weit davon entfernt, vollständig gelöst zu sein. Wird ein Algorithmus eingesetzt, um eine Generalisierung durchzuführen, so ist in den meisten Fällen eine Nachbearbeitung nötig, um alle Sonderfälle zu behandeln. Wenn jedoch ein bestimmtes Produkt zu unterschiedlichen Zeitpunkten mehrmals abgefragt wird, so bedeutet dies eine Wiederholung dieser Arbeiten (Dobson & McAvoy 1989). Alternativ hierzu besteht die Möglichkeit der kontrollierten Redundanz (Buttenfield & Delotto 1989). Durch das Speichern der Bearbeitungsergebnisse wird Arbeitsaufwand gespart, gleichzeitig ist es dem Kartenautor oder -anwender möglich, immer auf die optimale Repräsentation zurückgreifen zu können. Daher ist es, um die Bedürfnisse der Nutzer befriedigen zu können, sinnvoll, die Ergebnisse oder aber die Schritte, die zu diesem Ergebnis führten, in einer Datenbank zu speichern, um bei Bedarf auf diese zugreifen zu können (Kilpeläinen 1997). Diese Datensätze in eine MRDB Struktur zu integrieren, umfasst folglich den Vorteil einer konsistenten Datenhaltung und kann eine zu einem bestimmten Zeitpunkt notwendige Generalisierung ersetzen oder den zu leistenden Aufwand minimieren.

3.3.2 Webmapping

Die schnelle Entwicklung des Internets mit der Möglichkeit, von nahezu jedem Standpunkt auf entfernte Computer zugreifen zu können, hat auch die Entwicklung der modernen Kartografie beeinflusst. Der Kunde kann bei Bedarf direkt auf die Datenbanken oder ein davor geschaltetes Portal zugreifen. Gleichzeitig wurden Strukturen und Standards zur Verbreitung von Geodaten über das Internet, wie WFS oder WMS (OGC 2005c, 2006) geschaffen. Detaillierte Aussagen über diese Technologien sind in dem Abschnitt 2.3 zu finden.

Zwei Begriffe charakterisieren diese neuen Möglichkeiten, Karten bereitzustellen: Die Bezeichnung „on-demand Mapping“, also die Kartierung nach Bedarf, beschreibt die Möglichkeit, eine kartografische Darstellung anzufordern und dabei die individuellen Bedürfnisse das resultierende Produkt betreffend anzugeben. Zum anderen kennzeichnet der Begriff „Webmapping“ die Möglichkeit, über das Internet auf Karten zugreifen zu können.

Cecconi (2003) charakterisiert das „on-demand Webmapping“ unter anderem durch folgende Eigenschaften:

- Ein temporärer Datensatz wird zum Zweck der Visualisierung in einem reduzierten Maßstab generiert
- Die Präsentation der Daten muss den Präferenzen des Nutzers sowie den Anforderungen der Bildschirme Genüge leisten

- Die Darstellung wird automatisch erstellt, ohne die Möglichkeit der Kontrolle
- Die resultierende Karte muss innerhalb einer bestimmten Zeitspanne, abhängig von den Anforderungen und der aktuellen Situation des Nutzers, erscheinen (Feringa 2001)

Diese Charakteristika treffen auch auf den im Verlauf dieser Arbeit entwickelten Webdienst zur Bereitstellung von Karten unterschiedlicher Maßstäbe zu (vgl. Kapitel 4). Entscheidend an dieser Stelle ist die Aussage, dass die Daten entsprechend den Bedürfnissen des Nutzers aufbereitet und innerhalb kürzester Zeit präsentiert werden sollten. Hierdurch entsteht die Notwendigkeit, diese durch online-Generalisierung (in vielen Fällen auch on-the-fly Generalisierung genannt), also der Generalisierung der digitalen Daten in dem Moment des Bedarfs, einem bestimmten Präsentationsmaßstab anzupassen. Da jedoch derzeit keine ausgereifte Lösung zur Automatisierung der Generalisierung existiert, ungeachtet der langen Antwortzeiten, bedingt durch zeit- und rechenintensive Algorithmen (Jones et al. 2000), so bietet sich an dieser Stelle die Verwendung vorbereiteter Datensätze einer MRDB an. Sie ist ein wichtiges Element für on-demand Webmapping-Anwendungen, in denen der Nutzer eine Darstellung erhält, welche auf seine individuellen Bedürfnisse zugeschnitten ist (Bernier et al. 2005). Multiskalige Datenbanken unterstützen oder ersetzen die Notwendigkeit der Online-Generalisierung, respektive den Bedarf, die Geodaten innerhalb weniger Sekunden für einen bestimmten Maßstab aufzubereiten.

Cecconi (2003) behandelt in seiner Arbeit zu dem Thema der kartografischen Generalisierung im Rahmen des Webmappings die Möglichkeit, mit Hilfe einer MRDB neue Datensätze des gewünschten Maßstabs zu generieren. Hier wird davon ausgegangen, dass die MRDB nicht den gesamten möglichen Maßstabsbereich abdecken kann. Das System übermittelt die dem gewünschten Maßstab am besten entsprechenden Datensätze aus der MRDB an die sogenannte „Map-Making-Unit“. Diese Einheit übernimmt im einfachsten Fall den LoD⁵ aus der MRDB, welcher den Anforderungen am ehesten entspricht und verkleinert oder vergrößert diesen auf den gewünschten Maßstab, ohne eine Generalisierung vorzunehmen. Ab einem bestimmten Grad des Vergrößerns oder Verkleinerns reicht dieses Verfahren allein nicht mehr aus und die Daten müssen zusätzlich generalisiert werden. Um die erforderlichen Generalisierungsschritte durchzuführen, werden die beiden benachbarten LoD der MRDB, also derjenige mit der nächst kleineren und derjenige mit der nächst höheren Auflösung zu Hilfe genommen. Objekte, die in beiden Maßstäben vorkommen, werden übernommen und in einen durchschnittlichen Zustand überführt (Morphing). Objekte, die nur im Datensatz mit dem höheren LoD auftreten, werden abhängig vom Objekttyp generalisiert. Dieser Ansatz greift folglich auf multiskalige Datensätze zurück und kombiniert diese mit Techniken der Online-Generalisierung, um einen durch die vorhandenen Datensätze nicht abgedeckten mittleren Maßstabsbereich zu versorgen.

Jones et al. (2000) beschreiben ebenfalls ein System, welches Anwendungen unterstützt, die online Karten bereitstellen. Dieses System besteht aus zwei Hauptkomponenten, der multiskaligen Datenbank sowie der Komponente zur Online-Darstellung. Die letztgenannte ist dabei unterteilt in die Nutzerschnittstelle und das Modul zur Lösung von Darstellungskonflikten. Die multiskalige Datenbank speichert räumliche Objekte unterschiedlicher Auflösung. Das System ruft die geometrischen Repräsentationen aus der MRDB ab, welche der Anfrage des Nutzers am nächsten kommen. Diese Objekte werden anschließend entsprechend der vom Nutzer vorgegebenen Symbolisierung gerendert. Auf diese Art und Weise kann es zu Konflikten zwischen benachbarten Objekten kommen. Die individuell kombinierten Objekte und unterschiedlichen Symbolisierungen führen zur Unterschreitung der Minimaldimensionen oder Überlappungen im Kartenbild. Die Lesbarkeit der resultierenden Karte wird gemäß den Minimaldimensionen durch ein interaktives Modul zur Lösung der Darstellungskonflikte sichergestellt.

Beide genannte Arbeiten konzentrieren sich auf die Abfrage von Kartendaten aus einem proprietären System heraus.

3.3.3 Automatische Aktualisierung von Geodaten

Eine wichtige Bedingung in einer MRDB besteht in der Konsistenz der unterschiedlichen Repräsentationen. Diese Konsistenz wird gestört, wenn Veränderungen an einer Repräsentation vorgenommen werden und muss durch Regeln und Zwänge in der Datenbank sichergestellt werden. Hier liegt auf der anderen Seite jedoch gleichzeitig ein enormes ökonomisches Potenzial. Durch geeignete Vorschriften besteht nun die Möglichkeit, die Modifikationen in einem Datensatz automatisch auf alle verknüpften Datensätze zu übertragen. Mussten früher alle vorhandenen Datenbankinhalte in regelmäßigen Abständen aktualisiert werden, so genügt nun, einzig den am höchsten aufgelösten Datensatz nachzuführen und die Änderungen kaskadierend auf die Repräsentationen niedrigerer Auflösung zu übertragen. Ohne die Einführung von Konsistenzregeln und der Automation von

⁵Level of detail

Aktualisierungsvorgängen wird eine Datenbank, welche eine dynamische Realität repräsentiert, schnell obsolet (Balley et al. 2004). Zum heutigen Zeitpunkt werden die bei den Landesvermessungsämtern vorgehaltenen Datensätze unterschiedlicher Auflösung in regelmäßigen Abständen aktualisiert. Da digitale Daten jedoch immer häufiger direkt aus der Datenbank abgerufen werden und der Anwender nicht auf die Publikation dieser durch eine Neuauflage der Kartenwerke warten muss, sind die Daten möglichst aktuell zu halten. Um diese Veränderungen der Realität möglichst effektiv zu bearbeiten, wird beispielsweise von Vauglin (2000) empfohlen, die Aktualisierungen nicht mehr maßstabsbezogen vorzunehmen, sondern vielmehr eine objektorientierte Betrachtungsweise zu wählen. Hierdurch werden redundante Arbeiten vermieden, welche auftreten, wenn für jeden Maßstab die Objekte separat erfasst und aktualisiert werden. Gleichzeitig ist es möglich, wichtige Objektarten, wie Straßen oder Gebäude, in allen Repräsentationen aktuell zu halten, während weniger wichtige Objektarten weniger häufig aktualisiert werden können.

Kilpeläinen (1995) beschreibt einen Prozess, die sogenannte inkrementelle Generalisierung, bei der der notwendige Generalisierungsprozess in Module unterteilt wird. Dabei entspricht ein Modul einer Geodatenentität, welche unabhängig von seiner Umgebung prozessiert werden kann. Werden nun im Datensatz mit der höchsten Auflösung Änderungen eingetragen, so werden in einem nächsten Schritt Module definiert, welche von der Veränderung betroffen sind. Somit werden die betroffenen Elemente der folgenden Auflösungsstufen generalisiert. Nicht betroffene Elemente bleiben unverändert.

Harrie & Hellstrom (1999) beschreiben einen Prototypen zur automatischen Übertragung der Aktualisierungen auf alle Repräsentationen einer MRDB. Der Prozess besteht in diesem Fall aus vier Schritten:

- Prüfung, welche Objekte im Zieldatensatz erstellt oder gelöscht werden müssen
- Ausführen der bei der Prüfung festgestellten notwendigen Prozesse
- Anpassung an den Zielmaßstab durch Generalisierung
- Lösen der räumlichen Konflikte

Der Prototyp umfasst eine manuell aufgesetzte MRDB mit den Maßstäben 1:10k sowie 1:50k. Zunächst wurden Veränderungen von Straßen und Gebäuden im größeren Maßstab vorgenommen und durch inkrementelle Generalisierung automatisch auf den Zielmaßstab 1:50k übertragen. Ein Vergleich mit der manuellen Fortführung zeigte eine Steigerung der Produktivität beim Einsatz des automatischen Verfahrens. Schwächen ergaben sich durch die Abhängigkeit der Ergebnisse von der Reihenfolge der Aktualisierung, einige topologische Fehler in den Straßendaten sowie die Beschreibung der Generalisierungsprozesse in formalen Regeln: Nicht alle Vorgänge der manuellen Generalisierung können durch diese Regeln ausgedrückt werden. Daher entstanden Unterschiede zwischen den automatisch und den manuell erzeugten Aktualisierungen.

Anders & Bobrich (2004) beschreiben ein in dem Projekt *WIPKA*⁶ entwickeltes MRDB-System, welches auf einer föderierten Struktur basiert. Eine solche Architektur, wie sie auch von Conrad (1997) beschrieben wird, besteht aus verschiedenen lokalen Datenbanken, welche zu einem gesamten System zusammengeschlossen werden, ohne jedoch ihre lokale Autonomie zu verlieren (siehe hierzu auch Unterabschnitt 3.4.2). In diesem Fall ist es wichtig, die Eigenständigkeit der einzelnen DBMS zu erhalten, da jedes Landesvermessungsamt seine eigene Datenbank unterhält. Das Ziel des Projektes bestand darin, eine für das ATKIS definierte MRDB aufzubauen, welche es ermöglicht, eine inkrementelle Generalisierung durchzuführen. Es wird hierdurch zukünftig möglich sein, durch manuelle Änderungen in dem DLM mit der größten Auflösung (Basis-DLM) die übrigen Modelle geringerer Auflösung (DLM50, DLM250, DLM1000) durch Methoden der Modellgeneralisierung automatisch fortzuführen. Neben der Beschleunigung des Revisionsprozesses wird durch Überprüfung von Regeln und Zwängen sowie daraus folgenden Aktivitäten die Konsistenz im gesamten Datenmodell sichergestellt.

Dunkars (2004a) nutzt Prozesse des „Data Minings“⁷, um die Disparitäten zwischen den Repräsentationen unterschiedlicher Auflösungen zu untersuchen. Für jede Objektklasse wird die Wahrscheinlichkeit für die Darstellung ihrer Entitäten in der kleineren Auflösung bestimmt. Anschließend wird, basierend auf diesen Ergebnissen, eine inkrementelle Generalisierung nach Kilpeläinen (1995) durchgeführt.

Es existieren eine Vielzahl weiterer Ansätze, in denen Aktualisierungsvorgänge in den Datenbanken automatisiert auf andere Repräsentationen übertragen werden (siehe hierzu Devogele et al. 1996, van Wijngaarden et al. 1997, Uitermark et al. 1998, Spéry 1998, Badard 1999, Badard & Lemarié 2000, Kang et al. 2004, Haurert & Sester 2005, Skogan 2005).

⁶Wissensbasierter photogrammetrisch-kartographischer Arbeitsplatz

⁷Das systematische (in der Regel automatisierte oder halbautomatische) Entdecken und Extrahieren unbekannter Informationen aus großen Mengen von Daten

3.3.4 Analyseverfahren

Werden GIS oder andere Werkzeuge zu Analysezwecken eingesetzt, so bietet eine multirepräsentative Datenbasis die Möglichkeit, alle Informationen, welche mit einem bestimmten Phänomen verknüpft sind, abfragen zu können. Dies beinhaltet zum einen den oben erwähnten Vorteil, dass immer eine adäquate Visualisierung zur Verfügung steht. Darüber hinaus bestehen jedoch auch inhaltliche Unterschiede in den Datensätzen. Die Interoperabilität ermöglicht es, komplementäre Informationen über ein bestimmtes Faktum in den verschiedenen Datensätzen zu finden und zu kombinieren, wie es zum Beispiel in dem oben aufgezeigten Anwendungsfall des Risikomanagements im Projekt *MurMur* der Fall ist.

Sester et al. (1998) beschreiben die Möglichkeit der hierarchischen Datenanalyse: Für eine Übersichtsdarstellung werden Daten mit geringer Auflösung benötigt, wohingegen in bestimmten Bereichen die Notwendigkeit besteht, mehr Details zu erhalten. Hier wird das bekannte Prinzip „vom Groben zum Feinen“ verfolgt, bei dem zunächst die weniger komplexen Daten zur Analyse der Situation herangezogen werden. Die Interpretation wird für den Nutzer erleichtert, indem exzessive Details ausgeblendet werden und die Dichte der Daten reduziert wird. Die Komplexität des Problems kann nun kaskadierend erhöht werden, wobei der Problembereich gleichzeitig immer weiter eingeschränkt wird.

In einem von Bédard & Bernier (2002) vorgestellten Konzept hat der Nutzer die Möglichkeit, bestimmte von ihm gewünschte Eigenschaften eines Objektes auszuwählen. Dabei führt die Datenbank unterschiedliche semantische, geometrische und grafische Ausprägungen desselben Objektes, deren Kombination eine bestimmte Sichtweise auf die Objekte ausmacht. Dieses Konzept der Datenanalyse ist dem aus dem Bereich der Datenbanktechniken stammenden OLAP⁸ entliehen, wo durch gezielte Abfragen an eine Datenbank bestimmte Hypothesen bestätigt oder abgelehnt werden. Auch hier erlauben die vorliegenden Daten eine hierarchische Analyse; die gewünschte Auflösung der Informationen kann variiert werden. Daraus entstammen Operationen wie „drills“, also das „Bohren“ nach Informationen (Bernier et al. 2005), welche mit einem bestimmten Phänomen verknüpft sind. Diese Suche kann auf geometrischer oder auch auf semantischer Ebene erfolgen. Das intelligente Zooming (Timpf & Devogele 1997) schließlich vergrößert eine Karte nicht nur entsprechend des Maßstabes sondern passt Inhalt und Auflösung diesem an.

Gleichzeitig gilt es jedoch zu beachten, dass unterschiedliche Repräsentationen auch unterschiedliche Ergebnisse einer Analyse hervorrufen können. Geometrische Operationen sind beispielsweise abhängig von dem LoD der Objekte. Mit wechselnder Repräsentation verändert sich zur selben Zeit die Geometrie dieser. Details werden weggelassen oder Objekte zusammengefasst. Somit ändern sich Flächeninhalte, Abstände oder Anzahl der Objekte. Auch die Semantik erfährt bei einer Veränderung der Auflösung Modifikationen. Es ist daher wichtig, diese Abhängigkeit der spezifischen Eigenschaften einer Repräsentation vom Maßstab zu kennen und zu beachten.

Des Weiteren besteht eine Abhängigkeit von der Reihenfolge der Operationen, wie sie von van Beurden & Douven (1999) aufgedeckt werden. Werden Informationen einem detaillierten Datenbestand entnommen, auf diesen Daten Operationen angewendet und deren Ergebnis anschließend aggregiert, so entsteht ein unterschiedliches Ergebnis, als wenn die gleichen Operationen mit dem aggregierten Datensatz durchgeführt werden.

Auf der anderen Seiten sind viele Phänomene auch maßstabsabhängig und werden somit erst durch die Analyse unterschiedlicher Repräsentationen aufgedeckt (Buttenfield & Delotto 1989, Sester 2000). Es lassen sich topologisches Verhalten oder andere Einflüsse der Maßstabsänderungen analysieren. Cecconi (2003) extrahiert aus verschiedenen Auflösungsstufen einer MRDB die Unterschiede zwischen den Repräsentationen, um hieraus Regeln für die Generalisierung ableiten zu können.

3.4 Mögliche Wege zum integrierten, multirepräsentativen Datenmodell

Zu Beginn dieses Abschnittes sollen die Begriffe „Datenmodell“ und „Datenbankschema“, welche oftmals synonym verwendet werden, für diese Arbeit definiert werden (entsprechend der Definition von Devogele et al. (1998)): Das „Datenbankschema“ bezeichnet die applikationsspezifische Beschreibung der in einer Datenbank gespeicherten Daten, in GIS vielfach als Datenmodell tituliert. Es legt daher folgende Eigenschaften fest (Stein 2006):

⁸Online Analytical Processing

- Statische Eigenschaften
 - Identifizierbare Objekte (Basisdatentyp)
 - Beziehungen zwischen Objekten
 - Attribute von Objekten und Beziehungen
- Dynamische Eigenschaften
 - Operationen auf Daten
 - Abfolge und Koordination von Operationen
- Integritätsbedingungen für Daten und Operationen

Das „Datenmodell“ dagegen beschreibt den Satz von Modellierungskonzepten und ist demzufolge abstrakter formuliert als das Datenschema. In GIS wird dieses oftmals als „Metamodell“ bezeichnet.

Datenbanken enthalten in der Regel *einen* konsistenten und ganzheitlichen Datensatz, welcher die jeweiligen Anforderungen bestimmter Nutzergruppen respektive den entsprechenden Applikationen erfüllt. Während die reale Welt sich hingegen mit all ihren Phänomenen konsistent und einheitlich darstellt, erzeugen die unterschiedlichen Repräsentation dieser, abhängig von der jeweiligen Anwendung, inkonsistente Datensätze. Eine MRDB vereint diese unterschiedlichen Sichtweisen in einem einheitlichen und konsistenten System. Konsistenz ist erforderlich, um Widersprüche in der Datenbank auszuschließen. Diese führen zu einem fehlerhaften Verhalten. Anfragen, zu denen der Nutzer dasselbe Resultat erwarten würde, liefern dann unterschiedliche Ergebnisse. Konsistenz erhält die topologischen, semantischen, abstands- und richtungsabhängigen Eigenschaften eines individuellen Objektes sowie die räumlichen Beziehungen zwischen diesen (Paiva 1998). Es genügt daher nicht, unterschiedliche Sichtweisen auf ein und dasselbe Phänomen zu speichern, sondern es gilt, diese miteinander zu verknüpfen und so ein widerspruchsfreies Modell aufzubauen. Werden verschiedene Datenbanksysteme innerhalb einer Organisation unabhängig voneinander vorgehalten, so ergeben sich mögliche Nachteile in Form eines hohen Potenzials an Unvollständigkeit, Ungenauigkeiten und Inkonsistenzen bei der Datenerfassung und -prozessierung sowie Mangel an Koordination in Form von Vervielfachung des Aufwandes und der Ressourcen (Parent & Spaccapietra 2000). Die Datenbankintegration sorgt dafür, dass die unterschiedlichen Datensätze in einem einheitlichen System zusammengefasst werden. Durch sie wird die Interoperabilität zwischen existierenden Datensätzen sichergestellt (Devoegele et al. 1996).

3.4.1 Schritte in Richtung eines integrierten Systems

Die Datenbankintegration bedeutet den anspruchsvollsten und gleichzeitig den leistungsfähigsten Ansatz zur Interoperabilität. Daneben existieren alternative, einfachere Wege zur Erhöhung der Interoperabilität, welche von Devoegele et al. (1998) angesprochen und quasi die ersten Schritte in Richtung eines integrierten Systems darstellen.

Eine Möglichkeit besteht darin, auf eine Integration zu verzichten und dem Nutzer lediglich einen Katalog von Metadaten an die Hand zu geben, welcher die verfügbaren Datenquellen beschreibt. Dieser enthält beispielsweise Informationen über die Datenqualität, den Maßstab, angewendete Generalisierungsschritte etc.

Ein erstes Verfahren zur Integration ist die Integration per Hand durch Implementierung von spezifischen Lösungen. Hierbei wird zum Beispiel eine Applikation in Einzelteile zerlegt, wobei jedes dieser Teile jeweils auf nur eine Datenquelle zugeschnitten ist. Die resultierenden Daten werden anschließend an eine globale Applikation weitergereicht, welche die Daten entsprechend zusammengefasst. Oder die Daten werden durch auf die Daten zugeschnittene Routinen aus den verschiedenen Datenquellen in eine neue Datenquelle kopiert und dort verfügbar gemacht.

Eine zweite Möglichkeit zur Verbesserung der Interoperabilität besteht in der Standardisierung der Datenmodellierung und Objektbearbeitung, welche den Austausch zwischen heterogenen Systemen erleichtert. Zur Datenmodellierung seien die ISO/TC211⁹ oder das OGC genannt. Die von diesen Gremien definierten Standards beschreiben die Art und Weise, wie bestimmte Realweltobjekte behandelt und für ein Computersystem nutzbar gemacht werden (Buehler 2003). Hier besteht das Ziel darin, weg von der applikations-spezifischen Datenmodellierung hin zu Datenmodellen, welche die Interoperabilität verbessern. Es werden Schnittstellen entwickelt, um verschiedene DBMS zu verbinden und den Zugriff auf die Daten eines anderen Systems zu ermöglichen.

⁹Auf internationaler Ebene (ISO) arbeitet das TC 211 (Geographic Information/Geomatics) seit 1994 an der Normung auf dem Gebiet der Geoinformationen

Diese Lösung erlaubt zwar den Zugriff auf alternative Datenquellen, unterstützt jedoch nicht die automatische Erhaltung der Konsistenzregeln zwischen den einzelnen Datenbanken.

Um zu einem integrierten System zu gelangen, müssen existierende Modelle um neue Konzepte erweitert werden, wie Verlinkungen der multiplen Repräsentationen inklusive einer wohldefinierten Semantik (wie etwa: „Repräsentation A beschreibt dasselbe Phänomen wie Repräsentation B“) sowie damit verknüpfte Bedingungen und Operatoren. Durch die Verknüpfung korrespondierender Objekte wird die Konsistenz der Daten sichergestellt. Konsistenzbedingungen entstehen jedoch nicht durch Einführung dieser Verknüpfungen, sondern müssen explizit festgelegt und dann durch diese kontrolliert werden. So ist zum Beispiel festgeschrieben, dass ein bestimmtes Phänomen in allen Repräsentationen zu finden ist oder auch, dass zwei zu verknüpfende Repräsentationen geometrisch zueinander passen müssen, so dass beispielsweise die Mittelpunkte einer Straßenkreuzung in den verschiedenen Datensätzen übereinstimmen.

Der Aufbau einer derartigen, multiskaligen Datenbank wirft nach Devogele et al. (1996) folgende Probleme auf:

- Verknüpfungen zwischen den verschiedenen Objekttypen: Phänomene der realen Welt werden in unterschiedliche Datenmodelle überführt; hier sind Methoden zur Schemaintegration erforderlich
- Beschreibung der Verknüpfungen zwischen den Objekten aus unterschiedlichen Repräsentationen: Zur Beschreibung der Verknüpfungen zwischen den korrespondierenden Objekten unterschiedlicher Maßstäbe sind besondere Datenmodelle erforderlich
- Daten-Matching: Zur Bestimmung korrespondierender Objekte werden Methoden und Algorithmen benötigt, um zwei geografische Datensätze nach Gruppen von Objekten zu durchsuchen, die denselben Teil der realen Welt wiedergeben

In ihrer Arbeit beschreiben Devogele et al. (1998) eine konzeptionelle Vorgehensweise zur Integration geografischer Daten, aufbauend auf den Konzepten von Spaccapietra et al. (1992). Dabei werden zunächst drei Probleme aufgezeigt, die auftreten, wenn verschiedene Datenquellen in dasselbe System integriert werden sollen:

- Entwicklung eines korrekten Verständnisses der Semantik der existierenden Daten
- Festlegung der Zuordnungsstruktur zwischen den Eingangsschemata
- Wahl einer entsprechenden, integrierten Beschreibung aller Daten, basierend auf den Zielen der Integration und den verfügbaren Techniken zur Transformation

Ausgehend von diesen Problemen werden folgende Schritte zur Schemaintegration festgelegt:

- Schemavorbereitung
- Bestimmung der Gemeinsamkeiten und Diskrepanzen
- Integration: In dieser Phase werden die möglichen Konflikte gelöst und eine integrierte Beschreibung in Form eines virtuellen Schemas geschaffen. Es werden Regeln zur Integration definiert.

Die einzelnen Schritte werden von Parent & Spaccapietra (2000) ausführlich dargelegt und sollen im Folgenden zusammenfassend beschrieben werden.

Vorbereitung zur Integration Zunächst müssen vorbereitende Aktivitäten stattfinden, welche auf eine Zusammenführung der aktuellen Datensätze zielen. Dies umfasst zum Beispiel die Umformung der Daten in ein einheitliches Format. Heterogene Datenbankschemata in unterschiedlichen Datenbankmodellen müssen durch ein einheitliches Schema in einem gemeinsamen Datenbankmodell beschrieben werden. Zusätzlich sollten komplementäre Informationen hinzugefügt werden auf dem Wege zu einer einheitlichen Verständnisebene, zum Beispiel durch Ergänzung einer Beschreibung des verwendeten Koordinatensystems oder die Schaffung eines Thesaurus zur einfachen Übertragung von Informationen.

Identifizierung der Datenbankkorrespondenzen Hier entsteht eine präzise Beschreibung der Korrespondenzen zwischen den Ausgangsschemata. Die Grundidee besteht darin, einen gewissen Grad an Ähnlichkeit zwischen zwei Beschreibungen zu bestimmen, basierend auf der Übereinstimmung von Namen, Strukturen und Bedingungen (Spaccapietra et al. 1992). Gleichzeitig werden an dieser Stelle Konflikte aufgedeckt, welche bei der Integration zu lösen sind. Zwei Datenbanken besitzen Gemeinsamkeiten, wenn die realen Gegebenheiten, welche diese repräsentieren, über sich überlappende Elemente verfügen. Dabei werden Korrespondenzen zwischen den Objekttypen bestimmt und nicht zwischen einzelnen Instanzen dieser (Parent 2000). Der Integrationsprozess besteht darin, diese Korrespondenzen zu bestimmen und für jede dieser Verknüpfungen den Nutzern eine globale Beschreibung inklusive den an den verknüpften Elementen hängenden Daten zur Verfügung zu stellen.

Integration Der Schritt der Integration untersucht nun die einzelnen, zuvor benannten Korrespondenzen und bestimmt, ob eine Objektinstanz in das integrierte Schema übernommen wird. Die Integration versucht, mögliche Konflikte zu lösen, welche zwangsläufig auftreten, wenn verschiedene Datensätze zusammengeführt werden sollen. Es ist unwahrscheinlich, dass verschiedene Datensätze perfekt zusammenpassen. So beschreiben zwei Datensätze zum Beispiel selten exakt dasselbe Phänomen der realen Welt, sondern es existiert jeweils nur ein gewisser gemeinsamer Überlappungsbereich, geometrisch oder semantisch. Die korrespondierenden Objekte weisen Unterschiede in der Art der Repräsentation oder im Inhalt auf. In diesem Fall tritt ein Konflikt auf, den es zu lösen gilt. So existieren beispielsweise unterschiedliche Klassifikationen, so dass der eine Datensatz einen „Fluss“ als solchen spezifiziert, wenn dieser breiter als drei Meter ist und in dem anderen Datensatz dieser mindestens fünf Meter breit sein muss und ansonsten als „Bach“ bezeichnet wird.

Eine Möglichkeit, die oben angesprochenen Klassenkonflikte zu lösen, besteht darin, die einzelnen Elemente in eine entsprechende Spezialisierungs-/Generalisierungs-Hierarchie zu integrieren, wie dies beispielsweise bei Gotthard et al. (1992) beschrieben wird. Bei Bedarf wird ein entsprechender Ober- oder Untertyp eingeführt. Alternativ wird im Fall eines einfach zu haltenden Schemas durch Verschmelzung ein einheitlicher Datentyp eingeführt, welcher beide zu integrierenden Objekte gleichzeitig beschreibt. Dieser Objekttyp beschreibt dann nur die Schnittmenge der beiden Ausgangstypen, die Unterschiede gehen verloren.

Strukturelle Konflikte treten auf, wenn korrespondierende Typen mit verschiedenen Konstrukten beschrieben werden, wenn beispielsweise in dem einen Datenbankschema eine Klasse vorhanden ist, welche in dem anderen Schema durch ein Attribut einer anderen Klasse ausgedrückt wird. In diesem Fall würde das Attribut des letztgenannten Schemas in einen Klassentyp überführt werden.

Ein Beschreibungs-Konflikt entsteht schließlich, wenn beispielsweise Namen von Objekttypen, Attributen oder Methoden durch homonyme oder synonyme Verwendung Probleme hervorrufen. Wird eine Integration auf Basis der Objektbeschreibung oder auf Grund bestimmter Attribute durchgeführt, so kann es passieren, dass zwei Straßensegmente mit demselben Namen „Hauptstraße“ in Beziehung gesetzt werden, obwohl der eine Datensatz eine Straße in „Frankfurt/Main“ und der zweite Datensatz eine andere Straße in „Frankfurt/Oder“ beschreibt. Auch hier existieren für die denkbaren Konfliktsituationen unterschiedliche Lösungsansätze (siehe zum Beispiel Kim (1995), Larson et al. (1989)). Für die genannten Fälle wird zum Beispiel empfohlen, Präfixe einzuführen oder einen Alias zu beschreiben.

Es besteht weiterhin die Möglichkeit von Datenkonflikten, hervorgerufen durch Schreibfehler, unterschiedliche Informationsquellen, verschiedene Versionen sowie verzögerte Aktualisierungsvorgänge. Daher ist es wichtig, Informationen zur Verlässlichkeit bereitzustellen, um im Zweifelsfall entscheiden zu können, welcher Datensatz nun die korrekten Informationen enthält.

3.4.2 Möglichkeiten der Modellierung

Ein Großteil der vorangegangenen Forschungsarbeiten hat sich auf das zweite von *Devogele* genannte Problem, die Beschreibung der Verknüpfungen zwischen korrespondierenden Objekte durch geeignete Datenmodelle, konzentriert. Dabei sind die meisten dieser Datenmodelle in dem Bereich der multiskaligen Datenbanken angelehnt an die hierarchische Struktur der Übergänge zwischen den einzelnen Maßstäben (vgl. Timpf & Frank 1995, Kilpeläinen 1998, Timpf 1998, Zhou & Jones 2001).

Nach Weibel & Dutton (1999) lassen sich hierarchische Relationen auf die Bereiche der räumlichen Primitive, der Objekte, Attribute sowie der räumlichen Datenstrukturen aufteilen. In dem Bereich der räumlichen Primitive, also den geometrischen Komponenten, wie Punkte, Linien oder Polygone, konzentrieren sich die Arbeiten auf die Veränderung der Objektdetails. In Baumstrukturen (van Oosterom 1989, Cromley 1991) wird die Reihenfolge festgelegt, in der Knoten erscheinen bzw. nicht mehr erscheinen. Hierarchien können des Weiteren über die Objekte selbst definiert werden. So können bestimmte Objekte andere Objekte einschließen. Eine Siedlungsfläche beinhaltet zum Beispiel mehrere Gebäude, oder mehrere Gebäude sind durch einen Gebäudeblock zusammengefasst. Bei Timpf & Frank (1995) findet sich die Möglichkeit, einen gerichteten azyklischen Grafen zur Modellierung solcher Hierarchien zu verwenden. In Ruzak Mazur & Castner (1990) und Ruas (1995) werden Techniken zur hierarchischen Gliederung und Modellierung von Gewässer- oder Straßennetzwerken vorgestellt. Attribute werden in einer Hierarchie, sei es in einer Aggregation oder Generalisierung der Objekte, klassischer Weise von einer Oberklasse an die in der Hierarchie folgenden Klassen vererbt. In der Domäne der räumlichen Daten existieren Datenstrukturen, welche räumliche Objekte in verschiedenen Auflösungsstufen speichern (Quadtree, Pyramiden etc.).

Um nun zu einem integrierten Schema zu gelangen, muss zunächst die Strategie der zukünftigen Datenhaltung bestimmt werden. Im Folgenden sollen die unterschiedlichen Strategien zur Integration bestehender oder neu zu

generierenden Datenschemata aufgezeigt werden, um von mehreren mono-repräsentativen Datensätzen zu einer MRDB zu gelangen (siehe Abbildung 3.2).

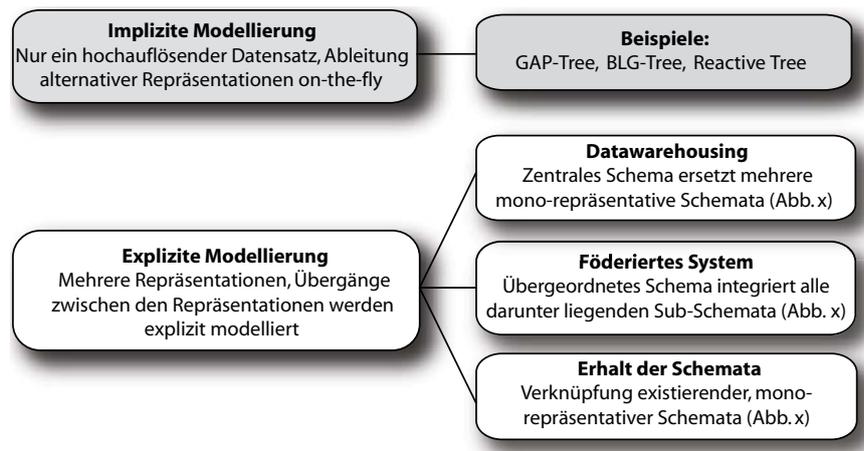


Abb. 3.2: Möglichkeiten der Modellierung einer MRDB

Implizite Modellierung

Aus Sicht der Datenhaltung und -modellierung besteht die optimale Konstellation darin, nur einen hochauflösenden Datensatz vorzuhalten, welcher gepflegt werden muss, und alle übrigen Repräsentationen anhand von bestehenden Regeln und Prozeduren, welche die Veränderungen der Daten beschreiben, on-the-fly aus diesem Datensatz abzuleiten. Allerdings ist es zur Zeit nicht möglich, allein mit Hilfe von Generalisierungsregeln unterschiedliche Auflösungsstufen auf Knopfdruck zu erzeugen.

Eine Möglichkeit, mehrere Repräsentationen eines Datensatzes implizit vorzuhalten, besteht nach Weibel & Dutton (1999) darin, die einzelnen Schritte zur Generalisierung im voraus zu berechnen und vorzuhalten. Eine neue Repräsentation kann dann, basierend auf dem einen, bestehenden Datensatz, sowie den Ergebnissen der Vorverarbeitung on-the-fly abgeleitet werden. Somit werden auf diese Weise implizit multiple Repräsentationen vorgehalten ohne explizit eine weitere Repräsentation generieren und speichern zu müssen.

Von van Oosterom & Schenkelaars (1995) werden sogenannte „reactive datastructures“ beschrieben, geometrische Datenstrukturen mit verschiedenen Auflösungsstufen, welche auf eine Interaktion des Nutzers, wie beispielsweise ein- oder auszoomen, unverzüglich reagieren können (van den Bos et al. 1984). Beispiele dieser Strukturen sind der *BLG¹⁰-Tree* (van Oosterom 1991), der *Reactive-Tree* (van Oosterom 1989) oder der *Gap-Tree* (van Oosterom 1995).

Der BLG-Tree unterstützt die Linienvereinfachung eines Generalisierungsprozesses, basierend auf dem Douglas-Peucker-Algorithmus (Douglas & Peucker 1973) (siehe Abb. 3.3). Dementsprechend wird für jede Linie $P_A - P_E$ der Baum aufgebaut, indem das Wurzelement durch den von der Linie $P_A - P_E$ am weitesten entfernt liegenden Stützpunkt definiert ist, respektive dem Punkt, welcher bei der Linienvereinfachung zuletzt eliminiert wird. So werden bei größer werdendem Maßstab schrittweise zusätzliche Stützpunkte eingefügt bzw. bei kleiner werdendem Maßstab Stützpunkte eliminiert, ohne dass aufwändige Berechnungen notwendig sind.

Der *Reactive-Tree* (van Oosterom 1991) (Abbildung 3.4) basiert auf dem R-Tree von Guttman (1984) und ordnet die Objekte im Baum zusätzlich entsprechend ihrer Wichtigkeit. Das bedeutet, dass wichtige Objekte nicht im Blattknoten abgelegt werden, sondern gemäß ihrer Bedeutsamkeit in einer höheren Ebene. Demzufolge können, ausgehend von einem hochaufgelösten Datensatz, bei kleiner werdendem Maßstab die Objekte je nach ihrer Relevanz eliminiert werden.

Der GAP-Tree wird genutzt, um die Lücken zu schließen, welche durch ausgeblendete Objekte entstehen. Dieser enthält vorverarbeitete, alternative Topologien für die ausgelassenen Polygone.

Der PR-File¹¹ (Becker et al. 1991) basiert auf dem R-File, ergänzt diesen jedoch um die Priorität eines Objektes. Das System sortiert die Objekte also räumlich und gleichzeitig nach ihrer Priorität. Dabei wird das Objekt nicht

¹⁰Binary line generalisation tree

¹¹Priority Rectangle File

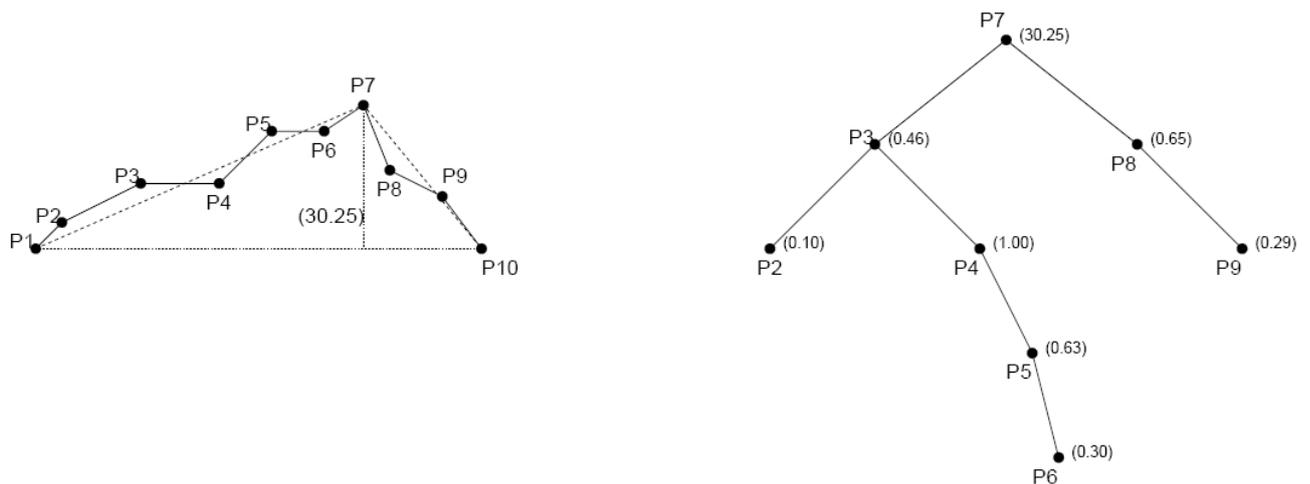


Abb. 3.3: BLG Baum: Links die zu generalisierende Linie, rechts der resultierende Baum (van Oosterom & Schenkelaars 1995)

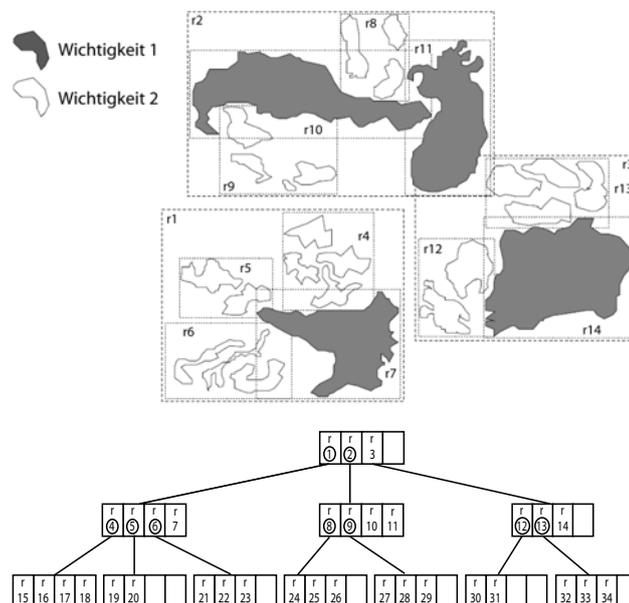


Abb. 3.4: Reactive Tree: Rechtecke (oben) und resultierender Baum (unten) (van Oosterom & Schenkelaars 1995)

als kleinste Einheit betrachtet, wie beim Reactive-Tree, sondern es werden die Endpunkte der Liniensegmente betrachtet und priorisiert, was einer Linienvereinfachung gleichkommt, ähnlich dem BLG-Tree.

Ähnliche Ansätze finden sich bei Sester & Brenner (2004). Auch hier werden die Veränderungen zwischen den unterschiedlichen Auflösungen vorgehalten. Bei größer werdendem Maßstab werden Knoten und Kanten zu den Geometrien hinzugefügt und bei Bedarf verschoben, bei einer Verkleinerung werden entsprechend die Details entfernt. Die Veränderungsvorschriften sind hierbei in einem File gespeichert. In diesem Fall wird das Verfahren zur kontinuierlichen Generalisierung eingesetzt.

Alle diese genannten Verfahren ermöglichen eine schnelle Generierung einer bestimmten Auflösungsstufe aus einem bestehenden, hochauflösenden Datensatz heraus, sind jedoch auf wenige, spezifische Generalisierungsfälle ausgelegt.

Explizite Modellierung

Existieren dagegen explizit unterschiedliche, alternative Repräsentationen desselben Phänomens, so gilt es, diese in einem integrierten System zusammenzufassen. Es sind unterschiedliche Wege denkbar, ein integriertes System zu modellieren.

Data Warehousing Eine Möglichkeit hierzu, das sogenannte *data warehousing*, besteht darin, die existierenden durch eine zentralisierte Datenbank zu ersetzen (Elmagarmid et al. 1999). Das Data-Warehouse oder Datenlager stellt also eine zentrale Datensammlung dar (siehe Abbildung 3.5). Jones et al. (1996) stellen ein System vor,

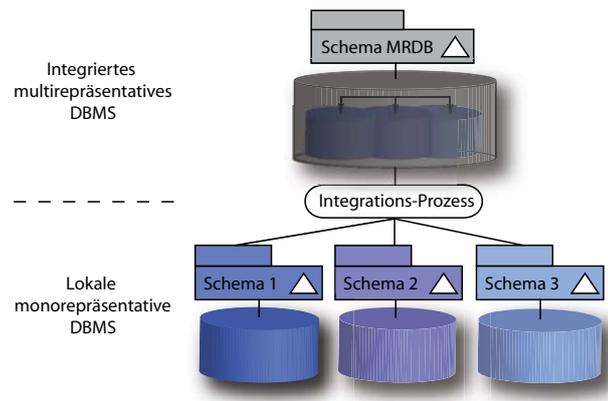


Abb. 3.5: Modellierung einer MRDB im Datawarehouse durch Integration in ein neues Datenmodell

welches geografische Datensätze in einem System unterbringt, das es erlaubt, zu jedem Objekt mehrere Geometrien und Attribute zu unterhalten. Alle existierenden Daten werden in eine einzelne Datenbank integriert und daher in einem globalen, konsistenten System unterhalten und gepflegt. Der Datenbankarchitekt hat die Möglichkeit, alle Repräsentationen durch einen Objekttypen auszudrücken. Aus der Modellierung eines Phänomens der realen Welt resultiert eine einzelne Instanz eines Objekttypen, welche verschiedene Repräsentationen beinhalten kann. Es wird versucht, alle korrespondierenden Objekttypen im integrierten Schema zu vereinigen. In diesem Fall können diese in einem multirepräsentativen Objekttypen zusammengefasst werden (siehe Abbildung 3.8 Option 1). Dies ist jedoch nur praktikabel für 1-1 Beziehungen (näheres zu den möglichen Kardinalitäten und den daraus resultierenden Datenschemata folgt im Unterabschnitt 3.4.3). Besteht dagegen eine 1-n oder n-m Beziehung zwischen den korrelierenden Instanzen, so bedeutet dies, dass die jeweiligen Repräsentationen verschiedene Teile des Phänomens beschreiben. Daher ist es wenig sinnvoll, diese Objekttypen in einem einzigen vereinigen zu wollen. Somit wird in einem solchen Fall eine alternative Lösung benötigt. Beispielsweise wird ein neues multirepräsentatives Objekt geschaffen, welches mit den mono-repräsentativen Objekttypen verknüpft wird (siehe Abbildung 3.8 Option 2). Steht dabei die Einfachheit und Lesbarkeit im Vordergrund, dann empfiehlt es sich, eine minimale Anzahl von Klassen im integrierten Schema vorzuweisen. Ist dagegen die Vollständigkeit oberstes Ziel, so wird jedes Element des Ausgangsschemas auch im integrierten Schema zu finden sein, die Ausgangsdaten sind im Zielsystem erschöpfend beschrieben. Selbst die im Ausgangsschema nicht vorhandenen Informationen werden komplementierend aufgeführt, um zukünftige Integrationen zu berücksichtigen (Parent 2000).

Föderiertes System Alternativ hierzu können die originären Datenbanken in ihrer Struktur erhalten bleiben und in ein übergeordnetes Schema überführt werden, welches als globales Schema über die lokalen Schemata gestülpt wird (siehe Abbildung 3.6). Hieraus resultiert ein sogenanntes föderiertes System. Auf der einen Seite entsteht dabei ein integriertes System, auf der anderen Seite bleibt die Autonomie der originären Daten bestehen. Die Untermenge der Daten, welche durch den Systemadministrator freigegeben ist, wird in eine virtuelle Datenbank integriert (virtuelle Informationsintegration). Virtuell bedeutet in diesem Fall, dass nur das Schema der föderierten Datenbank existiert, nicht jedoch deren Instanzen (Devegele et al. 1998, Parent 2000). Hier besteht daher eine bestimmte Sichtweise (view) auf die angeschlossenen Datenbanken. Monorepräsentative Datentypen werden dabei als Unterklassen eines multirepräsentativen Datentyps in eine Generalisierungshierarchie in datenbanktechnischem Sinne eingegliedert, wobei die Oberklasse die Gemeinsamkeiten der Unterklassen zusammenfasst. In dem EU-Projekt *GiMoDig* wurden auf diese Weise beispielsweise die großmaßstäbigen Daten europäischer Landesvermessungen integriert, indem ein globales Schema definiert wurde, welches alle Subschemata der einzelnen Länder zusammenfasst. Auf diese Weise hat der Nutzer einen einheitlichen Zugriff auf alle angeschlossenen Daten.

Verknüpfung monorepräsentativer Schemata Ebenso kann durch Hinzufügen von Verknüpfungen zwischen den bestehenden Schemata ein multirepräsentatives System geschaffen werden (Kilpeläinen 1998) (siehe Abbildung 3.7). Die Relationen zwischen den einzelnen Repräsentationen beschreiben gleichzeitig die Maßstabsübergänge zwischen diesen (Devegele et al. 1996). Diese Verknüpfungen modellieren die Sequenzen von Operationen,

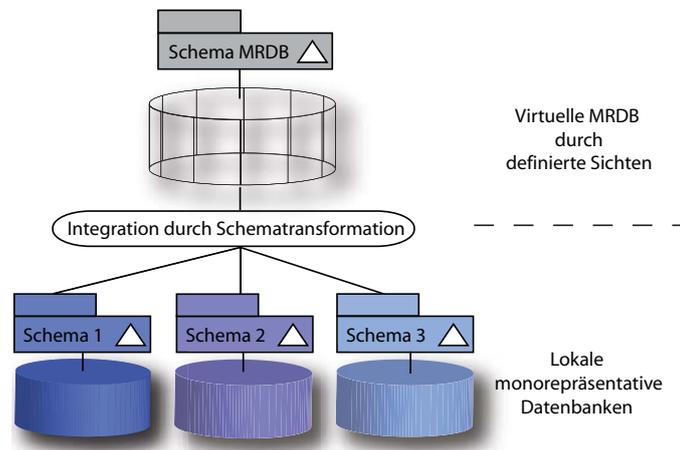


Abb. 3.6: Modellierung der MRDB in einem föderierten DBMS (virtuelle Informationsintegration). Die autonomen Datenquellen bleiben unverändert; durch Schematransformation entsteht eine virtuelle MRDB

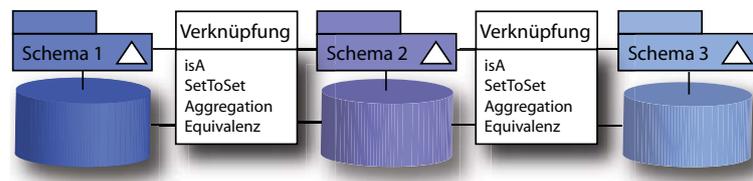


Abb. 3.7: Modellierung einer MRDB durch Hinzufügen von Verknüpfungen zwischen den monorepräsentativen Schemata

um von einer Darstellung zu einer anderen zu gelangen. In diesen Fällen kann das Schema alle originären Objekttypen der Ausgangsdatsätze übernehmen. Die ursprünglichen, mono-repräsentativen Objekttypen bleiben erhalten und werden miteinander verknüpft, beispielsweise durch *IsA*, *Equivalenz*, *Aggregation* oder *Set-to-Set* Beziehungen (Spaccapietra et al. 2000). Hierdurch entstehen zwei Instanzen zweier Objekttypen welche durch die Instanz einer Verknüpfung verlinkt werden (siehe Abbildung 3.8, Option 3). In *MurMur* wurden beispielsweise auf Grund der Diversität der zu integrierenden Datensätze diese als mono-repräsentative Objekttypen integriert und durch Relationen verknüpft. Dabei wird von Balley et al. (2004) ausdrücklich darauf hingewiesen, dass hierdurch der Konsistenzprüfung und Durchführung von Aktualisierungen genüge getan wurde und diese nicht unbedingt ein weiteres integriertes Schemas erfordern. Die Relationen zwischen den einzelnen Repräsentationen werden zusätzlich mit Integritätsbedingungen belegt, welche nur bestimmte Instanzen, beispielsweise mit bestimmten Attributwerten, an dieser Stelle integrieren. So werden zum Beispiel nur Straßenabschnitte mit demselben Straßennamen verknüpft. Gleichzeitig wird hier die Geometrie beider Objekte verglichen und es darf kein signifikanter Unterschied auftreten. Geringe Abweichungen entstehen teilweise als Folge von Generalisierungsvorgängen. Hierdurch wird auch sichergestellt, dass die an sich konsistenten monorepräsentativen Datensätze auch im integrierten System konsistent bleiben. Dabei lassen sich Inkonsistenzen jedoch nicht vollständig ausschließen. So verbleiben topologische Konflikte und semantische Fehler, da diese nicht einfach durch regelhafte Bedingungen extrahiert werden können.

Die Wahl des Ansatzes ist abhängig von der jeweiligen Ausgangssituation, den Daten und den zwischen diesen bestehenden Relationen sowie den zu erfüllenden Zielen und ist letztendlich dem Datenbankarchitekten selbst überlassen. Sollen also die vorhandenen Daten erhalten bleiben oder sollen diese in ein neues System überführt werden? Werden zum Beispiel autark gepflegte Datensätze verschiedener Anbieter, beispielsweise der Landesvermessungen, in ein einheitliches System integriert, so ist es wenig sinnvoll, diese Daten in ein neu definiertes Schema zu überführen, da die Daten nach wie vor bei den einzelnen Landesvermessungen vorgehalten werden. Somit ist hier beispielsweise ein föderiertes System zweckmäßig.

Die gewonnenen Informationen aus den drei Verfahren sind identisch, Unterschiede entstehen lediglich in der Art der Abfrage. Daher sollte zuvor auch herausgefunden werden, welche Abfragen am häufigsten auftreten und dementsprechend das richtige Schema gewählt werden, um diese möglichst effizient verarbeiten zu können. Werden bei einem Zugriff zum Beispiel häufig alle Repräsentationen eines Phänomens gleichzeitig benötigt, so ist es sinnvoll, alle Informationen in einem Datentyp zu integrieren. Soll dagegen jeweils gezielt auf eine bestimmte Repräsentation zugegriffen werden, so ist zu empfehlen, die monorepräsentativen Objekte separat vorzuhalten und diese lediglich durch Verknüpfungen in Beziehung zu setzen.

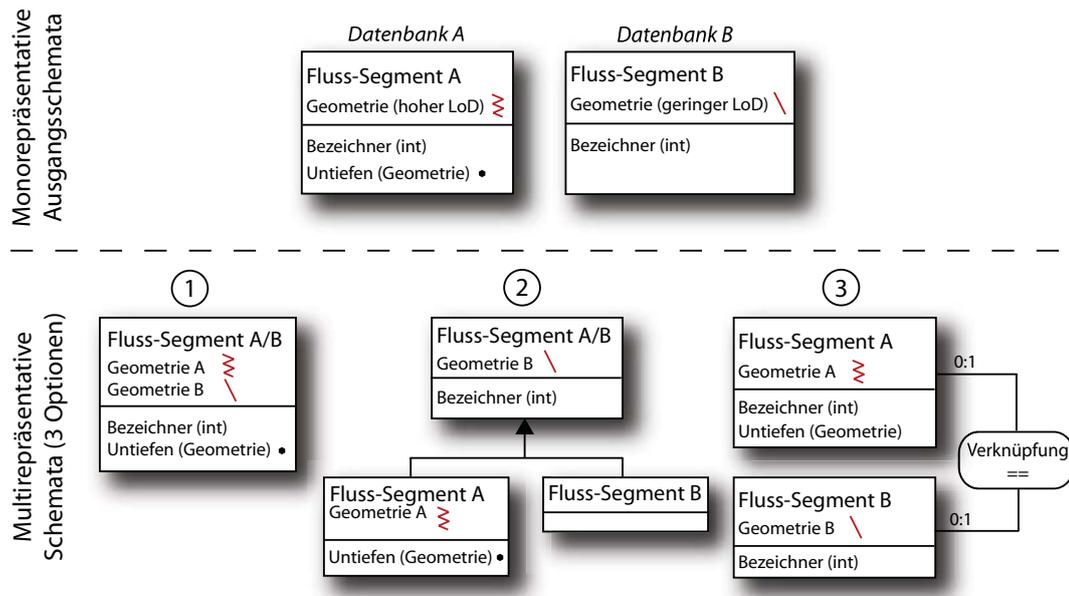


Abb. 3.8: Optionen zur Integration zweier monorepräsentativer Datensätze in ein multirepräsentatives Schema (aus Balley et al. (2004)), Option 1: Integration in einen Datentypen, Option 2: Einführung einer Oberklasse und Eingliederung in eine Generalisierungshierarchie, Option 3: Beide Klassen werden in das neue Modell übernommen und über einen Verknüpfungstyp verbunden

3.4.3 Grundsätzliche Modellierung verschiedener Verknüpfungskardinalitäten

Zur Modellierung eines integrierten System gehört unter anderem die Beschreibung der Zusammenhänge zwischen den Objektarten der verschiedenen Datenschemata. Diese Verknüpfungen bringen zum Ausdruck, dass die Objekte aus dem Datensatz *A* dasselbe Objekt modellieren wie bestimmte Objekte aus dem Datensatz *B*. So besteht zum Beispiel der Objekttyp „Kreuzung“ im Datensatz *A* aus verschiedenen Segmenten des Objekttyps „Straße“ im Datensatz *B*. Dabei gilt es zunächst, die beiden Schemata zu verknüpfen (Schemaintegration), um dann später hierauf aufbauend Instanzen dieser Schemata miteinander in Beziehung setzen zu können (siehe hierzu Unterabschnitt 3.4.5).

Die Charakteristik der Korrespondenzen hat dabei Einfluss auf die Wahl des zukünftigen, neu zu erstellenden Datenbankschemas. Abbildung 3.8 zeigt die möglichen Datenbankmodelle zur Integration zweier Objekttypen. Es ist daher sinnvoll, zunächst exemplarisch eine Verknüpfungstabelle anzulegen für jedes Paar korrespondierender Objekttypen im Ausgangsdatsatz. Aus dieser Kenntnis heraus, welche Kardinalität für welche Korrespondenz charakteristisch ist, kann der Datenbankarchitekt dann eine entsprechende Datenstruktur für das integrierte Schema bestimmen:

1-1 Beziehungen können in einen Objekttyp integriert werden, welcher alle Repräsentationen integriert oder zwei Objekttypen werden durch eine „IsA“ oder „Äquivalenz“ Verknüpfung verbunden (vgl. Abbildung 3.8, Optionen 1 und 3). *1-n Beziehungen* können durch zwei monorepräsentative Objekttypen beschrieben werden, welche durch eine „Aggregation“ miteinander in Beziehung gesetzt werden (vgl. Abbildung 3.8, Option 3). *n-m Beziehungen* sind dagegen schwieriger zu gestalten, denn das Phänomen, bestehend aus *n* bzw. *m* Objekten der jeweiligen Datensätze, ist in keinem dieser explizit beschrieben. Es wird geprüft, ob dieses Phänomen in einem anderen zu integrierenden Datensatz definiert ist, welcher als Vermittler dienen kann. Somit können die *n-m* Beziehungen in *1-n* Beziehungen aufgeteilt werden. Ist kein vermittelnder Objekttyp vorhanden und ist es gleichzeitig wichtig, das Phänomen explizit zu beschreiben, so muss ein neuer Objekttyp eingeführt werden, welcher über Aggregations-Beziehungen mit den zu verknüpfenden Datensätzen in Beziehung steht. Steht die explizite Beschreibung dagegen nicht im Vordergrund, so genügt es, die beiden mono-repräsentativen Objekttypen durch eine „Set-to-Set“ Beziehung zu verbinden.

3.4.4 Beispiele für Integrationsmodelle

Nach den hier aufgeführten, theoretischen Lösungsansätzen sollen im Folgenden einige spezifische Beispiele illustrieren, auf welche Art und Weise eine Datenintegration praktisch realisiert wurde.

Die Schwierigkeiten zur Integration liegen bei allen beschriebenen Verfahren vor allen Dingen in der Heterogenität der Datenmodelle, den konzeptionellen Schemata sowie der Semantik. Kavouras & Kokla (2002) beschreiben eine Möglichkeit zur Integration semantisch heterogener Datensätze in einen multirepräsentativen Kontext. Hierbei wird in einem ersten Schritt das Originalschema in fundamentale Kategorien zerlegt. In einem zweiten Schritt werden diese elementaren Fragmente nach aufwändiger Konzeptanalyse zu einem neuen Gerüst zusammengesetzt. Ein gemeinsames, übergeordnetes Schema ist erforderlich, um entweder die verschiedenen Schemata in dieses zu überführen oder aber mindestens eine Verknüpfung zwischen den beiden Schemata zu ermöglichen (entspr. Option 1 bzw. 2 in Abbildung 3.8).

Devogele et al. (1996) beschreiben den Aufbau einer multiskaligen Datenbank, die versucht, zwei Straßendatensätze zu integrieren. Da die beiden Datensätze unterschiedliche Definitionen der Objektklassen aufweisen, ist in einem ersten Schritt die Definition eines Schemas notwendig, welches es erlaubt, die beiden individuellen Schemata und Objektklassen zu verbinden. Das Ziel besteht hier in einem Schema, welches in der Lage sein sollte, ohne Informationsverlust alle Instanzen des Ausgangs-Schemas zu repräsentieren. Somit wird ein multirepräsentatives Schema definiert, welches sowohl die ursprünglichen Datentypen als auch übergeordnete Objekte, welche beide Repräsentationen integrieren, beinhaltet. Somit steht als Ergebnis hier ein multirepräsentatives Schema bereit, welches die Eingangstypen sowie die Übergänge zwischen den Instanzen beschreibt. Dieses Schema kann innerhalb eines föderierten Systems eingesetzt werden oder aber als Schema für eine neu definierte, multirepräsentative Datenbank. In einem zweiten Schritt werden die individuellen Objekte dieser Datensätze miteinander verknüpft durch Vergleich semantischer, topologischer sowie metrischer Informationen.

Jones et al. (1996) beschreiben das konzeptionelle Design einer multiskaligen Datenbank, genannt *GEODYSSSEY*. In dieser Datenbank sollen Instanzen von Phänomenen der realen Welt repräsentiert werden, Informationen ihrer geometrischen Form und Verteilung, räumliche und nicht-räumliche Beziehungen mit anderen Phänomenen, Klassifikationen, nicht-räumliche Attribute sowie Metadaten die Quelle und Qualität der ursprünglichen Daten betreffend. Das Datenbankdesign verfolgt dabei die Ziele einer maximalen Datenbankintegrität, die Minimierung der Interaktionen während der Aktualisierungsoperationen sowie die Automation der Abfrage räumlicher Informationen verknüpft mit einem bestimmten Phänomen, alles realisiert durch multiple Repräsentationen. Datenbankintegrität soll dabei erreicht werden durch die Trennung von Informationen zur Identifikation und Klassifizierung der räumlichen Phänomene von deren multiplen geometrischen Repräsentationen, da diese fehleranfällig sind. Geometrische Beschreibungen von Objekten sind entweder schon durch ihre Erfassung fehlerhaft oder auf Grund von Generalisierungsvorgängen falsch dargestellt. So verschafft beispielsweise eine bestimmte geometrische Repräsentation den Eindruck, dass eine Siedlung (repräsentiert durch einen Punkt) das Ende eines Straßensegmentes berührt. Diese Repräsentation genügt bestimmten Anforderungen, wie beispielsweise einem Routingsservice, gibt aber nicht die Situation wieder, wie sie sich in der Realität darstellt. So liegt die (nicht punktförmige) Siedlung in der Realität nicht direkt an der Straße, sondern ist eventuell einen Kilometer von dieser entfernt und über eine Zufahrtsstraße mit dieser verbunden.

Die Automation der Aktualisierungen soll durch Automation vormals manueller Prozeduren zur Bestimmung semantischer und geometrischer Äquivalenz zwischen neuen und existierenden Daten, durch Prozeduren zur Erhaltung der topologischen Konsistenz sowie durch spezifische Regeln zur Aktualisierung erreicht werden. Das Datenbankschema selbst wird unterteilt in einen intensionalen und einen extensionalen Teil. Der intensionale Teil enthält regelbasiertes und prozedurales Wissen, um Veränderungen oder Anfragen zu prozessieren. Der extensionale Teil dagegen enthält die multirepräsentativen semantischen, räumlichen und temporalen Daten, Metadaten sowie ein Verzeichnis der Realweltobjekte (den Phänomenen der realen Welt). Das Verzeichnis speichert die Klassifizierung, den Namen, nicht räumliche Attribute sowie Referenzen zu den einzelnen Repräsentationen. Die Phänomene werden also getrennt von ihren Instanzen gespeichert. Diese Realweltobjekte sind über den Namen mit der semantischen, geometrischen und temporalen Daten verknüpft.

Spaccapietra (2000) beschreibt das in *MurMur* entwickelte Datenmodell, welches auf einem existierenden Modell (MADS¹²) aufsetzt und dieses um die Fähigkeit der multiplen Repräsentation erweitert. Das verwendete Datenbankmodell wurde im vorangegangenen Abschnitt beschrieben. Zusätzlich besteht hier die Möglichkeit, jedes Datenbankelement mit einer Art Stempel zu versehen, welcher diesem Gültigkeit für ein bestimmtes Repräsentationsschema verleiht. Werden beispielsweise die Kriterien „Sicht“ und „Auflösung“ verwendet, so verkörpert eine bestimmte Repräsentation entweder eine bestimmte Sicht oder eine bestimmte Auflösung. Dabei kann jedes Datenbankobjekt mehrere Stempel beinhalten. Gleichzeitig gibt der Nutzer die Stempel beziehungsweise Schemata bekannt, mit denen er arbeiten möchte. Auf diese Weise ist es möglich, Repräsentationen auszuwählen, welche eine bestimmte Sicht auf die Daten erlauben.

¹²Modeling of Application Data with Spatio-temporal features

In einem von Bédard & Bernier (2002) vorgestellten Konzept werden die geometrische, semantische und grafische Multiplizität durch Verwendung unterschiedlicher Sichten miteinander kombiniert. Es entstehen verschiedene Kombinationsmöglichkeiten, eine Sicht auf ein bestimmtes Objekt zu formulieren, indem ein sogenanntes *VUEL*, ein *view element* eingeführt wird. Dieses VUEL ist ein Basiselement einer räumlichen Datenbanksicht, so wie das Pixel das Basiselement eines Bildes ist. Die Datenbank enthält in diesem Fall drei Dimensionen (Geometrie, Semantik, Semiotik), wobei jeder Punkt in diesem dreidimensionalen Raum durch ein VUEL repräsentiert wird. Jedes VUEL ist somit mit einer geometrischen, semantischen sowie einer grafischen Klasse verknüpft. Mehrere VUELS wiederum sind durch eine View-Klasse zusammengefasst. Die Geometrieklasse ist mit anderen, alternativen Geometrieklassen verknüpft und enthält zusätzlich Operationen, wie „drill-up“ und „drill-down“, um auf andere Geometrien zurückgreifen zu können. Hierdurch ist es möglich, sich in der geometrischen Dimension dieses Raumes zu bewegen. Das gleiche Konzept wird für den Semantikbereich angewendet. Das Graphikobjekt eines VUEL enthält schließlich ein Objekt „grafische Variablen“ mit den bekannten grafischen Variablen (Form, Farbe, Muster, Helligkeit, Richtung, Größe) als Attribute. Auch hier sind die verschiedenen Repräsentationen dieses Objektes miteinander verknüpft und auch hier sind die Operationen „drill-up“ und „drill-down“ integriert. Die View-Klasse fasst letztendlich die einzelnen VUEL-Klassen zusammen. Auch diese enthält Operationen, um geometrisch, semantisch oder semiologisch durch die unterschiedlichen Repräsentationen zu navigieren. Wird also durch einen *View* ein bestimmter Kartenausschnitt zusammengefasst, so kann durch diese Operationen eine bestimmte Klasse oder eine bestimmte Auflösung der Geometrien gewählt werden.

Der Unterschied zu den oben genannten Konzepten besteht also darin, dass das multirepräsentative Objekt zunächst aufgesplittet wird in die drei Teilbereiche Geometrie, Semantik und Semiotik. Dieses Bereiche werden dann wiederum in einem Objekt, in diesem Fall heißt es VUEL, zusammengefasst. Konzeptionelle Parallelen bestehen im integrierten Modell durch die Verlinkung korrespondierender Objekte. Sowohl Geometrie, Semantik als auch Semiotik dieses multiskaligen Systems sind durch kartografische Generalisierung entstanden und korrespondierende Objekte (sei es in der Geometrie, Semantik oder Semiotik) sind jeweils miteinander verknüpft. Durch die Aufspaltung ist es nun möglich, nicht nur zwischen den Repräsentationen zu wechseln, sondern jeweils nur Teile einer Repräsentation abzufragen und zu manipulieren. Jedes VUEL entspricht somit einer Kombination einer bestimmten Geometrie, Semantik und Darstellung. Auf diese Weise ist es möglich, zum Beispiel die hochauflösende Geometrie eines Gebäude mit einer blau gestreiften Signatur zu versehen und gleichzeitig die Gebäude-Attribute „Funktion“ und „Hausnummer“ anzuzeigen. Alternativ lässt sich eine Sichtweise definieren, welches dieses Objekt als rotes Rechteck mit von anderer Stelle erfassten Attributen, zum Beispiel „Geschosszahl“, darstellt.

Die hier beschriebenen Implementierungen zeigen, dass auf der einen Seite eine Vielzahl individueller Lösungen zur Modellierung multipler Repräsentationen existieren, was aus den jeweiligen Ansprüchen an das System sowie der Verschiedenartigkeit der Datensätze erwächst. Auf der anderen Seite lassen sich diese Verfahren jedoch auf eine der in Unterabschnitt 3.4.2 aufgezeigten, grundsätzlichen Modellierungen zurückführen. Dabei ist festzustellen, dass der am meisten verbreitete Ansatz auf die Einführung übergeordneter Datentypen zurückläuft, welche die gemeinsamen Eigenschaften aller Repräsentationen zusammenfassen ohne die ursprünglichen Schemata zu verwerfen (vgl. Abbildung 3.8, Option 2). Dieses neue Schema lässt sich innerhalb eine föderierten Systems genauso einsetzen wie in einer einzigen, integrierten Datenbank, indem entsprechend den modellierten Übergängen die ursprünglichen Daten in das neue Schema überführt werden. Kann allerdings auf Grund der Ähnlichkeit der Daten auf ein vermittelndes Schema verzichtet werden, so besteht der einfachste Weg darin, die bestehenden Daten direkt in Beziehung zu setzen, wie es das Beispiel *MurMur* zeigt (Abbildung 3.8, Option 3).

3.4.5 Matching der Instanzen

Der Begriff des Matchings umfasst die generische Beschreibung von Methoden und Algorithmen um zwei Datensätze nach Gruppen von Objekte zu durchsuchen, welche denselben Teil der realen Welt repräsentieren (Devegele et al. 1996).

Dabei muss unterschieden werden zwischen dem Matching auf dem Level des Datenschemas, bei dem korrespondierende Objektarten gesucht werden und dem Instanz-Level, bei dem es gilt, einzelne Objekte durch Vergleich ihrer individuellen Merkmale zu verknüpfen. Beide Stufen können nacheinander zur Bestimmung äquivalenter Objekte durchlaufen werden. Dabei ist das Matching Teil der „conflation“¹³. Dieser Begriff wird von Saalfeld (1988) für den Bereich der Geoinformationen definiert und beschreibt die Kombination von zwei oder mehr Datensätzen, um hieraus einen neuen Datensatz zu generieren oder Attribute von einem Datensatz auf den

¹³lateinisch: con flare = zusammenblasen

anderen zu übertragen. Man kann diesen Begriff übersetzen mit „Zusammenführung“ oder „Verschmelzung“. Die Datenintegration stellt dabei einen weiteren Aspekt der „conflation“ dar.

Yuan & Tao (1999) klassifizieren zwei Typen, das horizontale und vertikale conflation. Horizontale Angleichung beschreibt die Anpassung und Zusammenführung von zwei oder mehr Datensätzen entlang einer gemeinsamen Grenzlinie oder einem Überlappungsbereich. Vertikale Angleichung dagegen bedeutet die Vereinigung mehrerer Datensätze, welche dasselbe Gebiet beschreiben. In der vorliegenden Arbeit sollen Datensätze behandelt werden, welche dasselbe Gebiet in unterschiedlicher Ausprägung beschreiben. Daher liegt der Fokus hier auf der vertikalen Anpassung. Diese wird weiterhin zur Bestimmung von Veränderungen in den Datensätzen verwendet. Soll ein älterer Datensatz durch neuere Informationen aufdatiert werden, so dient die vertikale Conflation zur Aufdeckung von Unterschieden und muss entscheiden, ob diese Unterschiede Veränderungen oder Fehler beziehungsweise Ungenauigkeiten in den Datensätzen bedeuten. Somit können neue Informationen auf den älteren in der Datenbank vorliegenden Datensatz übertragen werden.

Eine vollständige Automatisierung des Matchingprozesses ist nach Blasby et al. (2003) aus zweierlei Gründen nicht immer möglich. Zum einen existiert zur Zeit kein Algorithmus, welcher alle Korrespondenzen identifizieren kann. Auch dann nicht, wenn man sich nur auf die Verschneidung von Vektordaten konzentriert. Dieses Problem kann reduziert werden durch die Hinzunahme einer manuellen Interaktionsmöglichkeit. Zum anderen existieren durchaus Fälle, in denen keine eindeutige Zuordnung möglich ist. In diesen Fällen fehlen Informationen, so dass selbst ein menschlicher Operateur nicht vermag, diese Verknüpfung vorzunehmen.

Blasby et al. (2003) unterteilen den Vorgang der Verschneidung in verschiedene Teilprozesse, wobei das Matching der Daten nur einen Teil dieses Prozesses darstellt:

- *Vorverarbeitung der Daten:* In diesem Schritt werden die Daten normalisiert, um sicherzustellen, dass diese kompatibel sind. Sie werden beispielsweise in dasselbe Koordinatensystem transformiert
- *Sicherung der Datenqualität:* Einige Verknüpfungsfälle benötigen Datensätze, welche einen gewissen Grad an innerer Konsistenz besitzen. In diesem Schritt wird somit die innere Konsistenz überprüft und bei Bedarf verbessert. Werden beispielsweise zwei Flächen angepasst, so werden diese zuvor bereinigt, indem Lücken oder Überlappungen innerhalb der Fläche erkannt und beseitigt werden oder diese auf topologische Fehler hin untersucht werden.
- *Angleichung der Datensätze:* In einigen Fällen sind die Datensätze so falsch gegeneinander ausgerichtet, dass eine initiale Ausrichtung erforderlich ist, um eine bessere Anpassung erreichen zu können. Diese Ausrichtung ist in der Regel grober Natur, welche keine Ausrichtung individueller Objekte durchführt. Die erforderliche Transformation wird manuell oder automatisch vorgenommen
- *Matching der Objekte:* In diesem Schritt werden korrespondierende Objekte der Datensätze bestimmt. Dieser Prozess geschieht entweder automatisch durch den Einsatz eines oder mehrerer der weiter unten genannten Verschneidungsverfahren oder durch manuell bestimmte Verknüpfungen
- *Identifizierung der Abweichungen zwischen den Datensätzen:* Hierzu ist es sinnvoll, statistische Zusammenfassungen der Datenqualität der Datensätze, also der Ergebnisse der Sicherung der Datenqualität, zu berücksichtigen, sowie die Diskrepanzen zwischen den Datensätzen zu visualisieren

Matchingverfahren

Die Matchingalgorithmen, welche etwa seit dem Beginn der 90er Jahre entwickelt wurden, konzentrieren sich jeweils auf einzelne Probleme des Matchings und können zunächst unterteilt werden in die geometrischen, topologischen und attributiven Methoden. Yuan & Tao (1999) fassen die Möglichkeiten wie folgt zusammen:

Geometrische Methoden Die Verknüpfung von Datensätzen mit Hilfe geometrischer Überlagerung ist das meist angewendete Verfahren auf dem Gebiet der räumlichen Daten. Dabei kommen bei der Verschneidung Kriterien, wie die Distanz zwischen den Objekten, Richtungsinformationen, Lagebeziehungen oder die Formmerkmale (bei Linien und Polygonen) als Entscheidungsmerkmal zum Tragen.

- *Abstand:* Identische Objekte der realen Welt sollten in verschiedenen Karten dieselbe Position besitzen, vorausgesetzt es liegt kein Fehler vor. Daher ist der Vergleich der Distanzen zwischen den Objekten der offensichtliche Weg, um korrespondierende Objekte zu bestimmen. Ein Objekt mit einem geringen Abstand zu einem Objekt des zu verknüpfenden Datensatzes repräsentiert mit hoher Wahrscheinlichkeit dasselbe Objekt der Realität. Dabei wird die „Euklidische Distanz“ (Gleichung 3.1) benutzt, um Abstände zwischen Punkten oder zwischen Punkten und Linien zu bestimmen:

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (3.1)$$

Die „Hausdorff Distanz“ (Gleichung 3.2 (Hausdorff 1919)) dagegen kann verwendet werden, um den Abstand zwischen zwei Linien zu ermitteln:

$$d_1 = \max_{p_1 \in l_1} \left[\min_{p_2 \in l_2} [dist(p_1, p_2)] \right]; d_2 = \max_{p_2 \in l_2} \left[\min_{p_1 \in l_1} [dist(p_2, p_1)] \right]; d(l_1, l_2) = \max(d_1, d_2) \quad (3.2)$$

Während die euklidische Distanz jeweils den Abstand zwischen zwei Punkte oder einem Punkt und einer Linie berechnet, lassen sich mit Hilfe der Hausdorff-Distanz zwei Punktmenge oder zwei Linien auf ihren Abstand hin untersuchen. Hierzu wird der maximale kürzeste Abstand zwischen beiden Linien gewertet (siehe Abbildung 3.9)

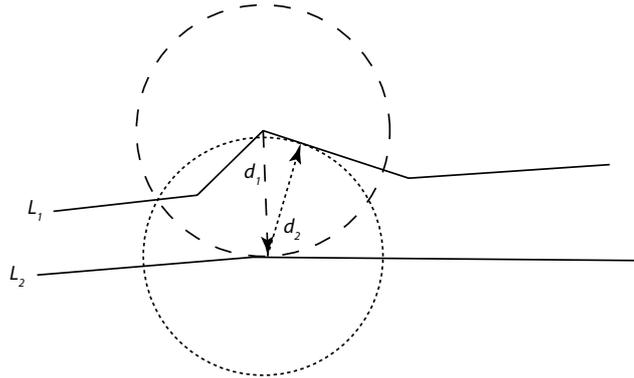


Abb. 3.9: Hausdorff Distanz zwischen zwei Linien

- *Winkelinformationen linearer Objekte:* Werden mehrere Kandidaten durch Abstandskriterien aufgedeckt, helfen Winkel zwischen Linien oder Richtungen von Linien bei der Entscheidung zur Auswahl des korrekten Matchings
- *Lagebeziehungen:* Werden zwei Datensätze miteinander verglichen, so interessiert auch, ob ein Punkt innerhalb oder außerhalb eines Polygons oder auf einer Linie liegt
- *Überlappungen:* Flächenhafte Objekte verschiedener Datensätze überlagern sich zu einem gewissen Prozentsatz. Je größer dieser Prozentsatz ist, desto größer ist die Wahrscheinlichkeit, dass diese Flächen zu verschneiden sind
- *Formmerkmale von Objekten:* Linien oder Polygone besitzen Formmerkmale. Eine Linie hat eine Länge oder Krümmung, ein Polygon besitzt Parameter wie Flächeninhalt, Dichte oder Fourier Form-Deskriptoren (Zahn & Roskies 1972)

Grundvoraussetzung für die Verwendung dieser geometrischen Kriterien ist eine vergleichbare geometrische Basis der beiden Datensätze. Daher gehört zu den vorbereitenden Arbeiten, beide Datensätze in ein einheitliches Koordinatensystem zu übertragen, die Datensätze entsprechend gegeneinander auszurichten und beispielsweise durch Methoden des „rubber sheetings“ (siehe hierzu z.B. Doytsher (2000)) eine geometrische Anpassung zu erreichen.

Topologische Methoden Liegen topologische Informationen in den Datensätzen vor, so können diese genutzt werden, um topologische Ähnlichkeiten in beiden Datensätzen aufzudecken. Topologische Vergleiche sind hauptsächlich zur Unterstützung des geometrischen Matchings sinnvoll, um die Ergebnisse zu validieren oder uneindeutige Fälle lösen zu können. Walter & Fritsch (1999) teilen den Suchbereich möglicher Matchingkandidaten in Teilbereiche auf, welche die Objekte selbst, deren Nachbarn sowie die Nachbarn dieser Nachbarn enthalten. Auf diese Weise wird das Matchingproblem jeweils auf eine lokale Umgebung beschränkt und bezieht sich nicht auf den gesamten Datensatz. Tomaselli (1994) nutzt ausschließlich topologische Informationen, um korrespondierende Inhalte in den Datensätzen zu finden. Voraussetzung hierfür ist eine identische Topologie in beiden Datensätzen. Unterschiede können dann beispielsweise in der geometrischen Auflösung oder Genauigkeit bestehen, so dass in diesem Fall ein topologisches Matching besser funktioniert als das geometrische Matching und somit Informationen von einem auf den anderen Datensatz übertragen werden können, ohne die Geometrien anpassen zu müssen.

Semantische Methoden Die Verknüpfung von Datensätzen anhand von semantischen Informationen basiert auf den gemeinsamen Attributen der Objekte. Nachdem diese bestimmt wurden, muss sichergestellt sein, dass in beiden Datensätzen die Attribute dieselbe Bedeutung besitzen. Gleichzeitig müssen Verfahren existieren, um Attributwerte des einen Datensatzes auf die Semantik des zweiten transformieren zu können, um beide vergleichen zu können. Ein einfacher Fall wäre zum Beispiel die Nutzung von Straßennamen zur Identifizierung korrespondierender Objekte (Sester et al. 1998).

Kombination der Methoden Abhängig von den verfügbaren Informationen ist eine Kombination der oben genannten Verfahren zu verwenden. So können sich die Verfahren ergänzen, um etwa die Ergebnisse des einen Verfahrens durch ein zweites zu validieren oder um bei nicht eindeutigen Fällen eine Entscheidung treffen zu können. Dabei ist das angewendete Verfahren anhängig von den Datensätzen, welche miteinander verschnitten werden sollen, in räumlichen Vektordatensätzen also die jeweilige Kombination der Grundelemente Punkt, Linie und Fläche. So sind beim Matching von Punktelementen zu Punktelementen beispielsweise Euklidische Distanzen anzuwenden und beim Matching von Linie zu Linie Hausdorff-Distanzen.

Ähnlichkeitsmaße Um bestimmte Charakteristika einzelner Objekte miteinander vergleichen zu können, ist es sinnvoll, Ähnlichkeitsmaße aus diesen Eigenschaften zu berechnen. Bei Samal et al. (2004) findet sich eine umfassende Beschreibung unterschiedlichster Ähnlichkeitsmaße. Die Gruppe der Ähnlichkeitsmaße lässt sich unterteilen in kontextunabhängige und kontextabhängige Maße. Kontextunabhängige Maße vergleichen Skalare, also Größen, welche durch eine reelle Zahl ausgedrückt werden sowie Zeichenketten, Lage- oder Formähnlichkeiten. Der geografische Kontext lässt sich in Form von Nachbarschaftsgraphen modellieren (Stadler 2004). Aus diesen Nachbarschaftsgraphen lassen sich dann sogenannte Cluster bilden. Anders & Bobrich (2004) beschreiben beispielsweise die Möglichkeit, aus verschiedenen, hierarchisch gegliederten Nachbarschaftsgraphen (Delaunay-Triangulation, Gabriel-Graph, Relativer Nachbargraf, Nächster-Nachbar-Graph) Punktmengen zu übergeordneten Strukturen, also Clustern, zusammenzufassen.

Beispiele zum Matching von Geodaten

Erstmalig wurde das Prinzip des „conflatings“ im Jahre 1985 implementiert (Saalfeld 1988). In diesem Fall ging es darum, die Daten des *U.S. Bureau of Census* (Amt für Statistik der USA) mit geologischen Daten des USGS¹⁴ zu verknüpfen. Die statistischen Daten sind sehr reichhaltig, besitzen dagegen minderwertige Geometrien. Das Ziel war es daher durch Verschneidung der Daten diese Informationen auf die höherwertigen Geometrien des USGS übertragen zu können (Brown et al. 1995).

Dunkars (2004b) beschreibt in seiner Arbeit eine Methode zur Verknüpfung zweier Datensätze unterschiedlicher Maßstäbe. Die angewendete Methode kombiniert die euklidische Distanz zwischen den Objekten mit einer auf bestimmten Attributwerten beruhende Wertigkeit. Dieser auf der Semantik beruhende Parameter wird abgeleitet aus einem zuvor durchgeführten manuellen Matching zwischen beiden Maßstäben, um herauszufinden, welche Objekte bzw. welche Objekteigenschaften, wie zum Beispiel der Straßentyp, in beiden Maßstäben wie häufig miteinander verknüpft werden.

Sester et al. (1998) stellen drei Ansätze vor, um Objekte in räumlichen Datensätzen miteinander in Verbindung zu setzen. In einem ersten Ansatz werden Verknüpfungen auf Grund geometrischer sowie topologischer Charakteristika, in den anderen beiden mit Hilfe semantischer Informationen gebildet. Walter & Fritsch (1999) beschreiben einen Ansatz zur Verknüpfung von linienhaften Datensätzen ähnlicher Maßstäbe, wobei hier auf Verfahren aus der Informationstheorie zurückgegriffen wird. Zwei manuell verknüpfte Datensätze werden statistisch analysiert, um dieses Wissen auf die zukünftigen Fälle übertragen zu können. Bei dieser im zweiten Schritt automatischen Vorgehensweise kommen geometrische und topologische Informationen der Daten zum Einsatz. Es ist hier insbesondere möglich, alle denkbaren Matching-Kardinalitäten zu bestimmen. Durch ein Verfahren, welches einen wachsenden Puffer verwendet, ist es auch möglich, n:m Beziehungen zu detektieren, was mit den anderen genannten Verfahren nicht durchführbar ist. Gabay & Doytsher (1994) beschreiben eine Methode, welche darauf abzielt, Datensätze mit geringen geometrischen aber größeren topologischen Unterschieden zu matchen. Filin & Doytsher (2000) betrachten die Knoten der zu verknüpfenden Objekte an Stelle der gesamten Objekte, in diesem Fall den Linien. Devogele (2002) definiert eine Technik, welche statt der Objekte korrespondierende Punkte der Objekte identifiziert.

¹⁴United States Geological Survey

Die beschriebenen Verfahren eignen sich gut, um ähnliche Datensätze miteinander verknüpfen zu können. Bei unterschiedlichen Maßstäben treten jedoch auf Grund der Verschiedenheit der Datensätze Probleme auf, wenn diese direkt gematcht werden sollen. Einige der aufgeführten Kriterien greifen zwar bei unterschiedlichen Maßstäben. So finden sich zum Beispiel durchaus semantische Informationen, wie Straßennamen, die einen Vergleich erlauben. Gerade die geometrischen Kriterien wie Länge, Flächeninhalt oder Formmerkmale werden jedoch durch die Generalisierung verfälscht. Es ist daher notwendig, die Auswirkungen der Generalisierung im Matchingprozess zu berücksichtigen. Jones et al. (1996) stellen ein System auf, welches mit Hilfe von Regeln Datensätze unterschiedlicher Maßstäbe verknüpft. Ziel war es, neue Datensätze mit den Daten in der Datenbank zu matchen, um Aktualisierungen einspielen zu können. Es wird der Datensatz mit dem größeren Maßstab generalisiert, um mit den Daten des kleineren Maßstabes vergleichbar zu sein. Die resultierenden Matchingkandidaten werden dann weiter untersucht im Hinblick auf semantische und geometrische Ähnlichkeit.

Der von Sester et al. (1998) beschriebene Ansatz wird auf Datensätze unterschiedlicher Maßstäbe übertragen, indem auch diese durch Generalisierung zunächst in einen einheitlichen Maßstab transformiert werden. In dem beschriebenen Beispiel werden durch Aggregationsregeln Gebäude in Siedlungsflächen umgewandelt. Die Regeln beschreiben die Übergänge zwischen den verschiedenen Datenbeständen. Hierzu müssen diese Zusammenhänge jedoch a priori bekannt sein, um sich formalisieren lassen. Sind diese nicht von Beginn an bekannt, so können Techniken des maschinellen Lernens eingesetzt werden, um diese aus bekannten Beispielen, wie bestehenden Maßstabsreihen, abzuleiten (Sester 2000).

An dieser Stelle bleibt festzuhalten, dass die einzelnen, vorgestellten Lösungen und Verfahren sich für bestimmte Teilbereiche des Matchingproblems eignen, andere Datensätze dagegen alternative Algorithmen bedingen. Einige der vorgestellten Lösungen arbeiten ausschließlich mit linienhaften Objekten, andere gehen davon aus, dass die zu verknüpfenden Datensätze Ergebnisse einer Generalisierung sind. Da es schwierig ist, alle Probleme des Matchings mit einem Algorithmus zu lösen, wird von Yuan & Tao (1999) die Strategie der komponentenbasierten Softwareentwicklung vorgeschlagen. Die einzelnen Komponenten können entsprechend dem aktuellen Problem individuell kombiniert werden.

3.5 MRDB Standardoperationen und Nutzerinterfaces

In den vorangegangenen Abschnitten wurden zahlreiche Ansätze zur Modellierung multipler Repräsentationen vorgestellt. Um diese Modelle in einer Datenbank umzusetzen, wird das entwickelte Datenmodell durch eine DDL¹⁵ auf das DBMS übertragen. Es werden neue Datentypen eingeführt, sogenannte ADTs¹⁶, aufbauend auf den bestehenden ADTs. Es gilt dabei, die herkömmlichen Datentypen um räumliche sowie multirepräsentative Komponenten zu ergänzen.

Neben Datenschemata und -strukturen erfordern multiple Repräsentationen auch neue Verfahren und Operatoren zur Datenabfrage. Es gilt nicht nur, heterogene Daten in einer homogenen Datenstruktur effektiv zu organisieren. Darüber hinaus erfordert eine nachhaltige Datennutzung auch Strukturen, um auf die alternativen Repräsentationen gezielt zugreifen zu können. Vorhandene Funktionalitäten müssen erweitert, neue Operationen ergänzt werden, um vorhandene GIS oder andere Nutzerschnittstellen auch für multiple Repräsentationen nutzbar zu machen. Die Problematik stellen an dieser Stelle die Mehrfachrepräsentationen dar, welche, wie weiter oben erläutert wurde, unterschiedliche Abfrageergebnisse hervorrufen können, abhängig von der jeweils involvierten Repräsentation. Unterschiedliche Auflösungsstufen beinhalten gleichzeitig unterschiedliche Geometrien, Attribute, Instanzen, Relationen und relative Informationen (Topologie) (Burnet 2000). Daher ist es zum Beispiel wichtig, dem Nutzer die Möglichkeit zu geben, eine spezifische Repräsentation für seine Anfrage zu bestimmen oder die Informationen von einer Repräsentation auf eine andere übertragen zu können. Wesentlich ist daher, auf Grundlage der heterogenen Daten, kohärente Resultate zu produzieren. Zur Integration dieser Funktionalitäten in die Datenbank wird eine DML¹⁷ aufgebaut oder eine bestehende mit neuen Funktionalitäten ergänzt. Eine DML ist der Teil der Datenbanksprache, mit deren Hilfe die Daten gelesen, geschrieben, geändert oder gelöscht werden können. In dem folgenden Abschnitt sollen die für eine MRDB relevanten Abfragefunktionen herausgestellt werden.

Burnet (2000) beschreibt in dem Projekt *MurMur* zusammengestellte Abfragen und Anforderungen, die von einer räumlichen Datenbank, welche multiple Repräsentationen beinhaltet, zur Verfügung gestellt werden sollten.

¹⁵Data Definition Language

¹⁶Abstract Data Types

¹⁷Data Manipulation Language

Auch wenn diese Beschreibungen auf das in *MurMur* entwickelte Datenschema abgestimmt und auf die dort zugrunde gelegten Anwendungsfälle ausgerichtet sind, können diese allgemeingültig auf die Erfordernisse anderer Schemata übertragen werden. Hierbei wird unterschieden zwischen elementaren Abfrageoperationen, welche direkt durch die in der Datenbank integrierten Abfragesprache ausgedrückt werden sollen und Servicefunktionen, welche ihrerseits aus einer Reihe von Datenbankabfragen bestehen. Die verschiedenen Funktionen können daher hierarchisch aufgegliedert werden, wobei die grundlegenden und elementaren Abfragen auf der Datenbankseite zu sehen sind, während in Richtung der Nutzerapplikation komplexere Operationen zu finden sind, welche auf diese Basisfunktionalitäten der Datenbank zurückgreifen.

- *Import/Export-Funktionalitäten* erlauben einen Import oder Export von Daten oder Schemata
- *Validierungs-Funktionen* stellen die Konsistenz der Daten sicher
- *Datenmanagement* umfasst das Löschen, Erstellen und Aktualisieren der Daten sowie die Weiterleitung von Operationen durch verschiedene Repräsentationen und die Darstellung von Objekten mit Hilfe der verschiedenen Repräsentationen
- *Analyse-Funktionen* ermöglichen es, verschiedene Informationen bezüglich der Objekteigenschaften abzuleiten (z.B. Flächeninhalt, Längen etc.) oder miteinander zu vergleichen (z.B. Verschneidungen)
- Die *Konsultation* ermöglicht die Auswahl und Präsentation der Daten durch grafische oder logische Interaktionen sowie die Erstellung und Auswahl einer bestimmten Sichtweise auf diese

Einige Funktionalitäten aus diesem Bereich sind in Tabelle 3.1 exemplarisch aufgelistet. Diese können ihrerseits

<i>Kategorie</i>	<i>Funktion</i>	<i>Beschreibung</i>
Import/Export	Datenimport	Überträgt die Daten aus dem externen in das interne Schema
Validierung	Konsistenzprüfung multipler Repräsentationen	Prüft die Konsistenz zweier verlinkter Repräsentationen. Ermöglicht zum Beispiel die Kontrolle, ob zwei Attributwerte, welche gleich sein sollten, nicht zu unterschiedlich sind
Datenmanagement	Daten editieren	Daten werden modifiziert oder durch den Nutzer generiert
	Operationen durch Repräsentationen propagieren	Eine Operation wird auf ein Objekt einer Repräsentation angesetzt, die Auswirkungen dieser Operation werden auf die anderen Repräsentationen übertragen
	Übertrage Attribute	Überträgt die Attribute eines Objektes von einer Repräsentation auf die homologen Attribute einer alternativen Repräsentation
	Übertrage Geometrie	Überträgt die Geometrie eines Objektes von einer Repräsentation auf die homologe Geometrie einer alternativen Repräsentation
Analyse	Erstelle Link	Erstellt eine neue Verknüpfung zwischen zwei Objekten
	Analysefunktionen	Distanz, Puffer, Verschneidung etc. kann Objekte derselben oder unterschiedlicher Repräsentationen miteinander in Beziehung setzen, z.B. zum Zweck des Data Minings
	Vergleich	Vergleicht die Semantik, Geometrie oder Topologie zweier Repräsentationen und stellt die unterschiedlichen Werte dar
	Statistik	Es werden die Eigenschaften mehrerer Objekte aus verschiedenen Repräsentationen zusammengefasst, zum Beispiel die durchschnittliche Länge aller Straßensegmente aller Repräsentationen
	Conflation Attribute	Stellt für ein Objekt einer Repräsentation eine Liste aller Attribute aus den verschiedenen Repräsentationen dar
Konsultation	Conflation Links	Stellt eine Liste aller Verknüpfungen zu den verschiedenen Repräsentationen dar
	Wahl der Sichtweise	Es werden nur die Daten einer bestimmten Sichtweise oder einer bestimmten Auflösung dargestellt; adaptive Darstellung
	Verknüpfungen anzeigen	Finde alle Objekte, welche mit einem anderen Objekt in Beziehung stehen

Tab. 3.1: Auswahl einiger grundlegender Funktionen eines multirepräsentativen Systems

für komplexere Funktionen genutzt werden, welche spezifischer auf eine bestimmte Anwendung zugeschnitten sind. Steht ein derartiges System zur Verfügung, welches unterschiedliche Repräsentationen in einem konsistenten Datenmodell vereint, so ergibt sich eine Vielzahl von Anwendungsmöglichkeiten durch die Kombination dieser. Jede Anwendung greift auf die DML und die elementaren Servicefunktionalitäten zurück.

Neben einem integrierten System und einer spezifischen DML ist es sicherlich notwendig, das vom Anwender zu bedienende Interface (GIS o.ä.) mit zusätzlichen, dem multirepräsentativen System entsprechenden Funktionalitäten auszustatten. Erst hierdurch wird es dem Anwender ermöglicht, die Vorteile des integrierten und interoperablen Systems auszuschöpfen. Während die bis hierher beschriebenen Basis-Funktionalitäten auf der Seite der Datenbank angegliedert sind, sollen im Folgenden einige Funktionalitäten beschrieben werden, welche sich auf der Nutzerseite befinden und auf die multirepräsentativen Inhalte und Funktionen zurückgreifen und diese zu komplexeren Operationen kombinieren.

Die nachfolgend aufgelisteten Funktionalitäten wurden im Projekt *MurMur* auf Grundlage spezifischer Anwendungsfälle entwickelt. Im ersten genannten kartografischen Anwendungsfall sollen in einer Landesvermessungsverwaltung die Kartendaten aller Maßstäbe in einem integrierten System zusammengefasst und zentral verwaltet werden. Für diesen Fall ist es vorrangig, auf die unterschiedlichen Darstellungen zugreifen zu können. In dem zweiten Anwendungsfall sollen Geoinformationen unterschiedlicher Inhalte und Maßstäbe in ein einheitliches System integriert werden, um zur Risikoanalyse genutzt werden zu können. Hieraus sind folgende Funktionalitäten der Nutzerschnittstelle denkbar:

- Abfrage aller verfügbaren Informationen aus den verschiedenen Repräsentationen: Jede Repräsentation enthält eigenständige Informationen geometrischer oder semantischer Art, welche in einer anderen nicht zu finden sind. Somit entsteht ein enormer Vorteil, wenn alle verfügbaren Informationen abrufbar sind und bei Bedarf gefiltert werden können
- Kartografische Darstellung aller verfügbaren Repräsentationen, sei es durch Auswahl einer bestimmten Repräsentation (beispielsweise die Darstellung einer bestimmten Straße aus einem bestimmten Datensatz) oder aller verfügbaren Repräsentationen gleichzeitig. Beide Funktionalitäten sind sinnvoll, um beispielsweise eine optimale Darstellung selektieren zu können
- Möglichkeit, alle verfügbaren Informationen zu einer kartografischen Darstellung zu kombinieren. So enthält ein Datensatz die detaillierte Geometrie einer Straße, ein anderer die Anzahl der Fahrspuren und ein dritter den Namen der Straße. Somit wird es möglich, eine reichhaltigere und vollständigere Kartendarstellung zu erreichen
- Abruf der semantischen Informationen einer oder aller verfügbaren Repräsentationen in Form einer Liste. Auf diese Weise besteht Zugriff auf alle Informationen, welche mit einem bestimmten Phänomen verknüpft sind
- Abfrage aller Details, welche allen Repräsentationen entweder gemeinsam sind oder welche die Unterschiede ausmachen zwischen den verschiedenen Repräsentationen: Das System sollte also in der Lage sein, gleiche Informationen als solche zu identifizieren, um die vorhandenen Daten richtig zu interpretieren und um Redundanz zu vermeiden. Auch die Analyse der Unterschiede hilft, die vorhandenen Informationen zu interpretieren und gleichzeitig unterstützt diese Funktion die Kontrolle der Datenkonsistenz
- Geometrische und topologische Interpretation der Unterschiede und Gemeinsamkeiten: Geometrische Indikatoren zwischen zwei oder mehr Repräsentationen sind wichtig, um die unterschiedlichen geometrischen Analysefunktionen (Länge, Flächeninhalt, Abstände etc.), welche allesamt mathematisch korrekte Ergebnisse liefern, in ein konsistentes Abfrageergebnis umzuwandeln. Neben den geometrischen Unterschieden sollte ein System auch die topologischen Zusammenhänge unterschiedlicher Repräsentationen beurteilen können. Dies kann zum Beispiel vorteilhaft sein, wenn unterschiedliche Repräsentationen in einer Karte kombiniert werden sollen
- Auswahl der Daten durch grafische Interaktion: In einem multirepräsentativen System kann der Nutzer durch Interaktion Daten aus jeder Repräsentation anfordern. Er kann bestimmte Objekte auswählen und die Darstellung dieser mit Hilfe alternativer Repräsentationen fordern
- Darstellung aller verfügbaren Informationen einer bestimmten Parzelle
- Darstellung der Objekte in Repräsentation A (z.B. Gebäude), welche sich innerhalb bestimmter Objekte des Repräsentation B (z.B. Risikogebiete) befinden

Die beschriebenen Funktionalitäten wurden in dem genannten Projekt implementiert und stehen dem Anwender in Form eines Query-Editors zur Verfügung, welcher die Funktionen zu komplexeren Abfragen kombiniert und dem Nutzer auf grafischem Wege erlaubt, Abfragen an das multirepräsentative System zu formulieren.

Eine weitere Möglichkeit dieses Services sieht vor, dass nur an bestimmten Stellen in der Karte eine bestimmte Auflösung eingesetzt wird, der Rest der Karte wird mit einer niedrigeren oder höheren Auflösung dargestellt. Es können einzelne ausgewählte Objekte oder aber ganze Zonen ersetzt werden durch alternative Repräsentationen. Probleme treten hierbei vor allen Dingen in den Randbereichen auf. Hier muss dafür gesorgt werden, dass die unterschiedlichen Repräsentationen nahtlos, das heißt ohne Versatz, zusammenpassen. Überlappungen oder andere geometrische Fehler müssen entsprechend korrigiert werden (siehe Abbildung 3.10). Hierzu wurden Anpassungsalgorithmen entwickelt, welche korrespondierende Objekte an den Schnittstellen selektieren und die Enden dieser Geometrien (z.B. Straßen) zusammenführen. Hierbei helfen die Verknüpfungen zwischen den Repräsentationen. Weiterhin integriert das System eine Generalisierungsfunktionalität durch Interpolation der



Abb. 3.10: Kombination zweier LoD in einer Karte (Vaughlin 2001)

vorhandenen Repräsentationen, um Darstellungen in Maßstäben generieren zu können, für die keine Daten vorliegen (eine Implementierung dieser Idee findet sich bei Cecconi (2003)).

Das von Bédard & Bernier (2002) beschriebene VUEL Konzept (siehe Abschnitt 3.4.4) enthält eine Nutzerschnittstelle in Form einer grafischen Oberfläche zum Zugriff auf die multirepräsentativen Funktionalitäten des Systems. In dem System besteht die MRDB aus drei Dimensionen, der Geometrie, Semantik und Grafik, welche jeweils in unterschiedlichen Repräsentationen auftreten. Die Nutzerschnittstelle erlaubt es nun, diese Variablen zu verändern. Der beschreibende Teil dieser Schnittstelle ermöglicht die Selektion von Objekten anhand der Objektklasse, wobei die Möglichkeit besteht, allgemeine oder spezifische Klassen zu selektieren, also sich in den Hierarchien zu bewegen. Der kartografische Teil erlaubt die Bestimmung der Auflösung einer Darstellung, während das grafische Element eine Veränderung der Darstellung durch Variation der grafischen Veränderlichen realisiert. Schließlich besteht die Möglichkeit, zwischen vordefinierten Sichtweisen mit bestimmtem Maßstab, semantischer, geometrischer und grafischer Repräsentation zu wählen. Weiterhin kann der Anwender eine der integrierten Multiplizitäten für ausgewählte Objekte variieren. So ist es beispielsweise möglich, bei bestimmten Objekten detailliertere oder weniger detaillierte Darstellungen zu erlangen. Dieses Konzept ist ähnlich dem aus dem Bereich des OLAP stammenden „drilling“, in dem auch eine bestimmte Variable bei der Abfrage der Daten verändert wird.

Timpf & Devogele (1997) beschreiben die Erweiterung bestehender Nutzerinterfaces zur Manipulation und Visualisierung multiskaliger Daten. Es werden die typischen Funktionen zur Interaktion, respektive *vergrößern/verkleinern*, *suchen*, *selektieren* und *verschieben* variiert sowie die Funktion *aktualisieren* neu eingeführt. Die letztgenannte Funktion erlaubt die Übertragung der Veränderung eines Objektes, beispielsweise der Farbe, auf die korrespondierenden Objekte. Die Änderungen sind hier jedoch lediglich temporär und werden nicht in die Datenbank übertragen. Die Zooming-Funktionalität bietet neben dem einfachen grafischen Zoom auch die Möglichkeit eines „inhaltlichen Zoomings“ sowie einer „intelligenten Vergrößerung“. Das „inhaltliche Vergrößern“ erlaubt die Abfrage zusätzlicher oder detaillierterer thematischer Informationen zu einem bestimmten Objekt, während das „intelligente Vergrößern“ mehr Details, also einen größeren LoD darstellt (siehe Abbildung 3.11). Die Auswahlwerkzeuge müssen in der Form erweitert werden, dass der Nutzer in einer bestimmten Repräsentation die Objekte auswählen kann und diese Auswahl dann auf andere Repräsentationen automatisch übertragen wird. Ergänzt wird das Auswahlwerkzeug vom Browser Tool, welches es ermöglicht, semantische Informationen von verschiedenen Repräsentationen zu konsultieren. Wird ein Bildausschnitt verschoben, so wird mit Hilfe erweiterter Verschiebungswerkzeuge auch eine Verschiebung in den übrigen Fenstern vorgenommen, welche die alternativen Repräsentationen beinhalten.

Die beschriebenen Empfehlungen und Ideen bezüglich der Grundfunktionalitäten einer multirepräsentativen Datenbank sowie den Möglichkeiten einer Nutzerschnittstelle sollen bei der Entwicklung des in der vorliegenden Arbeit beschriebenen MRDB-Service aufgegriffen werden.

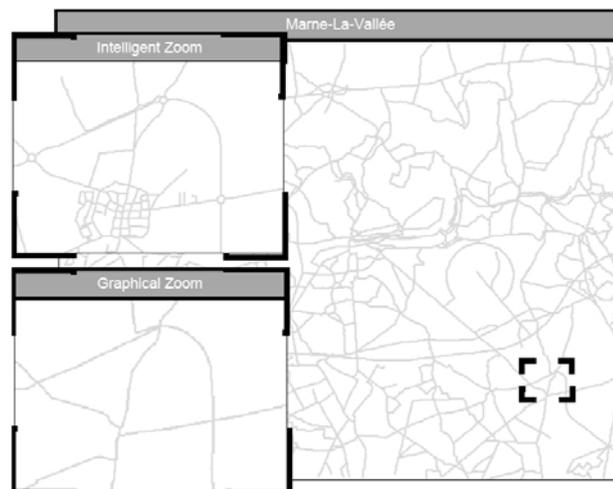


Abb. 3.11: Intelligentes Zooming (Timpf & Devogele 1997)

3.6 Zusammenfassung und Schlussfolgerungen

In dem vorangegangenen Kapitel wird der status quo auf dem Forschungsgebiet der MRDB ausführlich dargelegt. Es werden zunächst die Gründe beschrieben, warum man sich ursprünglich mit dieser Thematik beschäftigte. Unterschiedlichste Einsatzgebiete multirepräsentativer Datenstrukturen werden aufgelistet. Grundsätzlich besteht das Ziel darin, alle für ein bestimmtes Gebiet vorhandenen Informationen in einem System zusammenzufassen und zu integrieren, um auf diese Informationen zugreifen zu können. Gleichzeitig wird in einem multirepräsentativen System die Konsistenz aller integrierten Daten sichergestellt. Unterschiedliche, alternative Repräsentationen werden unter anderem zur Visualisierung genutzt. Verschiedene Kartenmaßstäbe können in einem solchen System integriert werden. Hier profitiert nicht durch die Darstellungskomponente, sondern vor allem auch die Datenhaltung von einem konsistenten System. Des Weiteren bietet die Interoperabilität gerade bei Analysen enorme Vorteile, da der Zugriff auf alle vorhandenen Informationen gewährleistet ist und somit die optimale Repräsentation herausgefiltert werden kann oder aber möglichst viele unterschiedliche Quellen für die Analyse verwendet werden können.

Untersuchungen im Rahmen des EU-Projektes *MurMur* haben gezeigt, dass aktuelle DBMS keine adäquaten Funktionalitäten bereitstellen, um multiple Repräsentationen zu verwalten; ebensowenig derzeitige GIS, welche lediglich das Prinzip der Views oder der maßstabsabhängigen Darstellung unterstützen und teilweise die Verwendung mehrerer Geometrien für ein Objekt erlauben (Zimányi 2000). Beide stellen nur unzulängliche Funktionalitäten beim Umgang mit multiplen Repräsentationen von Geodaten zur Verfügung (Balley et al. 2004). Das genannte Projekt hat hier erstmalig ein System vorgestellt, welches multiple Repräsentationen verarbeiten und verwenden kann. Sind dem entwickelten System multiple Repräsentationen bekannt, so kann dieses die Konsistenz durch entsprechende Regeln, welche Teil der Datenbeschreibung sind, prüfen. Weiterhin unterstützt dieses System angemessene Anfragen und erlaubt dem Nutzer, sich zwischen den verschiedenen Repräsentationen zu bewegen (Vangenot et al. 2002).

Es wird herausgestellt, dass es nicht nur darauf ankommt, das Datenmodell entsprechend zu gestalten und alle Repräsentationen in dieses zu integrieren. Vielmehr ist es wichtig, auch das übrige System dementsprechend anzupassen, um alle Vorteile einer multiskaligen Struktur nutzen zu können.

Die vorliegende Arbeit greift auf diese Erkenntnisse zurück und versucht dies auf die Architektur eines Webdienstes zu übertragen. So sollen auch das Zugriffs-Interface auf die MRDB (vgl. Kapitel 6) sowie die grafische Nutzerschnittstelle (vgl. Kapitel 8) der multirepräsentativen Struktur genüge leisten.

4 MRDB-Webdienst: Definition, Probleme und Lösungen

Die Vorteile der Webservices und Geo-Webservices werden im Kapitel 2 ausführlich dargelegt. Webdienste finden sowohl auf Seiten der Entwickler als auch der Nutzer begründete Akzeptanz. In dem darauffolgenden Kapitel 3 werden Argumente zur Organisation multirepräsentativer Datensätze in dem System einer MRDB vorgetragen. Jeder dieser beiden Themenkomplexe besitzt, auch in dem Bereich der Geoinformatik, jeweils seine Existenzberechtigung. In dem folgenden Kapitel soll nun eine Symbiose beider Technologien beschrieben werden, um die Vorteile einer MRDB auch innerhalb eines Webdienstes nutzen zu können und gleichzeitig, um hierdurch auftretende Probleme aufzudecken und Lösungsansätze zu finden.

Zahlreiche Gründe sprechen dafür, Informationen und Dienste per Webservice bereitzustellen (vgl. hierzu Abschnitt 2.1). So werden hierdurch eine Harmonisierung der Geschäftsprozesse, Datenkonsistenz, Flexibilitätssteigerung, Integrationsfähigkeit und Wiederverwendbarkeit der Daten und Dienste erreicht. Der Zugang zu den Komponenten erfolgt ausschließlich über Schnittstellen, deren Syntax und Semantik durch Spezifikationen geregelt ist, ohne dass die Komponente selbst verändert werden muss. Webservices bieten somit eine höhere Flexibilität als herkömmliche Softwarekomponenten. Gleichzeitig ist es hierdurch möglich, einzelne Implementierungen einem größeren Anwenderkreis zur Verfügung zu stellen, was sowohl für den Anbieter als auch den Nutzer eine höhere Effizienz in der Komponentenentwicklung und Datenerfassung bedeutet. Webdienste können mit Hilfe der Nutzung innovativer IT die Probleme der Interoperabilität, Portabilität, Integration und Kommunikation zwischen Anwendungen und IT-Systemen lösen. Die Umsetzung bleibt hierbei unabhängig von den eigenen Systemen, wie beispielsweise dem Betriebssystem.

Für einen großen Teil dieser Dienste sind Geodaten fundamental. Dabei ist diese Datengrundlage in vielen Fällen Mittel zu dem Zweck, einen bestimmten Service ausführen zu können. So benötigt ein Routing-Service einen Datenbestand, der Verkehrswege inklusive den Abbiegevorschriften, Ausbauzuständen oder ähnlichen Angaben beinhaltet. Andere Dienste bieten einen direkten Zugriff auf die entsprechenden Geoinformationen inklusive Such- oder Filterfunktionen. So ermöglichen einige Landesvermessungsämter in Deutschland, wie zum Beispiel die Landesvermessung Brandenburg¹, den Zugriff auf ihre Daten per WFS oder WMS.

Diese Datenbasis kann nun einen einfachen Datensatz enthalten, welcher die Funktionalitäten der angeschlossenen Dienste ermöglicht. Kapitel 3 hat jedoch gezeigt, dass es für viele Anwendungen notwendig oder zumindest vorteilhaft ist, auf mehr als eine Datenquelle zurückgreifen zu können. Kartendarstellungen benötigen beispielsweise für verschiedene Maßstäbe unterschiedlich aufbereitete und generalisierte Daten; für Analysezwecke ist es sinnvoll, Daten unterschiedlicher Zeitpunkte oder Auflösungen vorzuhalten. Andere Anwender wollen möglichst viele Informationen und somit Daten über ein bestimmtes Phänomen erhalten. Die Gründe für die Verwendung einer MRDB zur Integration heterogener, multirepräsentativer Datenbestände in ein konsistentes System verlieren dabei auch im Rahmen eines Webdienstes nicht ihre Gültigkeit. Die Integration einer MRDB in eine Webservice-Umgebung führt daher zu einer Summierung der Vorteile beider Komponenten.

In dem Bereich der Geoinformationen ergeben sich auf diese Weise eine Reihe neuer Nutzungsmöglichkeiten. In der vorliegenden Arbeit soll der Fokus insbesondere auf der Verwendung multiskaliger Datensätze liegen. Hierdurch finden sich neue Methoden und Möglichkeiten zur Abfrage und Darstellung geografischer Informationen. Die Verwendung multiskaliger Datensätze bewältigt die Probleme einer notwendigen Generalisierung auf pragmatische Weise, indem vorbereitete Datensätze für die verschiedenen Maßstabbereiche vorgehalten werden. Gleichzeitig werden durch die Verwendung unterschiedlicher Auflösungsstufen die Schwierigkeiten der kartografischen Darstellungen auf den kleinen Displays mobiler Geräte ansatzweise gelöst. Schließlich hat der Nutzer zu Analysezwecken nun die Möglichkeit, durch alle Repräsentationen zu navigieren, um die für ihn relevanten Informationen zu finden, sofern diese nicht automatisch vom System zur Verfügung gestellt werden. Die im folgenden Abschnitt aufgeführten Szenarien verdeutlichen die Möglichkeiten, die eine MRDB hier bietet.

Unterabschnitt 2.3.2 zeigt, wie ein Geo-Webservice grundsätzlich aufzubauen ist und welche Besonderheiten es zu beachten gilt. Entwicklungen des W3C haben ein Grundgerüst zur Realisierung eines Webdienstes aufgestellt und das OGC hat weiterführende Service-Spezifikationen für Geo-Webdienste bereitgestellt, auf die in dem folgenden Kapitel zurückgegriffen werden soll.

¹Geoservice Landesvermessung Brandenburg: <http://geoservice.geobasis-bb.de>

In Kapitel 3 werden unterschiedliche Möglichkeiten zur Modellierung einer MRDB sowie zur Integration heterogener Datenbestände in eine solche aufgeführt. Es gibt verschiedene Wege, unterschiedliche Datenquellen in ein konsistentes Datenmodell zu integrieren. Es stellt sich an dieser Stelle die Frage, welche Anforderungen von dem Datenmodell einer MRDB zu erfüllen sind, wenn diese von einem Webservice genutzt werden soll. Auf der anderen Seite bedingt auch die Webservice-Architektur eine Anpassung bei der gleichzeitigen Verwendung einer multiskaligen Datenbank. Der MRDB-Webservice sollte Funktionalitäten bereithalten, um deren Vorteile und Fähigkeiten auszuschöpfen. Diese Funktionalitäten sind dabei natürlich immer abhängig von der jeweiligen Anwendung. Trotzdem soll hier versucht werden, einen Katalog grundlegender Funktionalitäten zu erarbeiten, welche jedem Nutzer zur Verfügung stehen sollten. Das OGC hat zur Abfrage von Geodaten aus Datenbanken Services und Schnittstellen spezifiziert, wie den WFS oder WMS. Es ist zu klären, ob diese Spezifikationen auch dazu genutzt werden können, die Möglichkeiten einer MRDB, welche über die einer herkömmlichen Datenbank hinausgehen, auszuschöpfen. Es wird sich im weiteren Verlauf dieses Kapitels zeigen, dass es einer Erweiterung bestehender Services bedarf, um einem Anwender das volle Potenzial einer MRDB bieten zu können. Die größte Herausforderung stellen dabei die einem bestimmten Objekt zugeordneten Mehrfachrepräsentationen dar.

Somit wird einem Anwender durch den MRDB-Webservice ein Werkzeug an die Hand gegeben, welches ihm einen effizienten Zugriff auf die verfügbaren Informationen liefert. Zum anderen sollen auch andere Webservices, wie beispielsweise Kartendienste, von diesem profitieren, indem hierdurch zum Beispiel neue Formen der Geodatensvisualisierung ermöglicht werden. Sowohl der direkte Zugriff durch den Anwender, respektive einer Nutzerapplikation, als auch die Verwendung durch andere Services soll somit gewährleistet sein.

Der folgende Abschnitt ermittelt zunächst durch szenarienbasiertes Design die Anforderungen an das zukünftige System. Verschiedene Anwendungsszenarien und hierin auftretende Probleme werden aufgezeigt und Lösungsansätze durch die Einführung neuer Technologien, in diesem Fall der Webservices sowie multiskaliger Datensätze, ermittelt. Hieraus leitet sich die in Abschnitt 4.2 aufgezeigte Systemarchitektur ab. In den folgenden Kapiteln soll anschließend das hier entworfene System schrittweise exemplarisch implementiert werden.

4.1 Szenarienbasierte Systementwicklung

Zu Beginn der Entwicklung eines neuartigen Systems helfen Szenarien bei der Ermittlung der Probleme und Bedürfnisse und erleichtern so den Einstieg in die Ableitung von Lösungen. Szenarienbasiertes Design umfasst Techniken, in denen der Nutzen zukünftiger Systeme in einem frühen Status des Entwicklungsprozesses konkret beschrieben wird (Rosson & Carroll 2002). Szenarien ermöglichen zu jedem Zeitpunkt der Entwicklungen die Bedürfnisse und Probleme im System aufzudecken. Szenarien sind flexibel, sie können in unterschiedlichen Abstraktionsebenen formuliert werden. Sie können verglichen werden mit Prototypen, konkreten Designvorschlägen also, welche bewertet und diskutiert werden können. Sie enthalten oftmals implizite Informationen über die Konsequenzen eines bestimmten Designs. Des Weiteren sind die szenarien-basierten Entwicklungen nah an dem zukünftigen Anwender. Sie werden mit Begriffen formuliert, die den Tätigkeiten der Nutzer entsprechen. Auf Grund ihrer Flexibilität können diese in den verschiedenen Stadien der Softwareentwicklung und auch in unterschiedlichen Abstraktionsebenen eingesetzt und beschrieben werden.

Szenarien können am besten definiert werden anhand eines Kontinuums von Beschreibungen der realen Welt bis hin zu Modellen und Spezifikationen. An dem einen Ende dieser Dimension stellen Szenarien Beispiele aus der realen Welt dar, ausgedrückt durch eine natürliche Sprache, Bilder oder andere Medien. An dem anderen Ende befinden sich Modelle wie „use cases“ und andere Ablaufbeschreibungen (Sutcliffe 2003). Somit können Szenarien verschiedene Rollen einnehmen. Sutcliffe (2003) fasst diese in drei Kategorien zusammen:

- Eine Geschichte oder ein Beispiel als grundlegende Erzählung, basierend auf Erfahrungen der realen Welt. Diese Geschichten entsprechen dem allgemeinen, umgangssprachlichen Verständnis des Begriffes Szenario
- Eine zukünftige Vision eines entworfenen Systems mit Sequenzen des Verhaltens und möglicher Beschreibung der Zusammenhänge
- Ein einfacher Pfad durch ein Modell (use case) im Rahmen objektorientierter Softwareentwicklung (z.B. in der UML²)

Dieser Unterteilung folgend sollen für das hier beschriebene System eines Webservices nachstehend die Entwicklungsebenen durch entsprechende Szenarien reflektiert und diskutiert werden.

²Unified Modeling Language

4.1.1 Bedarfsanalyse

Das grundlegende Konzept sollte zunächst die Schlüsselaspekte einer Applikation beinhalten.

<i>Komponente</i>	<i>Grundkonzept</i>
Oberste Vision	Nutzer haben überall Zugriff auf alle verfügbaren, integrierten Informationen und Repräsentationen zu einem Phänomen der realen Welt
Grundlegende Argumentation	MRDB ermöglicht die Integration und somit Zugriff auf alle Informationen und Repräsentationen eines Phänomens Ein Zugriff auf die Informationen über Webservices ist von jeder Plattform aus möglich
Anwender: <i>Serviceanbieter</i> <i>Mobiler Nutzer</i> <i>Nicht-mobiler Nutzer</i>	Für jede Anwendergruppe existiert ein entsprechend zugeschnittener Service Zugriff auf alle verfügbaren Information über einen Service Hat dieselben Möglichkeiten, wie der mobile Nutzer, wird aber von diesem unterschieden
Grundlegende Annahmen	Nutzer hat Zugriff auf ein mit dem Internet verbundenes Gerät Alle relevanten Daten sind in einer MRDB integriert Es existiert für jeden Anwendungsfall ein Service, welcher über den MRDB-Service auf die Daten zugreift

Tab. 4.1: Grundlegendes Konzept zur Entwicklung eines auf einer MRDB beruhenden webbasierten Systems

Die Tabelle 4.1 beschreibt das Grundkonzept des MRDB-Webservices und ist gleichzeitig Basis für die im Folgenden beschriebenen Szenarien. Die primäre Vision entspricht der weiter oben aufgeführten Argumentation, dass der MRDB-Service eine höhere Ebene der Informationsabfrage darstellt. Dies wird realisiert durch die Kombination von Webdiensten, welche den Zugriff auf die Informationen von jeder Plattform aus ermöglichen, mit den in einer MRDB integrierten Daten. Die Liste der potenziell involvierten Parteien beschreibt Personen oder Institutionen, welche durch das System in irgendeiner Form beeinflusst werden. In diesem Fall kann der mobile, aber auch der nicht-mobile Nutzer auf einen Service zurückgreifen. Dieser Webdienst stellt bestimmte Funktionalitäten für seine Zielgruppe bereit und greift seinerseits auf den MRDB-Webservice zurück. In dem System existieren schließlich neben den rein funktionellen Belangen auch nicht-funktionelle Bedingungen, welche hier als anfängliche Annahme dokumentiert sind. Hierzu gehört, dass die Anwender ein (mobiles) Gerät besitzen, welches einen Internetzugang ermöglicht. Weiterhin müssen die für den Nutzer relevanten Informationen in einer MRDB-Struktur vorliegen und es müssen Services existieren, welche auf den hier vorgestellten MRDB-Webservice zurückgreifen.

Szenarienbasiertes Design setzt zu Beginn auf eine Bedarfsanalyse, aus der Problemszenarien und Anforderungen abgeleitet werden sollen. Sutcliffe (2003) nennt folgende Schritte einer Bedarfsanalyse:

- Beschreibung des unbefriedigenden Ist-Zustandes, den das neue System verbessern soll
- Vision, wie das neue System operieren sollte
- Beschreibung des Verhaltens der Nutzer und des Systems

Somit soll nachfolgend eine Bedarfsanalyse anhand von zwei Anwendungsszenarien durchgeführt werden. Wir begleiten *Karl*, einen Feuerwehrmann, bei seinem Einsatz und danach begibt sich *Christa* auf Wohnungssuche in einer ihr fremden Stadt.

Szenario 1a: Feuerwehr

Karl, 40 Jahre alt und Hauptbrandmeister, ist für den heutigen Abend als Schichtführer bei der Berufsfeuerwehr Hannover eingeteilt. Er war gerade dabei, die Protokolle des letzten Einsatzes in das System des Zentralrechners einzutragen, als plötzlich der Alarm ertönt. In der vor einer Minute noch ruhigen Leitstelle herrscht plötzlich hektisches Treiben. Karl informiert sich bei dem Leitstellenführer und Disponenten über den Ort des Einsatzes. Ein Passant hatte eine brennende Buschgruppe im 'Georgengarten' gemeldet, nähere Informationen liegen nicht vor. Karl sucht sich schnell den für diesen Bereich relevanten Alarmordner und begibt sich zum Einsatzfahrzeug. Nachdem er den Fahrer über das Einsatzziel informiert hat, fährt dieser in eiligem Tempo zum Zielort.

Da das Einsatzziel in einem Neubaugebiet liegt, welches noch nicht in dem Stadtplan verzeichnet ist, sucht Karl den entsprechenden Auszug aus der Katasterkarte in dem Alarmordner und navigiert den Fahrer auf den letzten Kilometern zum Zielort.

Vorher sucht Karl die relevanten Unterlagen, wie Übersichtskarten oder die Lage der Wasserentnahmestellen, aus dem Ordner zusammen. Von weitem schon ist das Feuer zu erkennen, viel größer als erwartet. Am Einsatzort angelangt sehen die Einsatzkräfte, dass das Feuer auf das benachbarte Gebäude übergegriffen hat. Karl sucht rasch die zu dem Gebäude verfügbaren Informationen aus seinem Ordner. Es handelt sich um ein Gebäude der Universität Hannover, dem Institut für Chemie. Er ruft die Leitstelle an, beschreibt die Lage des Gebäudes und fragt, ob sich in dem Gebäude gefährliche Güter befinden. In seinem Stadtplan und auch in seinem Ordner sind keine Informationen hierzu verzeichnet. In der Zwischenzeit suchen die anderen Einsatzkräfte die nächstgelegenen Wasserstellen. Von der Leitstelle erfährt Karl nach einiger Wartezeit, dass sich hochexplosive Chemikalien im Chemiesaal A des Gebäudes befinden. Er durchsucht alle Gebäudepläne und findet schließlich den Chemiesaal A im 2. Stock. Das Feuer ist zum Glück noch nicht bis dorthin vorgedrungen, so dass die Truppe den Löschangriff gefahrlos durchführen kann. Karl hatte schon darüber nachgedacht, wieviele Personen wohl in dem Stadtteil leben und eventuell zu evakuieren seien. Nach erfolgreichem Löscheinsatz schreibt Karl auf der Rückfahrt das Protokoll über den Einsatz.



Abb. 4.1: Aktuelles Szenario der Feuerwehr: Informationen sind in einem Alarmordner zusammengestellt (Quelle: <http://www.feuerwehr.tum.de>)

Dieses Szenario beschreibt einen typischen Rettungseinsatz (vgl. Abbildung 4.1), wobei hier der Fokus auf der Art und Weise der Informationsbeschaffung liegt. Entsprechenden Probleme und Bedürfnisse sind in Tabelle 4.2 zusammengestellt.

Aus diesen Problemen und Einschränkungen lässt sich nun folgendes, visionäres Szenario ableiten:

Visionäres Szenario 1b: Feuerwehr der Zukunft

Karl, 40 Jahre alt und Hauptbrandmeister, ist für den heutigen Abend als Schichtführer bei der Berufsfeuerwehr Hannover eingeteilt. Er war gerade dabei, sich in einer Zeitschrift über die neuesten Technologietrends der Feuerwehr zu informieren, als plötzlich der Einsatzalarm ertönt. In der vor einer Minute noch ruhigen Leitstelle herrscht plötzlich hektisches Treiben. Auf seinem PDA sieht Karl, dass im Georgengarten von einem Passanten eine brennende Buschgruppe gemeldet wurde. Der Fahrer des Einsatzfahrzeugs kann auf seinem PDA ebenfalls den Einsatzort erkennen, gleichzeitig wurde die optimale Route an diesen übermittelt. Auf dem Weg zum Einsatzort lässt sich Karl zunächst eine Übersichtskarte anzeigen, um sich einen Überblick über das Einsatzgebiet, der Siedlungsstruktur und den Zufahrtswegen zu verschaffen. Plötzlich erscheint eine Meldung auf dem Display, dass auch die Rauchmelder in dem Gebäude „Am Schneiderberg 1“ Alarm ausgelöst haben. Auf seiner Übersichtskarte erkennt Karl, dass dieses Gebäude direkt an den zuerst gemeldeten Brand angrenzt. Er wählt einen größeren Kartenmaßstab, um eine detaillierte Darstellung des Gebäudes zu erhalten. Dann ruft er die zu dem Gebäude gehörigen Informationen ab.

Er erkennt, dass es sich um ein Gebäude der Universität Hannover handelt, dem Institut für Chemie. Gleichzeitig erhält er die Information, dass hochexplosive Chemikalien im Chemiesaal A gelagert sind. Der Gebäudeplan zeigt ihm die Lage aller gefährlichen Güter automatisch an. Außerdem werden in der Karte die dem Gebäude zugeordneten Wasserstellen visualisiert. Karl informiert die Kollegen

<i>Status quo</i>	<i>Bewertung</i>	<i>Bedürfnisse</i>
Informationen des mobilen Einsatzleiters in einem Alarmordner	Unhandlich	Abfrage der Informationen digital über ein kompaktes Gerät, einfache Abfrage der verfügbaren Informationen direkt über das Kartenobjekt
	Informationen sind schwer zu finden	Zusammenstellung der relevanten Informationen durch einen Service, welcher die verfügbare Datenbasis kennt; Abfrage bestimmter oder aller zu einem Objekt gehörigen Informationen (z.B. Gebäudeplan, Einwohnerzahl) z.B. per Klick auf das Kartenobjekt („Information Drilling“)
	Informationen sind möglicherweise veraltet	Abfrage der aktuellen Informationen direkt aus der Datenbank
	Informationen werden doppelt geführt und müssen an zwei Stellen geändert werden	Nutzung und Pflege der Informationen über ein zentrales, von außen zugängliches System (z.B. über Webdienste)
	Keine übersichtliche Darstellung der relevanten Informationen (geografische sowie attributale Informationen)	Neue Möglichkeiten zur Visualisierung geografischer Informationen (Übersichts- und Detailkarten, multiskalige Karten) sowie zusätzliche Darstellung von Attributen und Listen (z.B. Einwohnerzahlen von Stadtteilen oder verantwortliche Personen von Gebäuden)
	Keine Analysefunktionalitäten	Bereitstellung zusätzlicher Funktionalitäten (z.B. Bestimmung gefährdeter Einwohner) in der Zugriffsschnittstelle (z.B. in dem Webdienst)
Informationen der Leitstelle	Alle relevanten Daten sind auf dem Zentralrechner der Feuerwehroleitstelle gespeichert [+]	Keine
	Geografische Daten und Informationsdaten der Feuerwehr sind nicht integriert	Notwendigkeit der Integration aller relevanten Daten zur Sicherung der Konsistenz und Verfügbarkeit
	Nur die Leitstelle hat Zugriff auf alle aktuellen, digitalen Daten	Öffnung des Systems zum Zugriff durch mobile Einsatzkräfte über Webservice-Schnittstellen
	Einsatzleiter vor Ort hat nur durch mündliche Kommunikation mit der Leitstelle Zugriff auf die relevanten Informationen	Einsatzleiter vor Ort und Leitstelle benötigen Zugriff auf dieselbe Informationsbasis

Tab. 4.2: Probleme und Bedürfnisse abgeleitet aus dem Szenario 1a (Feuerwehr)

hierüber. Da der Kontakt dieser Chemikalie mit Feuer eine verheerende Explosion auslösen kann, wie ihn die digitale Version des „Handbuchs der gefährlichen Güter“ informiert, denkt Karl über eine Evakuierung nach, hält hierzu jedoch Rücksprache mit der Leitstelle. Der Leitstellenleiter, der die gleichen Informationen auf seinem Bildschirm hat, erklärt Karl, dass die Gebäude in diesem Stadtteil ausnahmslos Universitätsgebäude sind, welche nachts nicht bewohnt sind. Die Leitstelle hat per Telefon schon alle Verantwortlichen der Gebäude in dem Stadtteil informiert. Am Einsatzort angekommen wissen die Feuerwehrmänner sofort, wo die Wasserstellen zu finden sind. Karl verschafft sich einen Überblick über die Lage und schickt seine Männer durch den Nordeingang in das Gebäude. Das Feuer ist weit entfernt von dem Chemiesaal A. Nach dem erfolgreichen Löscheinsatz trägt Karl die Protokolldaten auf seinem PDA ein und überträgt diese auf den Server in der Leitstelle. Gleichzeitig vermerkt er in der Karte, dass eine Wasserentnahmestelle defekt ist.

Diese Vision zeigt, wie der Informationsfluss idealerweise erfolgen kann. Alle relevanten Daten sind in dem DBMS zusammengefasst und integriert, so dass direkt alle zu einem Objekt gehörigen Informationen abgerufen werden können. Der direkte Zugriff der mobilen Einsatztruppe auf diese Daten sichert die Versorgung mit aktuellen Informationen. Außerdem können die Leitstelle und der mobile Einsatzleiter auf dieselben Daten zurückgreifen und ihre Entscheidungen treffen.

Ein zweites Szenario, in welchem, ähnlich wie in dem vorangegangenen, ein mobiler Nutzer ortsbezogene Daten und Informationen benötigt, soll diese Bedürfnisse und Probleme durch eine andere Sichtweise verdeutlichen.

Szenario 2a: Wohnungssuche

Christa, 19 Jahre alt und angehende Studentin, sucht eine Wohnung in ihrem neuen Studienort Hannover. Bei der Suche möchte sie gleichzeitig ihren neuen mit GPS ausgestatteten PDA testen. Sie wählt an ihrem PC die Stadtplandaten für Hannover aus und möchte diese auf ihren PDA übertragen. Das System vermeldet jedoch, dass nicht genügend Speicherplatz vorhanden ist. Somit entscheidet sie sich für ein kleineres Gebiet der digitalen Version von Hannover, da sie ja zur Not auch noch den manuellen Stadtplan zur Hand hat.

Nachdem sie am Hauptbahnhof angelangt ist, möchte sie zunächst in der Jugendherberge einchecken. Sie hat zu Hause schon deren Position in dem Kartenausschnitt auf dem PDA markiert. Unglücklicherweise ist das Display ihres PDA zu klein, um einen Kartenausschnitt mit der Jugendherberge und dem Hauptbahnhof anzuzeigen. Sie verkleinert deshalb das Kartenbild, bis beide Orte zu sehen sind. Da die Darstellung nun zu klein geworden ist, um die Straßennamen lesen zu können oder sonstige Orientierungshilfen zu bekommen, muss Christa sich auf die Bewegungsrichtung konzentrieren, die ihr das GPS liefert. An einer unübersichtlichen Kreuzung wählt sie daher zunächst die falsche Straße, merkt jedoch nach 100 Metern, dass ihre Position von der Route abweicht.

Am Nachmittag will Christa mehrere Wohnungen besichtigen, die sie in der Zeitung gefunden hat. Um einen Überblick über die Lage der Wohnungen zu erhalten, breitet sie ihre Papierkarte aus und markiert die Lage der Wohnungen. Eine der drei Wohnungen kann Christa nicht besichtigen, da sie die angegebene Straße nicht auf dem Stadtplan finden kann. Bei der Suche der ersten Wohnung muss Christa leider feststellen, dass die Gebäude dieser Straße allesamt keine Hausnummer tragen. In ihrem Stadtplan und auch in der Karte auf dem PDA finden sich zu allem Unglück auch nur generalisierte Gebäudeblöcke und Hausnummernbereiche, so dass Christa viele Klingelschilder durchsuchen muss, um die Wohnung zu finden. Alle Wohnungen sind sehr schön und natürlich haben die Vermieter nur positive Eigenschaften zu vermelden. Um eine objektive Auskunft zu erhalten, begibt sich Christa zum Rathaus, um hier Informationen zu den einzelnen Stadtteilen und Wohngegenden zu erfragen.

Tabelle 4.3 fasst die erkennbaren Probleme und Bedürfnisse dieses Szenarios zusammen.

<i>Status quo</i>	<i>Bewertung</i>	<i>Bedürfnisse</i>
Abfrage und Präsentation der Geoinformationen (Karten)	Verfügbarkeit der Daten nicht spontan (Vorbereitung notwendig)	Online-Zugriff auf die Daten notwendig
	Suche und Auswahl der relevanten Daten (Kartenausschnitt, Informationen etc.) manuell	Webservice muss die verfügbaren Daten kennen (GetCapabilities), wählt die optimalen Daten, generiert eine Karte und passt diese automatisch an die aktuellen Bedürfnisse an
	Papierkarte ist übersichtlich und bietet alle Informationen auf einen Blick [+] Repräsentation auf dem mobilen Gerät unübersichtlich (Maßstab, kleines Display)	Digitale Karte sollte die positiven Eigenschaften einer Papierkarte übernehmen (z.B. Übersichtlichkeit) Neue Formen der Visualisierung von Geoinformationen auf mobilen Geräten notwendig (z.B. multiskalige Karten, Lupeneffekt etc.)
Abruf relevanter Informationen	Alle vorhandenen Informationen sind separiert, nicht integriert	Bündelung aller Informationen in einem konsistenten, integrierten System
	Keine Zusatzinformationen verfügbar	Möglichkeit zur Abfrage aller zu einem Objekt verfügbaren Informationen notwendig („Information Drilling“)

Tab. 4.3: Probleme und Bedürfnisse abgeleitet aus dem Szenario 2a (Wohnungssuche)

Diese führen zu nachstehendem, visionären Alternativ-Szenario.

Visionäres Szenario 2b: Wohnungssuche der Zukunft

Christa, 19 Jahre und angehende Studentin, sucht eine Wohnung in ihrem neuen Studienort, Hannover. Bei der Suche möchte sie gleichzeitig ihren neuen mit GPS ausgestatteten PDA testen. Nachdem sie am Hauptbahnhof angekommen ist, möchte sie zunächst in der Jugendherberge einchecken. Sie ruft hierzu einen Kartenwebdienst über ihren PDA auf. Per Adresssuche lokalisiert sie die Jugendherberge. Der Service ruft die notwendigen Geodaten über einen weiteren Webservice ab, berechnet die kürzeste Route und zeigt diese in einer Übersichtskarte an. An einer unübersichtlichen Kreuzung weiß Christa kurzfristig nicht, welcher Weg nun der richtige ist. Bei einem Blick auf die Karte, welche ihr alle Details der Kreuzung in einer lupenartigen Vergrößerung zeigt, wird ihr schnell klar, dass sie das L-förmige Gebäude, welches ein Restaurant beherbergt, passieren muss.

Am Nachmittag möchte Christa mehrere Wohnungen besichtigen, die sie in der Zeitung gefunden hat. Wieder ruft sie den Service auf, lässt sich die Lage der Wohnungen sowie ihre eigene Position in einer Übersichtskarte anzeigen und stellt fest, dass eine der Wohnungen nicht weit von der Jugendherberge entfernt liegt. Der Routing-service berechnet wieder die kürzeste Strecke. Um die Wohnung schnell zu finden, lässt sie sich die Gebäude im Umkreis von 50 Metern um ihre Position mit höchster Auflösung zeigen. Die übrige Karte besitzt einen kleineren Maßstab, so dass sie trotzdem immer noch die Entfernung zum Ziel und die Lage der Wohnung einschätzen kann. Der Service zeigt ihr am Ende der Route das Haus mit der richtigen Hausnummer mit der höchsten verfügbaren Auflösung. Das Haus mit den zwei kleinen Vorsprüngen muss es sein.

Die Wohnung ist sehr schön und natürlich hat der Vermieter nur positive Eigenschaften zu vermelden. Sie lässt sich daher alle verfügbaren Informationen anzeigen, die mit dem Gebäude verknüpft sind und erfährt, dass dieser Stadtteil bei Studenten sehr beliebt ist auf Grund seiner preiswerten Mieten, der Nähe zur Universität, dem Stadtpark und dem Maschsee sowie seiner vielfältigen Kneipen- und Barszene. So hat Christa ihren Favoriten schon gefunden.

Was bedeuten die aufgezeigten Probleme und angedachten Lösungsmöglichkeiten nun konkret für die Implementierung? Die Tätigkeiten und Anforderungen lassen sich wie folgt zusammenfassen: Die kartografische Visualisierung sollte den aktuellen Bedürfnissen angepasst werden und alle einem bestimmten Realweltobjekt anhängigen Informationen sollten abgefragt werden können. Weiterhin sollte die mobile Verfügbarkeit dieser Funktionalitäten und die Aktualität der Daten sichergestellt sein.

Durch den Einsatz neuer Technologien sollen die Anforderungen der Nutzer so weit wie möglich erfüllt und deren Aktivitäten erleichtert werden. Die in den Szenarien beschriebenen Aktivitäten können durch den Einsatz neuer Methoden und Technologien, wie sie in Abbildung 4.2 aufgezeigt werden, schrittweise verändert und verbessert werden. Der Nutzer möchte auf raumbezogene Daten zugreifen und aus den gesammelten Informationen

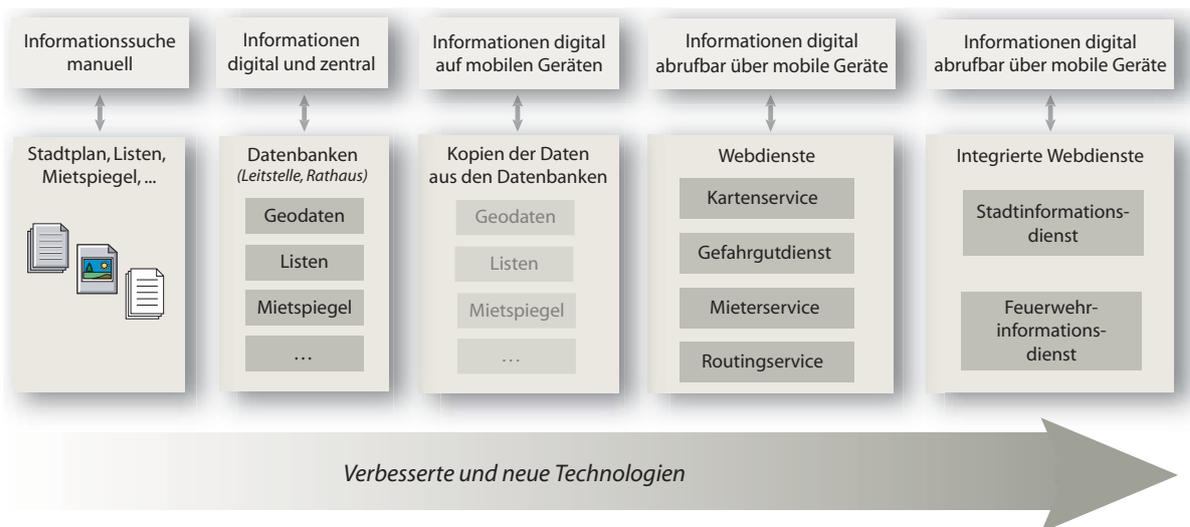


Abb. 4.2: Aktivitäten-Beschreibung beeinflusst durch neue Informationstechnologien

bestimmte Schlüsse ziehen. Ein Stadtplan gibt Aufschluss über die Lage und Umgebung bestimmter Orte, in der Regel dem eigenen Standpunkt, einem bestimmten Zielpunkt oder der Route von A nach B. Zusätzliche Informationen zu bestimmten Objekten des Raumes werden entweder aus Listen zusammengesucht oder aber sie

sind an zentraler Stelle digital gespeichert. Eine Aktivität besteht somit darin, die Stellen aufzusuchen, an denen diese Informationen vorgehalten werden, in den Szenarien also die Leitstelle oder das Rathaus. Abbildung 4.2 beschreibt die Möglichkeiten, diese Tätigkeiten durch Einsatz alternativer oder neuer Technologien zu verbessern. So ermöglichen Webservices und integrierte Datensätze einer MRDB einen bequemen und effizienten Zugriff auf die relevanten Informationen.

Erst die letzte, am rechten Rand der Abbildung dargestellte Konstellation erfüllt und verbessert die ursprünglichen Anforderungen des Nutzers, der seine Informationen mühsam sammeln musste, in optimaler Weise. Die zuvor größtenteils manuell durchgeführte Informationssuche kann nun bequem vom jeweiligen Standpunkt aus erfolgen, wenn der Anwender mit seinem mobilen Geräte auf einen Webservice zugreift, welcher seinerseits auf in einer MRDB zusammengefassten Informationen basiert.

Bis hierher wurden die Probleme und Anforderungen an das System definiert und eine technologische Strategie zur Lösung der Probleme und somit Verbesserung der Situation skizziert. Nachfolgend müssen Überlegungen zur praktischen Realisierung getroffen werden. Daher wird zunächst eine Systemarchitektur mit ihren Komponenten und Schnittstellen benötigt. Außerdem müssen die Anforderungen und Aufgaben der einzelnen Elemente klar definiert sein, bevor diese implementiert werden können.

4.2 Systemarchitektur

Aus den Untersuchungen des vorangegangenen Abschnittes leitet sich ein System ab, welches es einem mobilen Nutzer ermöglicht, auf einen spezifischen Webservice zuzugreifen, welcher seinerseits durch Daten einer MRDB gespeist wird.

In Abschnitt 2.2 wird eine Systemarchitektur für Webservices vorgestellt, welche aus der Datenbank-, Logik- sowie Präsentationsebene besteht. Nach dem Client-Server Prinzip bietet jede Ebene, mit Ausnahme der Präsentations- oder Anwenderebene, einen Service, auf die ein Client aus einer anderen oder derselben Ebene zugreifen kann. Die einzelnen Dienste und ihre Funktionalitäten werden in Form von Spezifikationen beschrieben, so dass ein Aufruf dieser problemlos möglich ist, sofern die Spezifikationen, welche quasi als Schnittstelle zu den einzelnen Diensten dienen, beachtet werden.

Dieser Beschreibung folgend werden in Unterabschnitt 2.3.2 die Geo-Webservices in vier Servicebereiche unterteilt. So ergeben sich Informationsmanagement-, Prozessierungs- und Applikationsdienste sowie Nutzeranwendungen (vgl. Abbildung 2.5, S.28). Die letztgenannten agieren dabei lediglich als Client, nicht jedoch als Server, so dass drei Dienstebenen gezählt werden können.

An diesen Aufbau ist auch das hier beschriebene System angelehnt, wobei die oben beschriebene Bedarfsanalyse eine Notwendigkeit von Prozessierungsdiensten ausschließt. Daher ergeben sich die drei in der Abbildung 4.3 dargestellten Servicebereiche:

1. Client-Services, zur Formulierung von Abfragen und Darstellung der Ergebnisse
2. Applikations-Services, welche spezielle und allgemeine Funktionalitäten für die Nutzergruppen bereitstellen; in dem beschriebenen Prototypen sollen diese exemplarisch implementiert werden
3. Informationsmanagement-Services; diese ermöglichen den Zugriff auf die Datenbank respektive die MRDB unter Ausnutzung der Vorteile, die eine solche dem Nutzer zur Verfügung stellt

Somit agieren drei beziehungsweise vier Komponenten (der Nutzer selbst stellt auch eine agierende Komponente dar, welche jedoch aus technischer Sicht nicht als solche gewertet wird) unterteilen, welche miteinander interagieren und jeweils bestimmte Anforderungen erfüllen müssen. Die Applikation auf dem (mobilen) Gerät muss sich den Anforderungen des Nutzers entsprechend verhalten. Diese Nutzeranforderungen sind in dem vorangegangenen Abschnitt zu finden. Die Anwendung gibt die Anforderungen teilweise an den jeweiligen Webservice weiter, welcher über die Service-Schnittstelle bestimmte Funktionalitäten bereitstellen sollte. Der Webdienst wiederum erwartet zur Erfüllung seiner Anforderungen bestimmte Funktionalitäten von dem Datenservice, wozu neben den Abfragemöglichkeiten natürlich auch der Datenbankinhalt zu zählen ist.

Die einzelnen Komponenten des Systems, ihre Anforderungen sowie deren Implementierung sollen im Folgenden jeweils separat beschrieben werden. Dabei bleibt festzuhalten, dass es sich anbietet, das System in einzelne Komponenten zu zerlegen, welche jedoch in einem Zusammenhang stehen. So greift der Endnutzer zum Beispiel in der Regel nicht direkt auf den Dateninformationsservice zurück, dessen Funktionalitäten werden jedoch indirekt durch die Anforderungen des Anwenders beeinflusst. Die Ausrichtung der einzelnen Dienste wird dabei immer spezieller, je näher man dem Nutzer kommt.

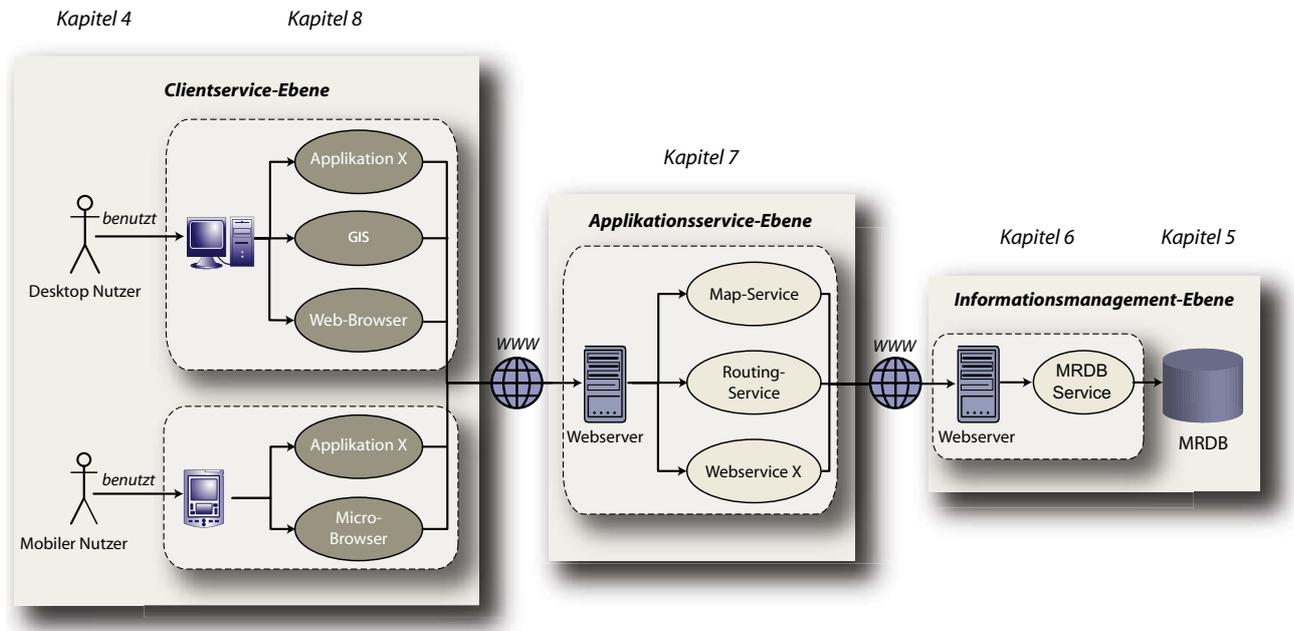


Abb. 4.3: Systemarchitektur zur Nutzung integrierter Daten einer MRDB innerhalb einer Webumgebung

Die beschriebene Implementierung eines Prototypen stellt nicht in jeder Ebene einen vollständig ausgeprägten Service bereit, sondern erreicht jeweils den Status eines „Proof of Concept“. Die Beschreibung des Systems beginnt bei dem Aufbau einer multiskaligen Datenbasis (MRDB) (Kapitel 5). Anschließend folgt die Implementierung eines Informationsmanagement-Dienstes (MRDB-Service), also der Ebene, welche die Inhalte einer MRDB über eine Serviceschnittstelle nach außen verfügbar macht (Kapitel 6). Dieser sollte von allen Applikationsdiensten verwendbar und somit so allgemein wie möglich konzipiert sein. Hiernach folgt eine Beschreibung der Applikationsdienste (Kapitel 7), welche mehrere Anwendungsfälle abdecken sollten, auf der anderen Seite jedoch eine von vielen Anwendungsmöglichkeiten des MRDB-Services darstellen und somit spezifischer formuliert sind als dieser. Die Applikation schließlich, welche auf dem mobilen Gerät arbeitet, sollte auf den speziellen Nutzer und Anwendungsfall zugeschnitten sein (Kapitel 8).

5 MRDB-Webdienst: Aufbau einer multiskaligen Datenbasis

Das folgende Kapitel beschreibt einen Weg zur Realisierung einer multiskaligen Datenbank mit der Intention, diese als Grundlage für einen Geodaten-Webservice einzusetzen. Die Konzeption der multiskaligen Datenbank (Abschnitt 5.2) wird hierzu in mehrere Schritte unterteilt, beginnend mit der Festlegung der konzeptionellen Auflösungsebenen, darauf folgend die Informationsbedarfsanalyse, der konzeptionelle Entwurf und schließlich, aufbauend auf dem gewählten DBMS, der logische Entwurf, welcher das Datenbankschema festsetzt.

Anschließend gilt es, Datenlücken in den Auflösungsstufen durch geeignete Generalisierungsschritte zu schließen (Abschnitt 5.3), um in einem letzten Schritt die bestehenden Testdaten, welche in dem Unterabschnitt 5.2.2 beschrieben werden, sowie die durch Generalisierung erzeugten LoDs in das DBMS zu integrieren. Hierzu müssen nicht nur die entsprechenden Daten in die Datenbank eingepflegt, sondern zusätzlich die Beziehungen zwischen den unterschiedlichen Auflösungsstufen bestimmt und gespeichert werden (Unterabschnitt 5.3.3).

5.1 Anforderungen und Charakteristika eines DBMS

Die primäre Funktion eines DBS¹, welches aus der Verwaltungssoftware, dem DBMS und der Menge der zu verwaltenden Daten besteht, liegt in der Speicherung und Verwaltung von Daten. Diese Funktionalität kann jedoch alternativ auch von einem filebasierten System übernommen werden, welches die Informationen in einzelnen Dateien und Ordnern verwaltet. Das filebasierte System besitzt jedoch diverse Nachteile, welche durch den Einsatz eines DBS kompensiert werden. Datenredundanz entsteht dadurch, dass die Anwendungssoftware verschiedener Nutzer, welche zwar ähnliche aber doch unterschiedliche Bedürfnisse an die Daten haben, jeweils eigene Dateien erzeugt und verwaltet. So benötigt die Verwaltung einer Universität eine Datei mit den Namen, Adressen, Studienfächern und Matrikelnummern der Studenten, der Dozent dagegen verwaltet eine Liste mit den Namen, Matrikelnummern und Prüfungsleistungen seiner Studenten. Die Daten werden somit teilweise redundant erzeugt und verwaltet. Weiterhin müssen Anwendungsprogrammierer die Struktur der Dateien kennen. Diese Lösung ist daher datenabhängig. Des Weiteren können verschiedene Anwender nicht gleichzeitig auf die in den Files abgelegten Daten zugreifen und damit arbeiten. Es fehlt die Zugriffskontrolle und Datensicherheit. Diese und weitere Probleme werden durch die Verwendung einer zentralen Speichereinheit in Form eines DBS gelöst. Codd (1982) fasst die Eigenschaften eines DBMS in folgenden Punkten zusammen:

- *Integration*: Einheitliche Verwaltung aller verfügbaren Daten in einem zentralen System
- *Datenmanipulation*: Mit Hilfe von Datenmanipulationsoperatoren werden Daten eingefügt, verändert oder gelöscht
- *Konsistenzüberwachung*: Transaktionen ziehen am Ende immer eine Prüfung der Konsistenz nach sich, also die Kontrolle auf widerspruchsfreie Inhalte, auch als Integritätssicherung bezeichnet
- *Katalog*: Ein für den Anwender verfügbarer Katalog zur Datenbeschreibung wird bereitgestellt
- *Datenunabhängigkeit*: Zugreifende Anwendungsprogramme werden geschützt vor Veränderungen in den Datenstrukturen, physikalischen Speichermedien etc.
- *Nutzerschnittstellen*: Es sind Schnittstellen vorhanden für den Datenbankadministrator, den Programmierer, welcher direkt auf die Datenbank zugreift sowie den Anwender, welcher die Daten direkt in seiner Applikation verwendet oder diese manipuliert
- *Datenschutz und -sicherheit*: Daten in der Datenbank werden vor unautorisierten und fehlerhaften Zugriffen geschützt und stellen somit immer einen konsistenten Zustand der Inhalte sicher; gleichzeitig sind die Daten etwa nach Systemfehlern wiederherstellbar
- *Synchronisation*: Konkurrierende Transaktionen werden synchronisiert

Die nachfolgend beschriebene MRDB soll daher auf einem derartigen DBMS aufgebaut werden.

¹Datenbanksystem

5.2 Datenbankkonzept einer MRDB

Die Gründe zur Verwendung einer MRDB sowie deren Vorteile und Nutzungsmöglichkeiten werden ausführlich in Abschnitt 3.1 dargelegt. Vor der Implementierung sollten hier trotzdem noch einmal die Gründe und Nutzungsmöglichkeiten einer MRDB aufgelistet werden, wobei speziell diejenigen zur Sprache kommen sollen, welche Einfluss auf die Implementierung haben, denn das Datenschema und andere Modellentscheidungen sowie später auch die Funktionen der Schnittstelle hängen in gewissem Maße von der intendierten Verwendung ab.

Eine MRDB fasst unterschiedliche Repräsentationen eines bestimmten Phänomens in einem integrierten System zusammen. Dabei gibt es diverse Gründe, warum verschiedene Repräsentationen erfasst wurden und nebeneinander existieren. In dem hier beschriebenen System soll die Multiplizität sich im Sinne von Multiskaligkeit darstellen. Unterschiedliche geometrische Repräsentationen bestimmter Objekte der Realität sollen gespeichert werden. Somit können beispielsweise verschiedene Maßstäbe bedient werden, ohne dass Generalisierungsfunktionalitäten in dem System vorhanden sein müssen. Darüber hinaus dienen alternative Auflösungsstufen eines Objektes auch dem Informationsgewinn. So werden, um eine übersichtliche Darstellung der Situation zu gewinnen, Daten geringer Auflösung genutzt, während für Detailinformationen eine höhere Auflösung benötigt wird. Der Anwender oder die von ihm verwendete Applikation sollte die Möglichkeit erhalten, aus einer Vielzahl von Repräsentationen unterschiedlich hoher Auflösung diejenige mit dem relevanten Informationsgehalt abfragen zu können; bei Bedarf auch alle verfügbaren Daten.

5.2.1 Festlegung der konzeptionellen Auflösungsebenen

Das System soll für jedes topografische Objekt Geometrien unterschiedlicher Auflösung bereitstellen, so dass diese zu einer Karte eines bestimmten Maßstabes und auch für einen bestimmten Gerätetyp kombiniert werden können. In Abschnitt 3.1 wird der Begriff „Auflösung“ näher spezifiziert, wobei die geografische Auflösung als minimale Größe eines Objektes oder Objektteiles definiert wird, welches in der Datenbank gespeichert ist, während die semantische Auflösung den Grad semantischer Details in den Daten bestimmt.

Die verwendete geografische Auflösung ist dabei nicht nur von dem jeweils gezeigten Maßstab, sondern bei digitalen Karten auch von dem eingesetzten Gerät und dessen Display abhängig. Gerade mobile Geräte haben auf Grund ihren kleinen Bildschirme und damit verbundenen begrenzten Auflösung (vgl. Unterabschnitt 2.4.1) Probleme, die gleiche Menge an geografischen Informationen zu übermitteln wie Papierkarten. Stehen alternative Geometrien mit geringerer Auflösung zur Verfügung, so können diese genutzt werden, um die Informationsmenge zu reduzieren, um somit den kleineren Displays gerecht zu werden.

Um nun Daten unterschiedlichster Auflösungsstufen generieren zu können, ist ein Generalisierungsprozess notwendig. Da jedoch zum einen keine operationelle Lösung verfügbar ist, um den gesamten Generalisierungsprozess zu automatisieren und zum anderen existierende Teilverfahren zeit- und rechenintensiv sind und darüber hinaus in vielen Fällen manuelle Eingriffe bedingen, so bietet es sich an, Daten im voraus zu generalisieren.

Hier stellt sich nun die Frage: Wie viele Repräsentationen oder LoDs sind notwendig, um alle Maßstabsbereiche lückenlos abdecken zu können und in welcher geografischen Auflösung müssen diese verfügbar sein? Anders formuliert: Bis zu welchem Maßstab kann eine bestimmte Repräsentation verwendet werden, bevor eine erneute Generalisierung notwendig wird? Darstellungsgrenzen lassen sich durch Minimaldimensionen ausdrücken. Diese geben Auskunft darüber, wie groß Objektteile sein müssen, um auf einem bestimmten Medium darstellbar und mit dem menschlichen Auge erkennbar zu bleiben. Tabelle 5.1 gibt einen Auszug empfohlener Minimaldimensionen für Papierkarten.

Streng mathematisch ergeben sich für jeden neuen Maßstab neue Generalisierungsparameter resultierend aus den vorgegebenen Minimaldimensionen. Ab einem bestimmten Maßstab ändern sich nicht nur die Parameter der einzelnen Operationen, sondern die Operationen selbst. Cecconi (2003) hat in seiner Arbeit ein Modul zur Online-Generalisierung entwickelt. Er bestimmt für jede Objektart unterschiedliche Komplexitätsbereiche, abhängig von dem Präsentationsmaßstab, so dass für jeden Bereich andere Generalisierungsmodule eingesetzt werden. So unterteilt er beispielsweise die Generalisierungsvorgänge für die Objektart *Gebäude* in drei Maßstabsbereiche. Bis zum Maßstab 1:50k findet nur eine Vereinfachung und Auswahl statt, zwischen 1:50k und 1:200k erhöht sich die Komplexität, da hier die Typifizierung und Verdrängung notwendig werden. Ab dem Maßstab 1:200k erfolgt dann der Übergang von Gebäudedarstellungen zu Siedlungsflächen. Ein vorbereiteter Datensatz ist somit an drei Stellen notwendig, ein hochauflösender Datensatz, ein Datensatz für den Maßstab 1:50k und ein Datensatz für den Maßstab 1:200k. Die dazwischenliegenden Datensätze werden in der zitierten Arbeit durch entsprechende Generalisierungsmodule erschlossen.

	<i>Mindestgröße</i>	<i>Mindeststrichstärke</i>	<i>Mindestabstand</i>
<i>Punktsignaturen</i>			
Runder Punkt	0,30mm		
Farbige Hohlform	0,60mm	0,10mm	
<i>Liniensignaturen</i>			
schwarz/weiß		0,08mm	
Doppellinie		0,08mm	0,25mm
<i>Flächensignaturen</i>			
Volles Quadrat	0,35mm		
Abstände			0,20mm
Vorsprünge	0,25mm		

Tab. 5.1: *Minimaldimensionen von Signaturen und Zwischenräumen (aus Spiess et al. 2002)*

In der vorliegenden Arbeit soll ein anderer Ansatz gewählt werden, welcher die Notwendigkeit einer Online-Generalisierung völlig ausschließt. Nach mathematischen Gesichtspunkten wäre zur Abdeckung aller Maßstäbe auch eine Vorbereitung aller Maßstäbe notwendig. Dies ist jedoch wenig praktikabel, da hierdurch ein großes und vor allen Dingen auch überflüssiges Datenvolumen entsteht, was gleichzeitig mit einem hohen Aufwand zur Erstellung und Integration verbunden ist. Vielmehr soll hier ein Ansatz gewählt werden, welcher an die bestehenden Kartenwerke angelehnt ist. So werden Karten für unterschiedliche Zwecke konzipiert und diese bedingen wiederum unterschiedliche Maßstäbe. Die oben beschriebenen Szenarien unterstreichen dies. Es gibt Situationen, in denen die lokale Information interessant ist, um beispielsweise ein bestimmtes Gebäude zu finden. Daneben werden Karten zur Übersicht genutzt, um eine Route für Fußgänger darzustellen oder um einen Überblick über eine Stadt oder einen Stadtteil zu gewinnen.

Nach Hake et al. (2002, S. 417) lassen sich topografische Karten unterteilen in:

- Topografische Grundkarten oder Plankarten bis 1:10k mit vorwiegend grundrisstreuer Darstellung
- Topografische Spezialkarten etwa zwischen 1:20k und 1:75k mit weitestgehend grundrissähnlicher Darstellung
- Topografische Übersichtskarten etwa ab 1:100k mit höherem Grad an Generalisierung
- Geografische Karten zur Darstellung landschaftlicher Raumverhältnisse ab 1:300k

Das ATKIS folgt einer ähnlichen Strategie und leitet aus seinen Datenmodellen Basis-DLM, DLM50, DLM250 und DLM1000 topografische Karten (DTKs²) der Maßstäbe 1:10k, 1:25k, 1:50k, 1:100k, 1:250k, 1:500k und 1:1Mio ab. Daneben existieren die DDGK5³ und in Niedersachsen die DSK10⁴. Auf die einzelnen Kartenwerke entfallen jeweils die in Tabelle 5.2 aufgezeigten Anwendungsgebiete.

Dieser Einteilung soll auch bei der Aufstellung der MRDB gefolgt werden, wobei auf die Maßstäbe ab 1:100k, welche zu einer überregionalen Darstellung und Anwendung führen, verzichtet werden soll. Somit sind nach dieser Strategie für jeden Anwendungsfall entsprechend generalisierte Kartendaten vorhanden. Im Bedarfsfall ist der Anwender nicht streng an den jeweils empfohlenen Maßstab gebunden, sondern kann die Darstellung nach Belieben vergrößern. Bei einer Verkleinerung sollte die nächstkleinere Auflösungsstufe zur Visualisierung gewählt werden. So können die Daten des Maßstabes 1:25k bis zum Maßstab 1:10k vergrößert werden, bevor die Daten des Maßstabes 1:10k verwendet werden. Bei einer Verkleinerung auf den Maßstab 1:30k sollten die Daten der Auflösungsstufe 1:50k verwendet werden.

5.2.2 Beschreibung der Testdatensätze

Durch die Implementierung einer MRDB sollen die möglichen Wege aber auch Hindernisse zur Realisierung einer solchen aufgezeigt werden. Das Ziel besteht darin, eine Datenbank aufzubauen, welche die oben beschriebenen Auflösungsstufen beinhaltet. Mit Hilfe der Testdaten werden die möglichen Ausgangssituationen zum Aufbau einer MRDB simuliert:

- Es existieren verschiedene Repräsentationen, welche jedoch noch nicht integriert sind

²Digitale Topographische Kartes

³Digitale Deutsche Grundkarte 1:5.000

⁴Digitale Straßenkarte 1:10.000

<i>Kartenwerk</i>	<i>Maßstab</i>	<i>Zweck</i>
Übersichtsplan, Grundkarte	bis 1:10.000	Arbeit der örtlichen Verwaltungen und Betriebe Lokale Zustellungsdienste und Rettungswesen Planungen durch Fachleute auf lokaler Ebene GIS und Analyse lokaler, raumbezogener Daten Stadtpläne und Ortspläne
Topografische Detailkarte	1:25 000	Möglichst vollständige Information über ein relativ kleines Gebiet Orientieren und Navigieren bei langsamem Fortbewegen im Gelände Detaillierte Information über die Geländestruktur Führung des Militärs im lokalen Bereich Kartieren von regionalen Planungen durch Fachleute und Planer Arbeiten der örtlichen Verwaltung und Betriebe Einsatz im Rettungswesen Grundlage für Darstellungen in kleineren Maßstäben Grundlage für Darstellung raumbezogener Daten in GIS
Topografische Detailkarte	1:50 000	Überblick über eine Region mit vielen Detailinformationen Orientieren und Navigieren im regionalen und lokalen Verkehr Information über die Geländestruktur Radwanderkarte und regionale Wanderkarte Führung des Militärs Planungen durch Fachleute und Planer Grundlage für Darstellung und Analyse raumbezogener Daten in GIS
Topografische Karte	1:100 000	Überblick über eine größere Region Zum Orientieren und Navigieren im regionalen Verkehr Erschließung der Siedlungen aufzeigen Information über die regionale Geländestruktur Einsatz in der Verwaltung und im Rettungswesen Führung des Militärs auf allen Stufen Grundlage für großräumige thematische Kartierungen Grundlage zur Darstellung raumbezogener Daten in GIS

Tab. 5.2: Anwendungsgebiete topografischer Karten in den unterschiedlichen Maßstäben (aus Spiess et al. 2002)

- Es ist nur ein hochauflösender Datensatz vorhanden, aus dem alternative, skalierte Repräsentationen abgeleitet werden müssen

Resultierend aus diesen Vorgaben sowie den Bedürfnissen der Anwendungsszenarien werden Gebäudedaten aus der ALK⁵, Siedlungsflächen und Ortslagen des ATKIS-Basis-DLM sowie Straßendaten der Firma *Tele Atlas* in einem 1,5 km x 1,5 km großen innerstädtischen Testgebiet in Hannover verwendet (siehe Abbildung 5.1).

Schließlich werden einzelne Gebäudepläne der Universität Hannover mit in das Konzept einbezogen. Durch die Integration dieser Daten soll die Möglichkeit zur Verlinkung von Zusatzinformationen aufgezeigt und getestet werden. Durch die Verknüpfungen werden dem Anwender alle zu einem bestimmten Objekt gehörigen Informationen zugänglich. Weitere Beispiele seien Fotos, Videos oder andere multimediale Informationen oder textliche Beschreibungen. In dem Szenario 1 benötigt der Feuerwehrmann z.B. den Gebäudeplan, um den Einsatz seiner Mannschaft planen zu können.

Gebäudepläne

Die Universität Hannover unterhält einen digitalen Bestand an Gebäudeplänen. Diese technischen Zeichnungen enthalten zweidimensionale Informationen über das Innere von Gebäuden, also die Lage und Form der Räume, Flure, Treppenhäuser, Fenster, Türen und so weiter sowie zusätzliche Textinformationen, wie Raumnummern oder Bemaßungen (siehe Abbildung 5.2). Die Pläne werden nach Gebäuden und Stockwerken getrennt. Sie liegen als lokale Zeichnung vor, was bedeutet, dass diese ihren Raumbezug lediglich über die textliche Beschreibung ihrer Adresse erhalten. Somit werden diese Daten erst durch eine Verknüpfung mit den georeferenzierten Gebäuden als zusätzlicher LoD verfügbar.

⁵Automatisierte Liegenschaftskarte

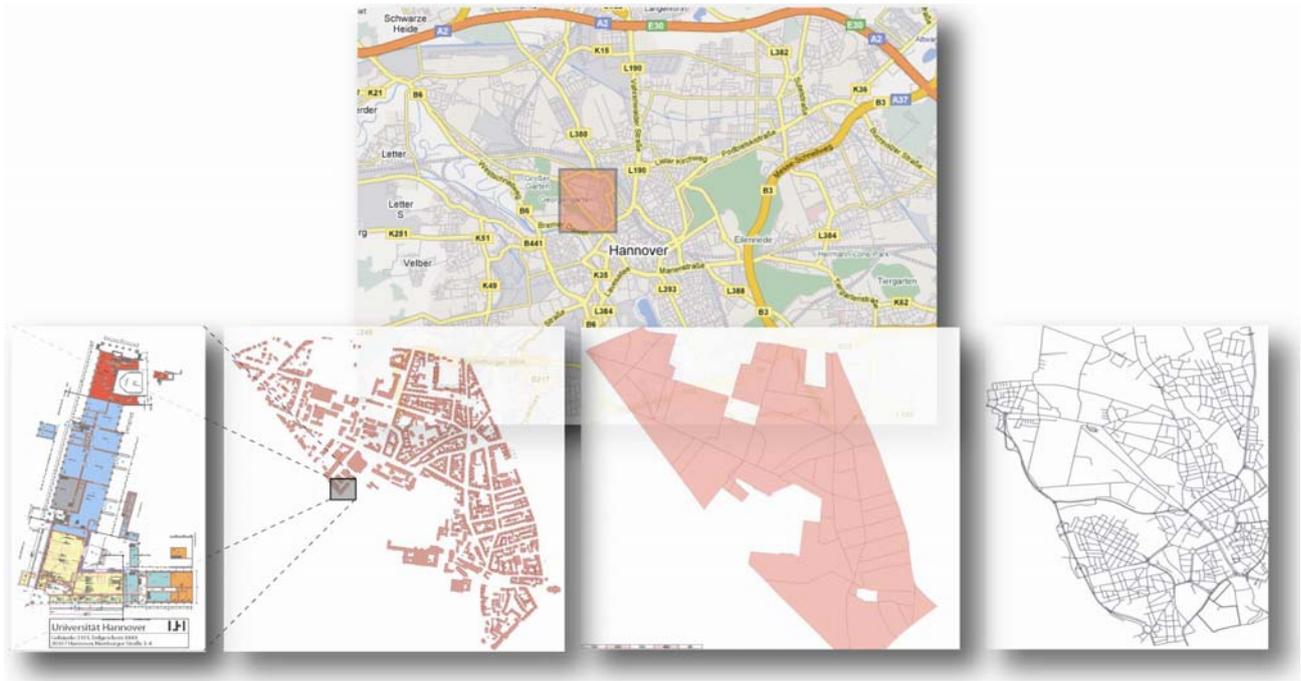


Abb. 5.1: Testdaten „Hannover-Nordstadt“: Lage des Testgebietes in Hannover (oben) (Quelle: <http://maps.google.com>); Gebäudeplan, ALK-Daten, ATKIS-Basis-DLM-Daten und Tele Atlas Daten (unten, v.l.)

ALK-Daten

Das Liegenschaftskataster besteht in Deutschland aus einem bildlichen sowie einem beschreibenden Teil. Die automatisierte Liegenschaftskarte stellt somit den bildlichen Teil des Liegenschaftskatasters dar und wird seit den 1970er Jahren digital bei den Katasterämtern geführt. Sie enthält:

- Angaben zur Geometrie: Flurstücksgrenzen, Grenzmarken, Grundrisse von Gebäuden
- Bezeichnende Daten: Flurstücksnummer, Flurnummer, Gemarkung, Lagebezeichnung
- Beschreibende Daten: Tatsächliche Nutzung, Ergebnisse der Bodenschätzung

Die Liegenschaftskarte (siehe Abbildung 5.3) wird für den Maßstab 1:1k erfasst und geführt, wobei die Daten zu Beginn der Automatisierung aus bestehenden Katasterkarten mit dem Maßstab 1:1k digitalisiert wurden. Die Genauigkeit der Neuerfassung liegt im Zentimeter- bis Dezimeter-Bereich, die der digitalisierten Informationen ist abhängig von der als Vorlage dienenden Karte und ist daher uneinheitlich. In der vorliegenden Arbeit werden ausschließlich die Gebäudegeometrien und -attribute der ALK genutzt. Die ALK stellt neben der DDGK5 den einzigen flächendeckend verfügbaren Gebäudedatenbestand in Deutschland dar.

Straßendaten der Firma *Tele Atlas*

Die Firmen *Tele Atlas* und *NAVTEQ* sind führende Anbieter digitaler Karten und dynamischer ortsbezogener Inhalte für die Navigation und ortsbezogene Dienste. So greifen die meisten Anbieter von Navigationssystemen und anderen geografischen Anwendungen auf die Daten eines dieser beiden Anbieter zurück.

Tele Atlas führt zwei Kartendatenbanken, *MultiNet™* und *Dynamap®*. Die *MultiNet™*-Daten beinhalten detaillierte und flächendeckende Informationen des Straßennetzes von Europa und Nordamerika. Der Datenbestand enthält Informationen über die Straßennummern und -klassifikationen, Kreuzungen, Verkehrsrichtungen und -beschränkungen. In der vorliegenden Arbeit sollen lediglich die Netzwerkgeometrien (NetworkClass) sowie die hiermit verknüpften Attributtabelle für die Straßennamen und Netzklassen verwendet werden. Folgende Straßenkategorien (Netclass 2) sind hier zu finden:

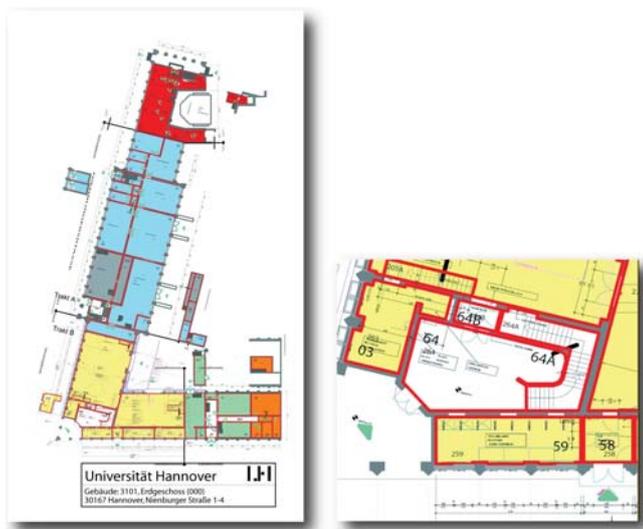


Abb. 5.2: Gebäudeplan der Universität Hannover (aus dem Datenbestand des Dezernats 3 - Gebäudemanagement der Universität Hannover)



Abb. 5.3: ALK Druckausgabe
(Quelle: <http://www.aed-sicad.de>)

- Netzklasse 1: Straßen von internationaler Bedeutung
- Netzklasse 2: Straßen von nationaler Bedeutung
- Netzklasse 3: Straßen von regionaler Bedeutung
- Netzklasse 4: Lokale Straßen von hoher Bedeutung
- Netzklasse 5: Lokale Straßen
- Netzklasse 6: Lokale Straßen von geringer Bedeutung
- Netzklasse 7: Eingeschränkte Straßen

Zusätzlich sind Informationen über Straßennamen, Hausnummern und weitere Angaben zu Orten von Interesse verfügbar (siehe Abbildung 5.4). Die Daten eignen sich zur Darstellung von Straßeninformationen in Karten, können darüber hinaus jedoch auch zur Routenberechnung verwendet werden. *Tele Atlas* garantiert eine Genauigkeit von zwölf Metern bis hin zu fünf Metern in innerstädtischen Bereichen. Die Informationen werden aus Papierkarten, Satelliten- und Luftbildern, von mobilen Kameras und mobilen Überwachungsteams zusammengestellt.

Digitale Landschaftsmodelle – ATKIS-DLM

Zur flächendeckenden Beschreibung der Topografie in Deutschland wurden im Zusammenhang mit dem ATKIS Projekt der deutschen Landesvermessung mehrere Kataloge zur Modellierung der Landschaft aufgestellt, namentlich der OK⁶25, OK50, OK250 und der OK1000. Hierin wird die Topografie zunächst mit Hilfe von Objektarten grob und durch Attributierung dieser dann fein gegliedert. Beruhend auf diesen Beschreibungen werden hieraus

⁶Objektartenkatalog

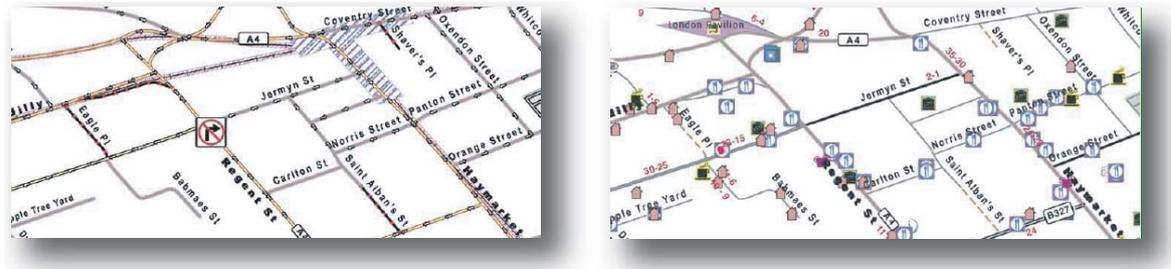


Abb. 5.4: Tele Atlas Daten zur Fahrzeugnavigation (links) mit PoIs (rechts) (aus: Tele Atlas MultiNet™ Broschüre 2006)

die digitalen Modelle Basis-DLM, DLM50, DLM250 sowie DLM1000 abgeleitet. Diese dienen wiederum als Basis zur Herstellung der digitalen topografischen Kartenreihen DTK.

Die Landschaftsmodelle unterteilen die Realwelt in die Objektbereiche 'Siedlung', 'Verkehr', 'Vegetation', 'Gewässer', 'Relief' und 'Gebiete' (wie Verwaltungsgebiete, Sperrgebiete etc.). Für das Basis-DLM (DLM25) gilt eine geometrische Genauigkeit von etwa drei, für das DLM50 von etwa fünf Metern. Das Basis-DLM stellt die genaueste Stufe bezüglich der Geometrie und der Inhalte dar. Aus diesem werden die Inhalte der übrigen Modelle durch semantische und Modell-Generalisierung abgeleitet.

Für das Testgebiet „Hannover-Nordstadt“ wurden nur Daten des Objektbereiches 'Siedlung' aus dem DLM25 und dem DLM50 übernommen. Diese sollen mit den ALK-Daten verknüpft werden und somit unterschiedliche Repräsentationen der besiedelten Gebiete bilden.

5.2.3 Datenbankentwurf

Zum Aufbau einer Datenbank, unabhängig davon, ob diese multi- oder monorepräsentative Daten enthalten soll, wird zuvor ein Datenmodell benötigt, welches sich nach den jeweiligen Anforderungen einer Anwendung richtet und eine formale Beschreibung des abzubildenden Sachverhaltes liefert. Der Entwurf einer Datenbank wird dabei auf Grund seiner Komplexität und Vielschichtigkeit in einzelne Teilschritte untergliedert, ausgehend von einer allgemeinen, nicht formalisierten, meist verbalen Anforderungsanalyse hin zu zunehmend formalisierten und implementierungsnahen Beschreibungen des Modells.

Der Datenbankentwurf wird klassischerweise in vier Bereiche untergliedert, die *Informationsbedarfsanalyse*, den *konzeptionellen*, den *logischen* und schließlich den *physischen Entwurf* (Anderl 2006) (siehe Abbildung 5.5).

Informationsbedarfsanalyse

Die Anforderungsanalyse erfasst zunächst den Informationsbedarf zukünftiger Anwendungen durch Interviews, Analyse von Arbeitsabläufen etc. Die Ergebnisse dieser Untersuchung werden in nicht-formaler, meist verbaler Form (Texte, Tabellen) dokumentiert.

Für den Fall der hier zu implementierenden, multiskaligen Datenbank folgen diese Anforderungen aus den vorangegangenen Abschnitten. Die in dem Unterabschnitt 5.2.1 beschriebenen konzeptionellen Auflösungsstufen topografischer Objekte sollen in der Datenbank abgebildet werden. Hierzu muss die Möglichkeit geschaffen werden, die Geometrie sowie zusätzliche Informationen eines Objektes in der Datenbank abzulegen. Jedes Realweltobjekt wird durch mehrere Repräsentationen beschrieben, wobei diese sich durch unterschiedliche geometrische Auflösungsstufen unterscheiden. Dabei sollte nicht zwingend jede Objektart in jedem LoD definiert sein, da durchaus Datensätze auch mehr als einen LoD bedienen können. Es muss lediglich sichergestellt werden, dass keine Lücken in den einzelnen LoDs entstehen oder diese in der Datenbank dokumentiert sind.

Weiterhin müssen in der Datenbank Informationen darüber zu finden sein, welche Objekte jeweils dasselbe Realweltobjekt repräsentieren beziehungsweise welche Objekte als alternative Repräsentation verwendet werden können. Das Modell muss darüber hinaus die hierarchische Struktur der Objekte widerspiegeln. Diese Relationen sollen dem Anwender einen einfachen Zugriff auf einzelne oder alle zu einem Phänomen gehörigen Informationen ermöglichen. Die maximale Auflösung der Objekte muss dabei in einem Basislevel zu finden sein, um andere Repräsentationen und Informationen aus diesem ableiten zu können.

Darüber hinaus sollten nicht nur alternative, sondern auch zusätzliche geometrische Informationen mit den entsprechenden Objekten verknüpft werden können, wie beispielsweise die Geometrien der innerhalb einer

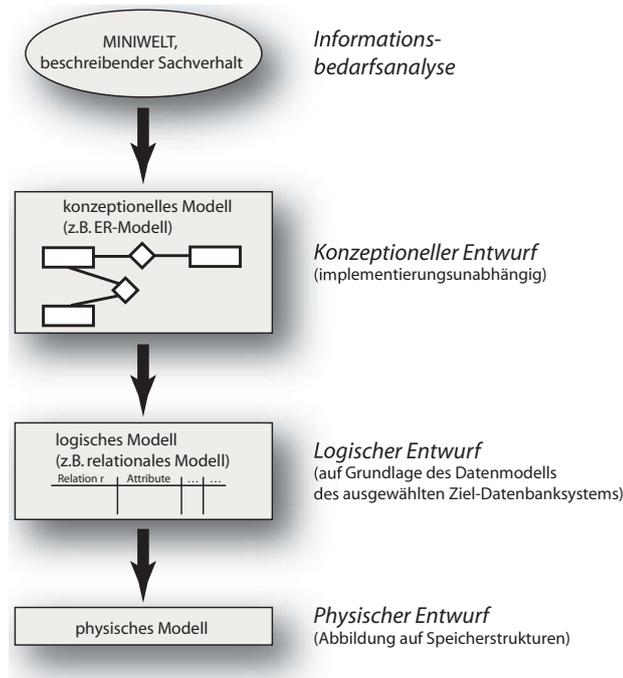


Abb. 5.5: Vorgehensweise beim Datenbankentwurf (nach Anderl 2006, S. 112)

Siedlungsfläche gelegenen Gebäude. Außerdem werden nicht nur die Objekte selbst, sondern auch Informationen über diese benötigt. Diese Informationen sollen dem Anwender oder der Applikation helfen, die optimale Repräsentation abfragen zu können. So sind Informationen über die verfügbaren Level, deren Genauigkeit, den empfohlenen Präsentationsmaßstab sowie weitere Metainformationen sinnvoll.

Schließlich sollen bestehende Datensätze möglichst einfach in das System integriert werden können, auch im laufenden Betrieb, also nach Fertigstellung des Systems. So soll es dem Datenanbieter möglich sein, neue Informationsquellen zu integrieren, ohne dass das Datenmodell angepasst werden muss.

Zusammenfassend lassen sich somit folgende Bedürfnisse aus der vorstehenden Analyse ableiten:

- Abbildung topografischer Objekte in der Datenbank
 - Speicherung und Abfrage der Geometrie eines Objektes
 - Speicherung und Abfrage der zu einem Objekt gehörigen Semantik
- Reflexion der Multiplizität
 - Verknüpfung der Objekte, welche dasselbe Phänomen oder Teilphänomen repräsentieren
 - Metadaten zur Beschreibung der Eigenschaften einer Auflösungsstufe

Es existieren darüber hinaus mehr Anforderungen an eine MRDB als die hier beschriebenen und implementierten. Eine MRDB erfordert Mechanismen, um die Konsistenz zwischen den Objekten einer Auflösungsstufe sowie den verschiedenen Repräsentationen auch nach Aktualisierungsvorgängen sicherzustellen. Integritätsbedingungen überprüfen beispielsweise auf topologische Fehler oder sonstige Widersprüche im System. Des Weiteren erleichtern inkrementelle Aktualisierungsmethoden unter Ausnutzung der Verknüpfungen die Automation eines Update-Prozesses über alle Auflösungsstufen hinweg. Diese Anforderungen wurden jedoch in anderen Arbeiten ausgiebig erörtert und zum Teil auch umgesetzt (Kilpeläinen 1998, 1995, Devogele et al. 1996, Parent 2000), so dass diese in der vorliegenden Arbeit in den Hintergrund gestellt werden sollen.

Konzeptioneller Entwurf

In dem konzeptionellen Entwurf werden die bei der Bedarfsanalyse gesammelten Anforderungen in eine formalisierte Beschreibungssprache transformiert. Hier entsteht das konzeptionelle Schema, welches eine Gesamtsicht auf die abzubildenden Daten beschreibt, diese jedoch unabhängig von implementierungs-, datenbanksystem- oder rechner-spezifischen Gesichtspunkten (Heuer & Saake 2000, S. 175). Dabei bestehen verschiedene Möglichkeiten,

diesen konzeptionellen Entwurf darzustellen, wie beispielsweise das ER-Modell⁷, der NIAM⁸, die OMT⁹ oder die UML. Diese erlauben durch entsprechende Hilfsmittel eine textliche und grafische Beschreibung. Gerade die grafische Modellierung ermöglicht eine übersichtliche Darstellung komplexer Zusammenhänge. Der Entwurf der multiskaligen Datenbank soll diesem Konzept folgen.

Modellkonzept In dem Abschnitt 3.4 werden verschiedene Möglichkeiten zur Modellierung einer MRDB aufgezeigt. Aus den dort genannten Gründen heraus soll an dieser Stelle die Möglichkeit einer impliziten Modellierung ausgeschlossen werden. Die Übergänge zwischen den einzelnen Repräsentationen werden daher explizit beschrieben. Die Möglichkeiten zur expliziten Modellierung reichen von *föderierten Datenbanksystemen*, welche bestehende Datenbanksysteme in einem übergeordneten Schema zusammenfassen und darüber hinaus die Integritätsbedingungen festlegen, über eine integrierte *Datawarehouse-Lösung*, in der die bestehenden Schemata der einzelnen Repräsentationen neu modelliert und die Daten in ein multirepräsentatives Schema überführt werden, bis hin zur Integration bestehender Schemata ohne deren originäre Struktur verändern zu müssen, indem lediglich die Beziehungen zwischen den einzelnen, bestehenden Repräsentationen beschrieben werden.

Somit besitzt jedes Objekt einer MRDB mehrere, verknüpfte Repräsentationen, wobei diese wahlweise gespeichert werden in

1. einer Datenbank mit einem integrierten Schema, welches alle Repräsentationen aufnimmt (Abbildung 3.5, S. 67),
2. einer Datenbank mit mehreren Schemata, wobei jedes Schema eine Auflösungsstufe aufnimmt (Abbildung 3.7, S. 68),
3. mehreren Datenbanken mit einem einfachen Schema (jede für einen Auflösungsstufe) (Abbildung 3.6, S. 68) oder
4. mehreren Datenbanken mit mehreren Auflösungsstufen.

Dem Beispiel der in dem EU-Projekt *MurMur* aufgezeigten Modellierung folgend (Unterabschnitt 3.2.2) sollen auf Grund der Diversität der zu integrierenden Datensätze diese als mono-repräsentative Objekttypen integriert und deren Beziehungen explizit modelliert werden. Somit wird nachfolgend der Fall 2 modelliert und implementiert.

Objektklassen An dieser Stelle sollen zunächst die drei Begriffe „Objektklasse“, „Objektbereich“ und „Objektart“ unterschieden werden. Die Objektklasse stammt aus der UML und fasst alle Elemente innerhalb eines Datenmodells zusammen, welche dieselben Eigenschaften, wie zum Beispiel Attribute, besitzen. Das konzeptionelle Schema beschreibt diese unterschiedlichen Objektklassen und ihre Beziehungen. Die beiden Begriffe Objektbereich und Objektart stammen dagegen aus dem ATKIS-Konzept zur Überführung der Realwelt in ein Modell. Dabei werden die Objekte der Realwelt einem bestimmten Objektbereich zugeordnet. Jeder Objektbereich enthält wiederum mehrere Objektarten. Aus den beschriebenen Testdatensätzen sollen die Objektarten 'Straße' (MultiNet-Daten), 'Gebäude' (ALK-Daten), 'Siedlungsfläche' und 'Ortslage' (ATKIS-Daten) in den Prototypen übernommen werden.

Die Objektart 'Gebäude' enthält neben der Geometrie die Attribute 'Hausnummer' und 'Straßenname'. Sie dient vor allen Dingen zur Orientierung in großmaßstäbigen Karten. So benötigt *Christa* in dem beschriebenen Szenario 2 (vgl. Unterabschnitt 4.1.1) eine detaillierte Darstellung eines Gebäudes, um die richtige Straßenkreuzung zu identifizieren oder um das auf ihrem PDA dargestellte Gebäude in der Realität finden zu können. Noch genauere Informationen enthält der 'Gebäudeplan', welcher exemplarisch eine mit einem Gebäude verknüpfte Informationsquelle darstellen soll. Diese CAD¹⁰-Zeichnung wird in der Datenbank abgelegt, so dass der Feuerwehrmann in dem beschriebenen Szenario 1 diesen beispielsweise über die Gebäudegeometrie in der Karte abfragen und die Lage der Räume ablesen kann. Der Gebäudeplan soll dabei nicht zu der Gruppe der Geobasisdaten gezählt werden, im Gegensatz zu den anderen hier vorgestellten Objektarten. Vielmehr stellt dieser eine raumbezogene Information dar, welche mit anderen Geodaten, in diesem Fall dem Gebäude, verknüpft wird und auf diese Weise über das Gebäudeobjekt zugänglich wird.

Die Objektart 'Siedlungsfläche' beschreibt Gebiete, welche ausschließlich oder vorwiegend dem Wohnen dienen. Die 'Ortslage' dagegen fasst 'Wohnbau-', 'Industrie- und Gewerbeflächen' sowie 'Flächen gemischter Nutzung' zusammen zu einem einzigen Gebiet. Diese Objektarten enthalten neben ihrer Geometrie den geografischen Namen

⁷Entity Relationship-Modell

⁸Nijssen's Information Analysis Method

⁹Object Modeling Technique

¹⁰Computer Aided Design

(GN) sowie die Einwohnerzahl (EWZ). Hierdurch soll es später exemplarisch möglich sein, die Einwohnerzahl oder auch Einwohnerdichte eines Stadtteils über ein bestimmtes Gebäude abfragen zu können. Die in den Szenarien geforderten Möglichkeiten zur Abfrage von Mietspiegeln oder gelagerten Gefahrgütern werden nicht in dem Prototypen integriert, beruhen jedoch auf demselben Modellierungs- und Abfrageprinzip, für das an dieser Stelle stellvertretend die Einwohnerzahl stehen soll. Die Objektart 'Ortslage' dient vor allen Dingen zur Orientierung in kleinmaßstäbigen Karten. Sie beschreibt dabei eine geringer aufgelöste Information als die 'Siedlungsfläche' und kann daher auch noch in kleinen Maßstäben bis etwa 1:200k verwendet werden, um beispielsweise mehrere Orte oder Stadtgebiete in einer Übersichtskarte darstellen zu können. Darüber hinaus lassen sich diese Objektarten als alternative Darstellungsform zu den Gebäuden verwenden, um den Detailreichtum auf dem Bildschirm des mobilen Gerätes reduzieren zu können.

Die Straßendaten werden mit ihrer Geometrie sowie den Attributen 'Netzklasse' und 'Straßenname' übernommen. Das Straßennetzwerk dient in vielen Anwendungen neben der Orientierung vor allen Dingen als Grundlage für Routingfunktionalitäten. Der Straßenname hilft bei der Navigation. Die Netzklasse kann beim Routing dazu verwendet werden, bestimmte Straßenkategorien zu meiden, zum Beispiel Nebenstraßen. Desweiteren soll die Netzklasse bei der späteren Generalisierung verwendet werden, um bei kleineren Maßstäben unwichtigere Straßenklassen zu eliminieren.

Das Schema dieser Datensätze soll durch die Integration nicht verändert werden. Jeder zu integrierende Datensatz kann nun zunächst einem Objektbereich zugeordnet werden, wobei die vorhandenen Datensätze auf die Bereiche 'Verkehr' und 'Siedlung' entfallen. Diese Zuordnung ist für die spätere Verlinkung wichtig, da nur Objekte derselben Objektbereiche horizontal und vertikal, das heißt zwischen den verschiedenen Auflösungsstufen und innerhalb derselben Auflösungsstufe, verknüpft werden sollen. Die genannten Objektbereiche stellen die in den beschriebenen Szenarien primär benötigten Daten zur Orientierung und Informationsabfrage dar.

Neben den vorhandenen sind neue Objektklassen erforderlich, welche aus den bestehenden Basisdaten abgeleitet werden, um die gewünschten LoDs in den Bereichen Verkehr und Siedlung vollständig abzudecken.

Verknüpfungskardinalitäten Der nächste Schritt zu einem integrierten Schema besteht nun darin, Verknüpfungen zwischen den korrespondierenden Elementen festzuhalten. Dabei ist es wichtig, die bestehenden und zu modellierenden Kardinalitäten zu bestimmen, bevor das konzeptionelle Schema aufgestellt wird. Es gilt, die Charakteristik der Beziehungen zwischen den Datensätzen zu erfahren, um eine angemessene Verknüpfungsstrategie einzusetzen. Abbildung 5.6 zeigt die möglichen Kardinalitäten, wobei hier nicht alle Formen der Generalisierung be-

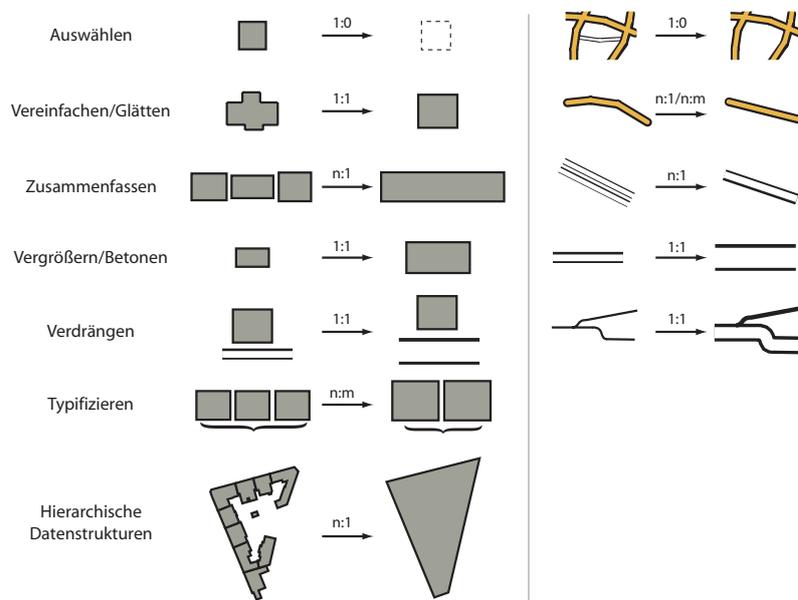


Abb. 5.6: Kardinalitäten zwischen den Objekten verschiedener Auflösungsstufen

trachtet werden, sondern lediglich die in der vorliegenden Arbeit auftretenden Fälle. Generalisierungsoperationen, wie Klassifizierung oder Symbolisierung, finden hier keine Berücksichtigung. Trotzdem wird schon anhand dieser Untermenge möglicher Verbindungen deutlich, dass alle denkbaren Verknüpfungskombinationen beachtet werden müssen, also 1:0, 1:1, 1:n sowie n:m.

Dabei wirft der Fall einer n:m-Beziehung den komplexesten Fall auf, sowohl bei der Modellierung als auch der späteren Integration. In diesen Fällen ist keine eindeutige Zuordnung zwischen den einzelnen Objekten möglich. Beschreiben beide Datensätze weiterhin kein eindeutiges Phänomen, über das diese beiden Gruppen von Objekten in Beziehung gesetzt werden können, so muss dieser Fall über eine Set-to-Set Relation beschrieben werden. Aus den einfachen werden komplexe Objekte. Auch die Definition einer Gruppe von Objekten ist in diesen Fällen schwierig, man denke beispielsweise an typifizierte Objekte. Welche Gruppe typifizierter Gebäude repräsentiert welche Gruppe der ursprünglichen Gebäude? An dieser Stelle soll dieses Problem umgangen werden, indem alle n:m Beziehungen in 1:n Beziehungen aufgespaltet werden. Dies bedeutet jedoch keine Einschränkung des Systems, da der Anwender jederzeit durch zusätzliche Module eine n:m-Beziehung beschreiben kann. Ist der Nutzer zum Beispiel an semantischen Gruppen von Straßensegmenten interessiert, welche über den Straßennamen identifiziert werden, so können die bestehenden Objekte zu Gruppen zusammengefasst werden und die Beziehungen zwischen den Objektgruppen aus den 1:n Relationen abgeleitet werden (vgl. Abbildung 5.7). Neben der Möglichkeit einer semantischen Gruppierung können die Aggregationen auch über die Geometrie bestimmt werden, indem die Elemente zwischen zwei eindeutig zuzuordnenden Punkten, zum Beispiel Straßenkreuzungen, gruppiert und in Beziehung zueinander gesetzt werden. Weitere Möglichkeiten zur Bestimmung von n:m Beziehungen finden sich unter anderem in Mantel (2002). Somit hat der Nutzer die Möglichkeit, ein zusätzliches Modul einzuführen, welches auf dem bestehenden Schema aufsetzt und dieses um eine Gruppierungsfunktionalität ergänzt, ohne dass das bestehende Schema geändert werden muss. Abbildung 5.7 beschreibt dieses Prinzip. In dem oberen Bereich

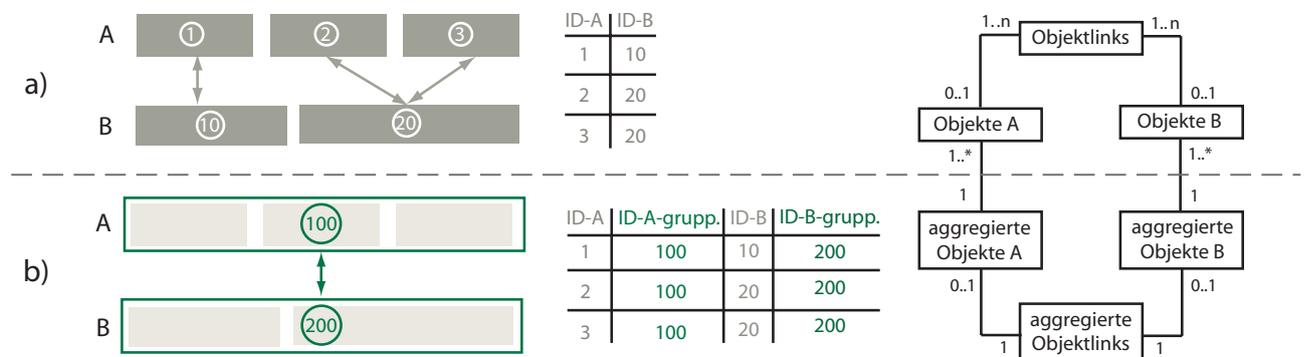


Abb. 5.7: Erweiterung des Systems um n:m-Beziehungen (tlw. nach Mantel & Lipeck (2004))

a) ist ein System dargestellt, wie es sich auch in dem beschriebenen Konzept wiederfindet. Teil b) zeigt den Fall eines zusammengesetzten Objektes: Die Applikation des Nutzers gruppiert die Objekte nach bestimmten Kriterien und leitet die Verknüpfungen aus den bestehenden ab. Das konzeptionelle Schema wird entsprechend ergänzt.

Verknüpfungsstrategie Schließlich gilt es nun, die Verknüpfungen zwischen den zu integrierenden Objekten zu modellieren. Die primäre Aussage einer Verknüpfung innerhalb einer MRDB besteht darin, dass die verknüpften Elemente alternative Repräsentationen (oder allgemein Sichtweisen) darstellen, welche unter Berücksichtigung der Unterschiede zwischen den Auflösungsstufen, alternativ eingesetzt werden können. So kann die vereinfachte die hochauflösende Geometrie in einer Karte ersetzen, besitzt jedoch beispielsweise einen unterschiedlichen, weil vereinfachten Flächeninhalt. Wir sprechen hier von den vertikalen Verknüpfungen zwischen den einzelnen LoDs, welche die vertikale Konsistenz sicherstellen. Die Eigenschaften der jeweiligen LoDs, wie Maßstabsbereiche, Auflösung o.ä. können in einer Metadatentabelle oder zusammen mit den Links gespeichert werden.

Zusätzlich existieren alternative Sichtweisen innerhalb eines LoD, beispielsweise die Gebäude und die Siedlungsflächen, welche beide Informationen über die Verteilung von Gebäuden und daraus abgeleitet die Bevölkerung zulassen. Eine horizontale Verknüpfung kann somit als eine Verknüpfung zwischen alternativen Sichtweisen interpretiert werden, welche jeweils für die Sichtweise relevante Informationen enthalten. Die Strategie besteht somit darin, sowohl alternative Auflösungsstufen als auch alternative Sichten innerhalb derselben Auflösungsstufe zu verknüpfen. Auf diese Weise wird es möglich, dem Feuerwehrmann einen detaillierten Grundriss des gefährdeten Gebäudes und gleichzeitig eine Übersicht über die Wohngegend mit den jeweils verknüpften Informationen zu liefern. Der Wohnungssuchenden aus dem Szenario 2 können auf diese Weise zusätzlich zu den Gebäudeinformationen auch Daten zu der Siedlungsfläche oder der Ortslage, in der sich das Gebäude befindet, angezeigt werden.

Aus den vorangegangenen Überlegungen folgt, dass in einem ersten Schritt unterschiedliche Auflösungsstufen desselben Objektbereiches (hier Verkehr und Siedlung) oder sogar nur derselben Objektart (Gebäude, Siedlungs-

fläche, Ortslage, Straßen), als Teilmenge der Objektbereiche, miteinander zu verknüpfen sind; also zum Beispiel die hochauflösenden Gebäude mit den vereinfachten Gebäuden. Dabei soll die Verknüpfung jedoch nicht nur die Konsistenz zwischen den unterschiedlichen Auflösungsstufen andeuten und erhalten. Vielmehr liegt das Ziel des beschriebenen MRDB-Systems liegt nun darin, Informationen aus den unterschiedlichen Auflösungsstufen je nach Bedarf bereitzustellen. Durch sogenanntes „Information drilling“ möchte der Feuerwehrmann durch einen Klick auf die zusammengefassten Gebäude eine nicht generalisierte Darstellung der einzelnen Objekte erhalten. Durch eine Top-Down Suche sollen also schrittweise höher aufgelöste Informationen abgefragt werden. Das Problem an dieser Stelle bereiten Objekte, welche als Folge der Generalisierung, zum Beispiel auf Grund ihrer geringen Größe, respektive ihres geringen Flächeninhaltes, weggefallen sind. So werden ab einem bestimmten Maßstab beispielsweise kleine Nebenstraßen oder die Hütte in einem Waldstück nicht mehr dargestellt. Möchte der Nutzer nun detailliertere Informationen beispielsweise zu einem bestimmten Straßensegment erhalten, so wird das System die verknüpften Objekte höherer Auflösung aufrufen, wird jedoch niemals die weggefallenen Objekte anzeigen können, da diese mit keinem Objekt geringerer Auflösung verlinkt sind (siehe Abbildung 5.8, links).

Um dieses Dilemma aufzulösen, wird an dieser Stelle folgende Annahme getroffen: Wünscht der Nutzer zusätzliche Informationen zu einem bestimmten Objekt, beispielsweise dem Straßensegment, so interessieren ihn lediglich höher aufgelöste Informationen hierzu, also zum Beispiel einzelne Fahrspuren, nicht jedoch abgehende Nebenstraßen. Ist dagegen nicht das Objekt, sondern die Situation relevant, so erfolgt die Abfrage nicht durch „Information drilling“ über das einzelne Objekt entlang der Verknüpfungen, sondern es wird ein neuer Datensatz anhand der den Datensätzen zugeordneten Auflösungsbereichen zusammengestellt. Hier tauchen dann die generalisierten Nebenstraßen wieder auf. Hat also die Studentin *Christa* aus dem Szenario 2 die Orientierung verloren und möchte eine detaillierte Darstellung ihrer näheren Umgebung, so wird eine räumliche Anfrage formuliert, welche die entsprechenden Objekte der höchsten Auflösungsstufe abfragt. Somit werden dann beispielsweise auch kleine Nebengebäude dargestellt.

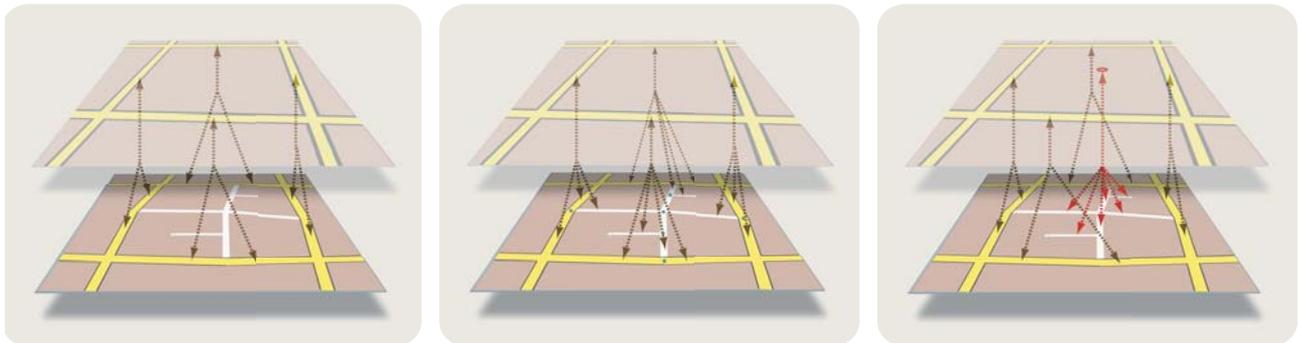


Abb. 5.8: Möglichkeiten zur Verknüpfung wegfallender Objekte mit der Ebene kleinerer LoDs: Keine Verknüpfung (links); Verknüpfung mit den benachbarten oder angeschlossenen Objekten derselben Objektart (z.B. abgehende Nebenstraßen) (Mitte); Verknüpfung mit dem Objekt, in dem sie aufgehen (z.B. Nebenstraße zu Siedlungsfläche) (rechts)

Ein alternativer Weg, welcher an dieser Stelle nicht eingeschlagen werden soll, führt über alternative Verknüpfungsstrategien. Zum einen ist es möglich zu versuchen, die Objekte mit benachbarten Objekten derselben Objektart zu verknüpfen, indem zum Beispiel wegfallende Straßen mit den Hauptstraßen kleinerer LoDs verknüpft werden. So kann eine Nebenstraße über den nächstliegenden Kreuzungspunkt mit einer Hauptstraße verknüpft werden (Abbildung 5.8, Mitte). Wegfallende Nebengebäude werden dem am nächsten stehenden Gebäude zugeschlagen. Hierzu können auch semantische Informationen, wie Eigentümer oder Straßename hinzugezogen werden.

Zum anderen kann argumentiert werden, dass wegfallenden Elemente in der sie umschließenden Fläche aufgehen. Somit können Gebäude, aber auch Straßen, mit der sie umgebenden Siedlungs- oder Waldfläche verknüpft werden (Abbildung 5.8, rechts). Ein durch einen Mausklick ausgelöstes 'Information drilling' würde dann alle in der Siedlungsfläche befindlichen Gebäude und Straßen darstellen.

Somit kann an dieser Stelle festgehalten werden, dass die Strategie zur Verlinkung von Objekten in einer MRDB stark von der gewünschten Anwendung abhängt. Ursprünglich sollten die Verknüpfungen die Konsistenz eines integrierten Systems sicherstellen. Eine Verknüpfung zweier Instanzen bedeutet, dass Veränderungen des einen Objektes auch Auswirkungen auf die alternativen Repräsentationen haben und an dieser Stelle die Konsistenzbedingungen neu zu überprüfen sind. So beschreibt zum Beispiel die mit einem Link verknüpfte Konsistenzregel, dass das generalisierte Gebäude A durch Zusammenfassung der Gebäude B, C und D entstanden ist. Wird das Objekt B gelöscht, so muss diese Zusammenfassung neu berechnet werden. Derartige Konsistenzregeln sollen in

dem hier vorgestellten Prototypen nicht implementiert werden, da der Fokus auf der Nutzung der integrierten Daten und weniger auf der Erhaltung der Konsistenz liegen soll. In der hier vorgestellten Arbeit erfüllen die Verknüpfungen die Anforderung, den Anwender bei der Informationssuche und -analyse zu unterstützen. Er soll die Möglichkeit erhalten, alle verfügbaren Informationen zu einem bestimmten Phänomen abfragen zu können. Die Verknüpfungen erfolgen somit zunächst zwischen den Objekten einer Objektart, wobei jeweils die verschiedenen Auflösungsstufen untereinander verknüpft werden sollen. Zusätzlich bieten horizontale Verbindungen zwischen den verschiedenen Objektarten desselben Objektbereiches, welche sich auf derselben Auflösungsstufe befinden, also beispielsweise zwischen Gebäuden und Siedlungsflächen, einen Informationsgewinn. Auch andere Informationen, wie die genannte Integration eines Gebäudeplans, können durch Verknüpfungen auf horizontaler Ebene beschrieben werden.

Die Verknüpfungen sind dabei bidirektional, das heißt, eine Abfrage der verknüpften Objekten kann in beide Richtungen, also feiner oder gröber werdend, erfolgen. Der Bedarf einer Top-Down-Verknüpfung (geringe Auflösung zu hoher Auflösung) wurde bereits beschrieben. Zusätzlich soll jedoch die Möglichkeit offeriert werden, von einer hochauflösenden zu einer geringer aufgelösten Darstellung (Bottom-Up) zu gelangen. So bewegt sich *Christa* in dem Szenario 2 beispielsweise von der ersten Wohnung zu der nächsten. In dem Moment wird die bereits besichtigte Wohnung uninteressant und kann mit weniger Details dargestellt werden, wohingegen die zweite Wohnung nun mit einer höheren geometrischen Auflösung repräsentiert werden sollte.

Schließlich muss entschieden werden, ob es bei der vertikalen Verlinkung ausreicht, lediglich benachbarte LoDs miteinander in Beziehung zu setzen oder ob explizit alle möglichen Kombinationen modelliert werden müssen. Auch wenn auf den ersten Blick die erstgenannte Option einfacher und die zweite redundant erscheint, so soll trotzdem diese zweite Möglichkeit in dem Konzept vorgesehen werden. Hierdurch wird die Effizienz des Systems gesteigert, da zur Abfrage der Beziehungen zwischen zwei bestimmten, nicht benachbarten Ebenen nicht alle Verknüpfungen nacheinander abgearbeitet werden müssen, sondern die relevante Verbindung direkt abgefragt werden kann. So möchte der Feuerwehrmann im Szenario 1 beispielsweise eine möglichst detaillierte Darstellung eines bestimmten Gebäudes. Fragt dieser diese Information durch Klick auf ein Objekt mit der geringsten Auflösung (LoD4) ab, so müsste der Webservice zunächst die hiermit verknüpften Objekte der nächst höheren Auflösungsstufe (LoD3) abfragen. Nachdem das Ergebnis an den Service zurückgesendet wurde, müsste dieser, basierend auf diesem Resultat, eine neue Anfrage formulieren, welche die verknüpften Objekte in der nächst höheren Auflösung (LoD2) abfragt. Somit wären auf diese Weise eine Reihe von Abfragen und Datentransfers notwendig. Eine direkte Verknüpfung zwischen dem LoD0 und dem LoD4 ermöglicht dagegen einen direkten Zugriff auf die Verknüpfungen.

Auf eine diagonale Verknüpfung, also zusätzlich zwischen allen LoDs und allen Objektarten, beispielsweise der Objektart 'Gebäude' des LoD0 mit der Objektart 'Wohnfläche' in der Auflösungsstufe LoD1 soll an dieser Stelle auf Grund der Übersichtlichkeit des Systems verzichtet werden. Diese Informationen können bei Bedarf implizit abgeleitet werden.

Somit ergibt sich aus den vorangegangenen Überlegungen das in Abbildung 5.9 dargestellte konzeptionelle Schema der multiskaligen Datenbank zur Unterstützung von Webservice-Anwendungen.

Wahl des Datenbankmodells

In dem vorangegangenen Abschnitt wurde ein konzeptioneller Datenbankentwurf vorgestellt, welcher eine allgemeingültige, systemneutrale Beschreibung der zukünftigen Datenbankinhalte liefert. In einem folgenden Schritt, dem logischen Entwurf, gilt es, diesen konzeptionellen Entwurf auf das ausgewählte Datenbankmodell zu übertragen. Grundsätzlich sind an dieser Stelle vier Modellvarianten zu unterscheiden:

- Hierarchisches Datenmodell
- Netzwerkdatenmodelle
- Relationales Datenmodell
- Objektorientiertes Datenmodell

Darüber hinaus sind das *objektrelationale Datenbankmodell*, welches das relationale Datenmodell um objektorientierte Eigenschaften erweitert, sowie das *multidimensionale Datenbankmodell* zu nennen. Die auf diesen Modellen aufsetzenden Datenbankinstanzen werden folglich als hierarchische, Netzwerk-, relationale, objektorientierte oder objektrelationale Datenbanken bezeichnet. Nach Anderl (2006) sind das hierarchische als auch das Netzwerkdatenmodell in dem Bereich technischer Anwendungen von geringerer Bedeutung während relationale Datenbanksysteme momentan Stand der Technik sind. Eine detailliertere Beschreibung der Modellvarianten soll

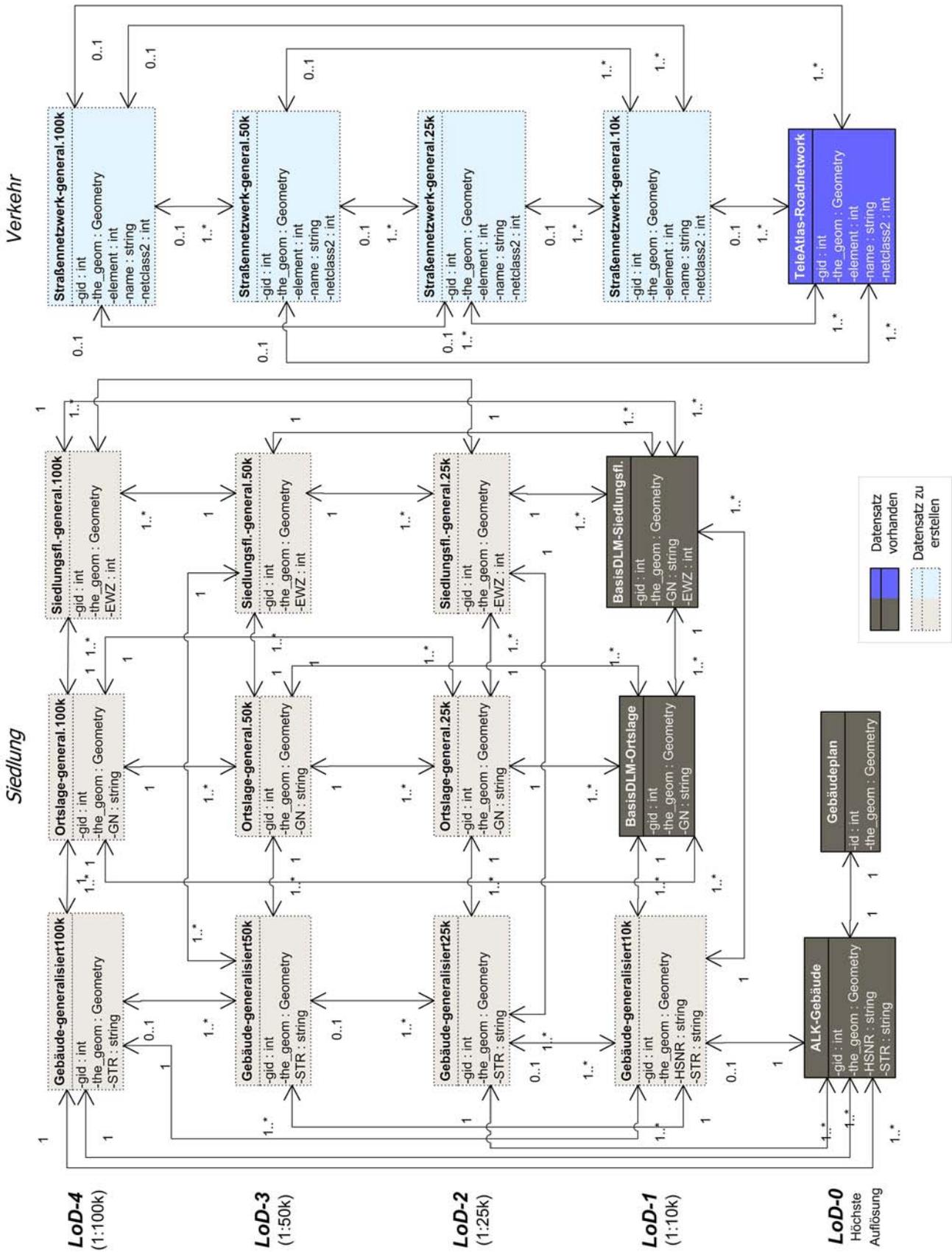


Abb. 5.9: Konzeptionelles Schema der MRDB: Beschreibung der Auflösungsstufen (LoD), Objektklassen und Verknüpfungen

an dieser Stelle ausgelassen werden und kann in der entsprechenden Literatur (siehe z.B. Heuer & Saake (2000) oder Anderl (2006)) nachgelesen werden.

Bei der Wahl eines geeigneten Datenmodells oder letztendlich eines geeigneten DBMS sind die in der Informationsbedarfsanalyse beschriebenen Anforderungen an das System zu berücksichtigen. Eine essenzielle Bedingung besteht in der Speicherung geometrischer Objekte. Ein räumliches DBMS ist charakterisiert durch die räumliche Erweiterung eines DBMS um folgende Eigenschaften (Shekhar & Chawla 2003):

- Ein SDBMS¹¹ ist ein Software-Modul, welches auf einem DBMS (beispielsweise einem objektrelationalen oder objektorientierten DBMS) aufsetzt und mit diesem zusammenarbeitet
- SDBMS unterstützen verschiedene räumliche Datenmodelle, enthalten räumliche abstrakte Datentypen (ADTs) und eine Abfragesprache (DDL), von der aus diese ADTs aufrufbar sind
- SDBMS unterstützen die räumliche Indizierung, effiziente Algorithmen für räumliche Operationen und domänenspezifische Regeln für die Optimierung von Abfragen

Auf dem Markt verfügbare Systeme zeichnen sich in der Regel dadurch aus, dass diese ein Modul anbieten, welches auf existierende, relationale oder objektrelationale Datenbanken aufsetzt und diese um die oben genannten Funktionalitäten erweitert. So ergänzt beispielsweise das Produkt *Oracle Spatial 10g* die *Oracle 10g* Datenbank, welche Daten sowohl relational als auch objektrelational speichert, um räumliche Funktionalitäten.

Somit stehen an dieser Stelle eine Reihe von SDBMS zur Verfügung. Die Wahl fällt letztendlich auf das freie, objektrelationale, Open Source Datenbanksystem *PostgreSQL*¹², welches durch die ebenfalls freie und als Open Source deklarierte räumliche Erweiterung *PostGIS*¹³ auch zur Speicherung und Verwaltung von räumlichen Daten eingesetzt werden kann. Dieses SDBMS erfüllt alle oben genannten Anforderungen, *PostGIS* folgt den Spezifikationen des *OGC* für räumliche Objekte, *PostgreSQL* unterstützt die SQL-92 und SQL-99 Standards der *ISO* und *ANSI*. In dem Bereich der definierten Anforderungen unterscheidet sich dieses System daher in keinerlei Hinsicht von den kommerziellen Produkten und wurde gleichzeitig konform zu bestehenden Standards entwickelt.

Logischer Entwurf

Es gilt nun, im logischen Entwurfsschritt, das konzeptionelle Modell (vgl. Abbildung 5.9) in das Datenmodell der ausgewählten Datenbank (Datenbankmodell) zu überführen, in dem vorliegenden Fall (*PostgreSQL*) also in ein objektrelationales Modell, welches über *PostGIS* zusätzlich Objekttypen und Operationen für geometrische Objekte bereitstellt.

Schema der Objekttabellen Die Objekte werden in der objektrelationalen Datenbank in verschiedenen Tabellen organisiert, wobei jede Tabelle Entitäten einer bestimmten Objektklasse aufnimmt. Das Schema der einzelnen Datenbanktabellen kann aus den in Abbildung 5.9 dargestellten Klassen abgeleitet werden. So entsteht beispielsweise aus der Objektklasse 'ALK-Gebäude' zunächst ein Tupel bestehend aus vier Elementen und somit eine Tabelle mit vier Spalten ('gid', 'the_geom', 'HSNR', 'STR'). Zusätzlich wird von dem DBMS automatisch eine weitere Spalte 'OID' als Objektidentifikator angelegt. Die Geometrie des Objektes wird in der Spalte 'the_geom' entsprechend der OGC-SFS (OGC 1999) abgelegt. Ein Gebäude wird somit beispielsweise in der Form 'POLYGON ((3548889.4 5806454.3, 3548891.1 5806466.5, ..., 3548889.4 5806454.3))' gespeichert. Das *PostGIS*-Modul legt zusätzlich zwei Tabellen 'geometry_columns' und 'spatial_ref_sys' an, wobei die erstgenannte Tabelle die Geometrien der Datenbank verwaltet. Enthält eine neu definierte Tabelle auch eine Geometriespalte, so erfolgt automatisch ein entsprechender Eintrag in der Tabelle 'geometry_columns'. Die zweite Tabelle verwaltet die Referenzsysteme, auf die in der Tabelle 'geometry_columns' verwiesen wird. Hier können Projektionsvorschriften und Parameter verschiedener Koordinatensysteme erfasst werden.

Die Unterteilung der Realwelt in einzelne Objektarten und schließlich in Objektinstanzen erfolgt durch eine entsprechende Datenmodellierung. So wird zur Modellierung eines Gebäudes eine neue Datenbankinstanz mit einer flächenhaften Geometrie gebildet. Bei dem Straßennetzwerk besteht der Bedarf, dieses aus Linienzügen bestehende Netz in einzelne Objekte zu zerlegen. Der Straßen-Datensatz wird in einzelne Straßensegmente unterteilt, welche jeweils zwischen zwei definierten Knotenpunkten verlaufen. Ein Knotenpunkt wird gebildet, wenn eine Straße durch eine andere Geometrie, wie kreuzende Straßensegmente, Eisenbahnlinien, Grenzen etc.,

¹¹Spatial Database Managementsystem

¹²<http://www.postgresql.org>

¹³<http://postgis.refrains.net>

unterbrochen wird. Ein so definiertes Straßensegment stellt dann eine Objektentität dar, welche in der Datenbank als Linienzug abgelegt werden kann.

Das auf diese Weise entstandene Datenbankschema, repräsentiert durch verschiedene, an dieser Stelle noch leere Tabellen, kann nun mit Objektinstanzen gefüllt werden, vorausgesetzt diese entsprechen dem definierten Schema.

Schema der Verknüpfungen Neben den Objekten gilt es zusätzlich, die Verknüpfungen zwischen diesen in dem Datenbankschema zum Ausdruck zu bringen. An dieser Stelle sind verschiedene Möglichkeiten denkbar, welche in der Abbildung 5.10 a-d dargestellt sind :

- Speicherung der Verknüpfungen als zusätzliches Attribut zum Objekt (Fall a/b)
- Eine Verknüpfungstabelle für jede Beziehung (Fall c)
- Speicherung aller Verknüpfungen in einer Tabelle, unterteilt nach Objektarten (Fall d)

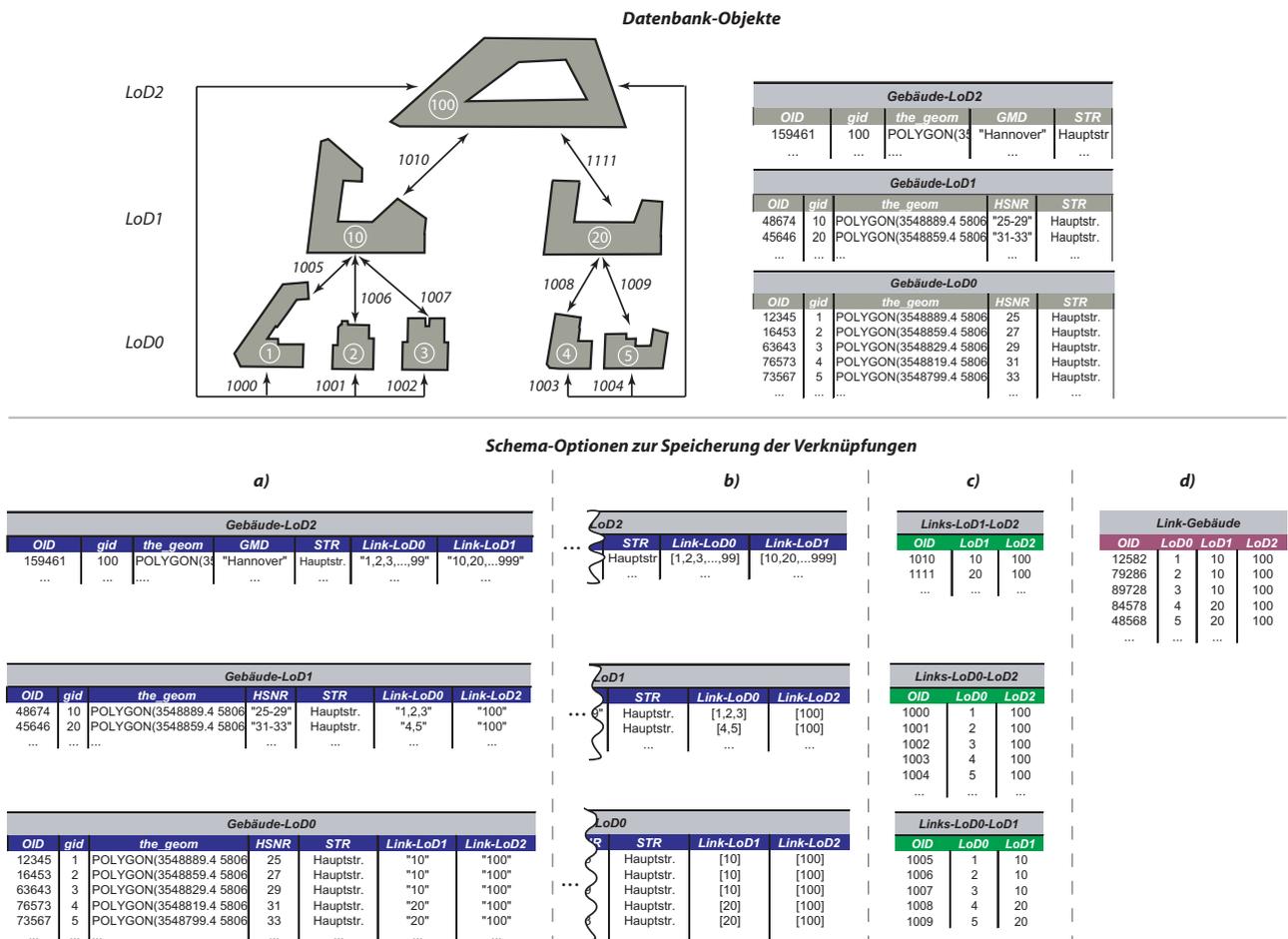


Abb. 5.10: Realweltobjekte und entsprechende Datenbank-Instanzen (oben); Schema möglicher Verknüpfungen (unten): a) zusätzliche Attribute (Link-LoD0, Link-LoD1) als String, b) zusätzliche Attribute (Link-LoD0, Link-LoD1) als Array, c) Link-Tabellen für jede Verknüpfung, d) Eine Link-Tabelle pro Objektart

Die Abbildung stellt im oberen Teil die Speicherung unterschiedlich aufgelöster Gebäude-Objekte dar. In jeder Auflösungsstufe (LoD) wird für jede Objektart (z.B. Gebäude) eine Tabelle angelegt. In dem gezeigten Beispiel werden die Gebäude der höchsten Auflösungsstufe in der Tabelle 'Gebäude-LoD0', die generalisierten Gebäude in der Tabelle 'Gebäude-LoD1' und die stärker generalisierten Gebäude in der Tabelle 'Gebäude-LoD2' gespeichert. Jede dieser Tabellen enthält eine Spalte zur Speicherung der Geometrie sowie weitere für die Attribute, wie im vorangegangenen Absatz beschrieben wird.

Der untere Teil der Abbildung stellt nun die vier möglichen Schemata der durch die Pfeile angedeuteten Verknüpfungen dar.

Fall a/b: In diesen Fällen werden die Identifikatoren (ID) der verknüpften Objekte direkt als zusätzliche Attribute zum Objekt gespeichert. Das bedeutet, jede der genannten Tabellen zur Speicherung der Objekte erhält

zusätzlich für jeden weiteren, vorhandenen LoD mindestens eine zusätzliche Spalte. Die Tabelle 'Gebäude-LoD0' erhält somit die zusätzlichen Spalten 'Link-LoD1' und 'Link-LoD2', um dort die Objektidentifikatoren der verknüpften Objekte einzutragen (z.B. 10 und 20 für Link-LoD1 und 100 für Link-LoD2 als Attributwerte zu dem Objekt '1'). Hier wären daher zwei zusätzliche Spalten notwendig. Da die Anzahl der verknüpften Objekte variabel ist, kann nicht für jede Verknüpfung eine neue Spalte angelegt werden. Dies würde zu einer großen Anzahl leerer Felder führen. Außerdem wird auf diese Weise eine Zuordnung schwierig, auf welchen LoD die jeweilige Spalte nun verweist, da die Verknüpfung zu einer bestimmten Auflösungsstufe dann mehrere Spalten umfasst. Vielmehr können die IDs der verlinkten Objekte in einem Attribut vom Typ *String* (z.B. *String link="1,2,3"*) oder *Array* (z.B. *Integer links[3]={1,2,3}*) zusammengefasst werden.

Beide Möglichkeiten bieten den Vorteil, dass die Verknüpfungsinformation direkt an dem jeweiligen Objekt hängt. Wird ein Objekt über eine Webdienst-Schnittstelle, beispielsweise einem WFS, abgefragt, so erhält der Anwender direkt auch Auskunft über die IDs der verknüpften Objekte und kann diese Information weiterverarbeiten. Fragt die Applikation beispielsweise das generalisierte Gebäude 10 des LoD1 aus der Datenbank ab, so enthält der resultierende Datensatz neben den objektspezifischen Beschreibungen wie Geometrie oder Hausnummer zusätzlich das Attribut 'Link-LoD0' mit den Attributwerten '1,2,3' sowie das Attribut 'Link-LoD2' mit dem Wert '100'. Der Nutzer kann nun den String oder den Array in seine Elemente zerlegen, um dann die Objekte '1,2,3' aus dem LoD0 abzufragen, um höher aufgelöste Objekte zu erhalten, oder das Objekt 100 aus dem LoD2 für eine geringer aufgelöste Darstellung.

Beide Lösungen sind jedoch bei der Datenabfrage wenig flexibel. Ein Daten-String bietet keine direkte Möglichkeit, in der Datenbank auf die einzelnen Elemente zurückzugreifen. Das Datenbanksystem betrachtet diesen als ein Objekt. Möchte der Nutzer die einzelnen Teilelemente dieser Zeichenkette, also die IDs der verknüpften Objekte, nutzen, so muss der String zunächst von einer zusätzlichen Applikation ausgelesen und zerschnitten werden. Darüber hinaus stößt diese umständliche Form der Datenverarbeitung auch bei Aktualisierungsvorgängen auf Probleme. Werden neue Objekte in die Datenbank integriert, so bedeutet dies, dass an einigen Stellen auch die Verknüpfungen aktualisiert werden müssen. In dem Fall einer Zeichenkette bedeutet dies, dass diese erneut von einer zusätzlichen Applikation ausgelesen, der String ergänzt und in die Datenbank zurückgeschrieben werden muss. Hierdurch wird es schwierig, die Konsistenz innerhalb der Datenbank aufrechtzuerhalten, da innerhalb des DBMS keine direkte Möglichkeit besteht, auf die einzelnen Elemente der Zeichenkette zuzugreifen. Wird im LoD0 ein neues Objekt (z.B. Gebäude 6) hinzugefügt, so bedeutet dies, dass auch eine entsprechende Verknüpfung, zum Beispiel mit dem Objekt 20 eingetragen werden müsste. Der Eintrag '4,5' in der Spalte 'Link-LoD0' in der Tabelle 'Gebäude-LoD1' ist nicht mehr aktuell. Es ist jedoch über die Datenbanksprache SQL nicht möglich, den String zu erweitern. SQL-Datenabfragen (*SELECT*) können außerdem die in der Zeichenkette verborgenen Informationen nicht für weitere Bedingungen nutzen. Liefert die Abfrage *SELECT 'Link-LoD0' FROM 'Gebäude-LoD1' WHERE 'gid'=20* zwar noch den String '4,5', so kann dieses Ergebnis innerhalb der Datenbank nicht weiter verwendet werden. Die Abfrage verknüpfter Objekte ist somit nicht ohne Weiteres möglich.

Der hier beschriebene Webdienst möchte dagegen genau diese Funktionalitäten bieten. Aus einem konsistenten System heraus sollen auf möglichst einfache Weise alle zu einem Objekt verfügbaren Informationen abgefragt werden können. Somit muss hier nach einer alternativen Möglichkeit gesucht werden.

Im Gegensatz zu der Zeichenkette bietet ein *Array* die Möglichkeit, auf seine einzelnen Elemente innerhalb der Datenbank zuzugreifen. Es ist des Weiteren möglich, Werte innerhalb eines Arrays zu suchen und somit verknüpfte Objekte direkt abfragen zu können. Trotzdem bereitet auch die Verwendung von Arrays Probleme. Das Auswählen, Sortieren oder Löschen von Elementen ist nicht problemlos möglich. Das PostgreSQL-Nutzerhandbuch¹⁴ sagt hier: *"Arrays are not sets; searching for specific array elements may be a sign of database misdesign. Consider using a separate table with a row for each item that would be an array element. This will be easier to search, and is likely to scale up better to large numbers of elements. The array field should generally be split off into a separate table. Tables can obviously be searched easily."* Arrays erschweren also die Suche innerhalb der Datenbank und bereiten darüber hinaus Probleme, wenn diese zu groß werden. Außerdem ist es erstrebenswert, eine normalisierte Form der Datenbank zu erreichen. Eine Normalisierung ist notwendig, um Redundanzen der Daten zu vermeiden, die einen erhöhten Speicherplatz benötigen, das Durchsuchen und Analysieren der Daten verlängern und bei der Änderung von Daten zu Inkonsistenzen führen können. Die erste (von drei) Normalform besagt unter anderem, dass jedes Attribut der Relation einen atomaren Wertebereich haben muss. Das heißt, zusammengesetzte, mengenwertige oder geschachtelte Wertebereiche sind nicht erlaubt. Diese Regel wird durch die Verwendung eines Arrays an dieser Stelle verletzt. Schließlich ist nicht sichergestellt, dass der Datentyp *Array* von einem zugreifenden Client

¹⁴<http://www.postgresql.org/docs/8.1/static/arrays.html>

interpretiert werden kann. So werden beispielsweise die Arrays nicht direkt von PHP¹⁵, einer Skriptsprache zur Erstellung dynamischer Webseiten, unterstützt. Gerade in einer Webserviceumgebung ist es wichtig, die Zusammenarbeit zwischen den einzelnen Modulen nicht zusätzlich zu erschweren.

Fall c, d): Somit ist an dieser Stelle zu empfehlen, die Beziehungen in einer separaten Verknüpfungstabelle zu speichern. Es verbleibt schließlich die Überlegung, ob die Verknüpfung zwischen mehreren Ebenen, also verschiedenen LoDs, durch separate (Fall c) oder durch eine einzige Tabelle (Fall d) erfolgen soll. Ein Beispiel soll diese Optionen verdeutlichen. In dem Beispiel der Abbildung 5.10 bestehen zwischen den verschiedenen Gebäude-Repräsentationen insgesamt drei Verknüpfungen (LoD0-LoD1, LoD0-LoD2, LoD1-LoD2). Die Abbildung skizziert die resultierenden Verknüpfungstabellen c) und d). In dem Fall c) erhält jede Verknüpfungsebene, in dem Beispiel also LoD0-LoD1, LoD0-LoD2 und LoD1-LoD2 eine eigene Verknüpfungstabelle. Jede weitere Objektart (z.B. Straße) bedingt pro Ebene eine weitere Tabelle. Die Tabelle enthält für jede Ebene, auf welche sie verweist, eine Spalte. Da im Fall c) nur jeweils zwei Ebenen verlinkt werden sollen, sind hier jeweils zwei Spalten notwendig. Jede Zeile repräsentiert somit eine Verknüpfung oder einen Pfeil aus dem oberen Teil der Abbildung. Die obere Tabelle 'Links-LoD1-LoD2' des Falles c) besagt zum Beispiel, dass Objekt 10 mit Objekt 100 und Objekt 20 mit Objekt 100 verknüpft ist. Der Fall d) verfolgt eine ähnliche Strategie, mit dem Unterschied, dass alle Verlinkungen in einer Tabelle pro Objektart zu finden sind. So besagt die erste Zeile der Tabelle 'Links-Gebäude' im Fall d), dass das Objekt 1 (im LoD0) mit dem Objekt 10 (im LoD1) und dem Objekt 100 (im LoD2) verlinkt ist. Gleichzeitig ist auch das Objekt 10 mit dem Objekt 100 verknüpft.

Werden für jede Verknüpfung separate Tabellen verwendet, so entstehen mehr Tabellen und auch mehr Tabelleneinträge insgesamt, was an der Abbildung abzulesen ist. Aus der Sicht des Webservice-Systems und auch des Anwenders ist die Speicherung der Verknüpfungen in möglichst wenig Tabellen vorzusehen. Die in dem Szenario beschriebene Studentin benötigt beispielsweise alle zu einem Objekt verlinkten Informationen. Die Serviceschnittstelle muss im Fall c) diese Informationen aus mehreren Tabellen zusammensuchen. Konkret bedeutet dies, dass mehrere Anfragen an die Datenbank gestellt werden, die jeweils einen Suchprozess auslösen und die Ergebnisse anschließend zu einem Resultat kombiniert werden müssen. Möchte der Nutzer zum Beispiel alle verknüpften Repräsentationen zum Objekt 100 erhalten, so wird zunächst die Tabelle 'Links-LoD1-LoD2' nach der ID 100 durchsucht. Das System findet hier die IDs '10' und '20' und sendet diese an den Client. Anschließend wird in einer zweiten Anfrage die Tabelle 'Links-LoD0-LoD2' untersucht, welche die IDs 1, 2, 3, 4, 5 zurückgibt. Nun können in einer dritten Anfrage die Objekte 1, 2, 3, 4, 5, 10 und 20 sowie deren Attribute abgefragt werden.

Dies bedeutet also einen erhöhten Datenverkehr zwischen der Schnittstelle und der Datenbank sowie zusätzliche Arbeitsschritte. Letztendlich entsteht für den Anwender eine längere Wartezeit, was gerade bei mobilen Anwendungen zu vermeiden ist. Somit wird in dem zu implementierenden Prototypen die Lösung d) verwendet.

Eine Trennung sollte lediglich zwischen den horizontalen und vertikalen Verknüpfungen erfolgen, so dass eine Tabelle Auskunft gibt über die Beziehungen zwischen den Repräsentationen einer Objektart in seinen verschiedenen Auflösungsstufen (vertikal), eine andere Tabelle über die Verknüpfungen verschiedener Objektarten innerhalb einer Auflösungsstufe (horizontal).

Somit ist es möglich, zwischen den einzelnen Repräsentationen gezielt nach Informationen zu suchen, wie es in den Szenarien gefordert wird. Der Feuerwehrmann Karl kann zur Abfrage detaillierter Informationen ('Information drilling') zu einem Gebäude in seiner Anwendung auf das generalisierte Gebäude 10 des LoD1 klicken. Anschließend wird die Tabelle 'Links-Gebäude' in der Spalte 'LoD1' nach dem Vorkommen des Attributwertes '10' durchsucht. Die Datenbank liefert drei Tabellen-Zeilen zurück, aus der die Attributwerte für das Attribut 'LoD0' gefiltert werden können, in diesem Fall also '1, 2, 3'. Anschließend kann der Service die Geometrien der Gebäude mit dem Attributwert 1, 2 und 3 abrufen und darstellen.

Sollen neue Objekte und daraus folgend auch neue Verknüpfungen in die Datenbank eingefügt werden, so muss lediglich eine neue Zeile in die Verknüpfungstabelle eingetragen werden. Wird das Objekt 6 im LoD0 neu eingefügt, welches mit den Objekten 20 und 100 verknüpft ist, so wird eine neue Zeile angelegt mit den Attributwerten LoD0='6', LoD1='20' und LoD2='100'. Ein ähnliches Vorgehen wäre in dem Fall c) zu verfolgen. Hier müssten jedoch nicht eine, sondern in dem Beispiel zwei Tabellen um jeweils eine Zeile ergänzt werden. In beiden Fällen ist es daher problemlos möglich, Änderungen vorzunehmen. Müssen Objekte und Verknüpfungen entfernt werden, so sind lediglich die entsprechenden Zeilen zu löschen. Selbst das Löschen oder Hinzufügen von Auflösungsstufen ist möglich. Wird ein vierter LoD ergänzt, so muss im Fall d) lediglich eine vierte Tabellenspalte ergänzt werden. Im Fall c) müssten drei neue Tabellen angelegt werden, welche die Verknüpfungen LoD0-LoD3, LoD1-LoD3 sowie LoD2-LoD3 beschreiben.

¹⁵PHP Hypertext Preprocessor

Implementierung des Verknüpfungsschemas Um aus den in der Verknüpfungstabelle gespeicherten IDs heraus auf das eigentliche Objekt zurückgreifen zu können, muss der Datenbank der Zusammenhang zwischen dem Objekt und den Werten der Verknüpfungstabelle bekanntgegeben werden. Dies erfolgt durch Definition von sogenannten Primärschlüsseln (primary key) und Fremdschlüsseln (foreign key), welche eine Verknüpfung zwischen dem Objektidentifikator (ID) und einem Attribut einer zweiten Tabellen (z.B. den Link-Tabellen) erlauben. Das als Primärschlüssel dienende Attribut muss in der Datenbank eindeutig sein, um Inkonsistenzen zu vermeiden. Objektorientierte Datenbanken bieten hier die Möglichkeit, den eindeutigen Schlüssel 'OID' zu verwenden. Dieses Attribut bleibt jedoch nach außen hin unsichtbar, solange es nicht explizit abgefragt wird (bei der Abfrage `SELECT * FROM 'table'` wird dieses Attribut nicht mit ausgegeben). Hierdurch sind Probleme bei der Implementierung der Schnittstelle zu erwarten, so dass besser in jeder Geodatentabelle ein zusätzliches Attribut 'gid' zur Identifizierung der Objekte eingeführt wird, welches automatisch vergeben wird, wenn neue Objekte ergänzt werden. Schließlich muss durch Zwangsbedingungen (constraints) in den Fremdschlüsseln sichergestellt werden, dass bei einer Veränderung des Primärschlüssels, vor allen Dingen beim Löschen eines Objektes, auch die entsprechenden Verknüpfungen in den Link-Tabellen verändert oder gelöscht werden, um die Konsistenz der Datenbank zu erhalten.

Das Schema wird schließlich in der Datenbank umgesetzt, indem die Tabellen durch entsprechende SQL-Befehle (`CREATE TABLE`) generiert und Primärschlüssel sowie Fremdschlüssel definiert werden.

5.2.4 Metadatentabelle

Die Metadatentabelle (Abbildung 5.11) soll das Verknüpfungsschema der MRDB nach außen hin wiedergeben. Die

Metadaten						
Repräsentation	Objektart	Objektbereich	ScaleMin	ScaleMax	LinktabelleV	LinktabelleH
GebäudeLoD0	Gebäude	Siedlung	1000	5000	GebäudeLinks	SiedlungLinksLoD0
GebäudeLoD1	Gebäude	Siedlung	5001	10000	GebäudeLinks	SiedlungLinksLoD1
GebäudeLoD2	Gebäude	Siedlung	10001	25000	GebäudeLinks	SiedlungLinksLoD2
GebäudeLoD3	Gebäude	Siedlung	25001	50000	GebäudeLinks	SiedlungLinksLoD3
GebäudeLoD4	Gebäude	Siedlung	50001	100000	GebäudeLinks	SiedlungLinksLoD4
Gebäudeplan	Gebäude	Siedlung	100	500	-1	SiedlungLinksLoD0
OrtslageLoD1	Ortslage	Siedlung	5000	10000	OrtslageLinks	SiedlungLinksLoD1
OrtslageLoD2	Ortslage	Siedlung	10001	25000	OrtslageLinks	SiedlungLinksLoD2
OrtslageLoD3	Ortslage	Siedlung	25001	50000	OrtslageLinks	SiedlungLinksLoD3
OrtslageLoD4	Ortslage	Siedlung	50001	100000	OrtslageLinks	SiedlungLinksLoD4
SiedlungsflächeLoD1	Siedlungsfläche	Siedlung	5000	10000	SiedlungsflächeLinks	SiedlungLinksLoD1
SiedlungsflächeLoD2	Siedlungsfläche	Siedlung	10001	25000	SiedlungsflächeLinks	SiedlungLinksLoD2
SiedlungsflächeLoD3	Siedlungsfläche	Siedlung	25001	50000	SiedlungsflächeLinks	SiedlungLinksLoD3
SiedlungsflächeLoD4	Siedlungsfläche	Siedlung	50001	100000	SiedlungsflächeLinks	SiedlungLinksLoD4
StrassenLoD0	Strasse	Verkehr	1000	5000	StrassenLinks	-1
StrassenLoD1	Strasse	Verkehr	5001	10000	StrassenLinks	-1
StrassenLoD2	Strasse	Verkehr	10001	25000	StrassenLinks	-1
StrassenLoD3	Strasse	Verkehr	25001	50000	StrassenLinks	-1
StrassenLoD4	Strasse	Verkehr	50001	100000	StrassenLinks	-1

Abb. 5.11: MRDB-Metadatentabelle

MRDB umfasst mehrere Objektklassen (z.B. 'GebäudeLoD0', 'GebäudeLoD1'), welche in den Datenbanktabellen organisiert sind. Dabei verkörpert jede Objektklasse eine bestimmte Repräsentation einer bestimmten Objektart. Diese Zusammenhänge müssen zunächst durch die Metadatentabelle aufgezeigt werden. Daher werden die Spalten 'Repräsentation' und 'Objektart' eingefügt. So ist in der Abbildung 5.11 abzulesen, dass die Objektart 'Gebäude' durch die Objektklassen 'GebäudeLoD0', 'GebäudeLoD1', 'GebäudeLoD2', 'GebäudeLoD3' und 'GebäudeLoD4' repräsentiert wird. Da innerhalb derselben Auflösungsstufe zusätzlich Objekte desselben Objektbereiches verknüpft werden sollen, ist es wichtig, zu wissen, zu welchem Objektbereich (Siedlung, Verkehr) die jeweilige Objektklasse zu zählen ist. Schließlich bilden die Spalten 'ScaleMin' und 'ScaleMax' einen Parameter zur Unterscheidung der einer Objektart zugeordneten, alternativen Repräsentationen. Jede Repräsentation vertritt somit eine Objektart für einen bestimmten Maßstabsbereich. Die Spalte 'LinktabelleV' gibt Auskunft darüber, in welcher Tabelle Informationen zu der vertikalen Verknüpfung, also zwischen den einzelnen Auflösungsstufen, zu finden sind. So sind die Verknüpfungsinformationen der Klasse 'GebäudeLoD0' in der Linktabelle 'GebäudeLinks' zu finden. Auf die gleiche Weise sind auch Informationen zu den horizontalen Verknüpfungen in der Tabelle 'LinktabelleH' zu finden.

Soll beispielsweise zu dem Objekt GebäudeLoD2.100 eine höher aufgelöste Darstellung aus der MRDB abgefragt werden, so wird zunächst die 'Repräsentation' zu dem Objekt, hier 'GebäudeLoD2', in der Metadatentabelle gesucht. Es lässt sich ablesen, dass dieses Objekt zu der Objektart 'Gebäude' zu zählen ist. Anschließend wird mit Hilfe von 'ScaleMin' und 'ScaleMax' eine Repräsentation gesucht, welche der gewünschten Auflösung entspricht. Wird eine Repräsentation für den Maßstab 1:5k gesucht, so sucht die Applikation eine Zeile, deren Werte zwischen 'ScaleMin' und 'ScaleMax' dem Wert 5000 entsprechen. Die Zeile enthält den Hinweis auf die Objektklasse 'GebäudeLoD0'. Anschließend wird in der Zeile zu 'GebäudeLoD2' der Name der Verknüpfungstabelle ('GebäudeLinks') gesucht. In dieser Tabelle 'GebäudeLinks' finden sich nun in der Spalte 'GebäudeLoD0' die IDs des mit dem Objekt '100' verknüpften, hochauflösenden Objektes (z.B. '1,2,3,4,5').

Durch die Speicherung aller relevanten Informationen in einer Metadatentabelle ist die Erweiterung der MRDB um zusätzliche Repräsentationen oder Objektarten problemlos möglich. Werden zusätzliche Repräsentationen in die Datenbank eingespielt, so gilt es, lediglich der Metadatentabelle eine Zeile mit den entsprechenden Einträgen zuzufügen. Widersprüche in Folge sich überlappender Maßstabsbereiche werden durch Anpassung der Werte 'ScaleMin' und 'ScaleMax' entsprechend nivelliert.

Somit ergibt sich das in Abbildung 5.12 aufgezeigte Datenbankschema.

5.3 Datengenerierung und -integration

Die Datenintegration stellt den nächsten logischen Schritt im Aufbau einer MRDB dar. An dieser Stelle gilt es, die vorhandenen Daten auf das Datenbankschema zu übertragen. Der Prozess der Datenintegration kann in drei Teilschritte aufgegliedert werden. Zunächst müssen die vorhandenen Datensätze in der Datenbank abgelegt werden. Da das Datenbankschema entsprechend den vorhandenen Datensätzen gewählt wurde, ist hier keine zusätzliche Transformation notwendig. Gleichzeitig wird davon ausgegangen, dass jeder Datensatz, welcher in die Datenbank eingepflegt wird, für sich folgerichtig und konsistent ist. Somit erübrigt sich an dieser Stelle eine Prüfung der Konsistenz. In dem nächsten Schritt müssen fehlende Datensätze, also Auflösungsstufen, ergänzt werden. Grundsätzlich ist davon auszugehen, dass die Daten mindestens in ihrer höchsten Auflösung vorliegen. Geringere Auflösungen können hieraus durch Generalisierungsverfahren abgeleitet werden. Eine Ableitung von geringer zu höher aufgelösten Datensätzen ist ohne weitere Annahmen zu den Eigenschaften eines Objektes nicht möglich. Fehlende Daten können in diesem Fall nur durch eine zusätzliche Datenerfassung ergänzt werden.

In einem letzten Schritt gilt es schließlich, die Verknüpfungen zwischen den unterschiedlichen Repräsentationen zu generieren, wodurch die Konsistenz der Datensätze untereinander gesichert wird. Hier sind grundsätzlich zwei Wege denkbar. Vorhandene Daten werden durch geeignete Matchingverfahren (vgl. Unterabschnitt 3.4.5) miteinander in Beziehung gesetzt. Neu zu erzeugende Daten können auf die gleiche Weise mit den vorhandenen Daten verknüpft werden oder es wird schon während des Generalisierungsprozesses das Wissen über die Verknüpfungen zwischen den hochauflösenden und den generalisierten Objekten in der Datenbank abgelegt. Der zweite genannte Fall bietet den Vorteil, dass hier keine Fehlinterpretationen auftreten können, wie es beim Matching unter Umständen möglich ist. Während der Generalisierung werden alternative Repräsentationen aus vorhandenen erzeugt und somit kann die Verknüpfung zwischen beiden jeweils bei der Speicherung der neu erzeugten Objekte mit angelegt werden. Dieser Aufwand ist außerdem wesentlich geringer, als die Implementierung und Anwendung eines zusätzlichen Matchingverfahrens.

Beide Fälle, also die Verknüpfung vorhandener Datensätze durch Matchingverfahren sowie die Erzeugung alternativer Repräsentationen durch automatisierte Generalisierungsverfahren inklusive der Ableitung der Beziehungen aus diesem Prozess, werden im Rahmen der vorliegenden Arbeit exemplarisch umgesetzt und in Unterabschnitt 5.3.3 beschrieben.

5.3.1 Die Arbeitsumgebung: Java, JTS, JUMP

Bevor die einzelnen Implementierungen bezüglich Datenmatching und Generalisierung erläutert werden, soll in diesem Abschnitt die zur Implementierung verwendete Arbeitsumgebung kurz vorgestellt werden. Diese wird des Weiteren zur Umsetzung der in Kapitel 7 dargestellten Servlets der Applikationsebene sowie als potenzielle Client-Schnittstelle (vgl. Kapitel 8) eingesetzt.

Die im Folgenden dargestellten Entwicklungen beruhen ausschließlich auf der Programmiersprache *Java*. Um räumliche Objekte bearbeiten, manipulieren und visualisieren zu können, wird auf die Bibliotheken des freien

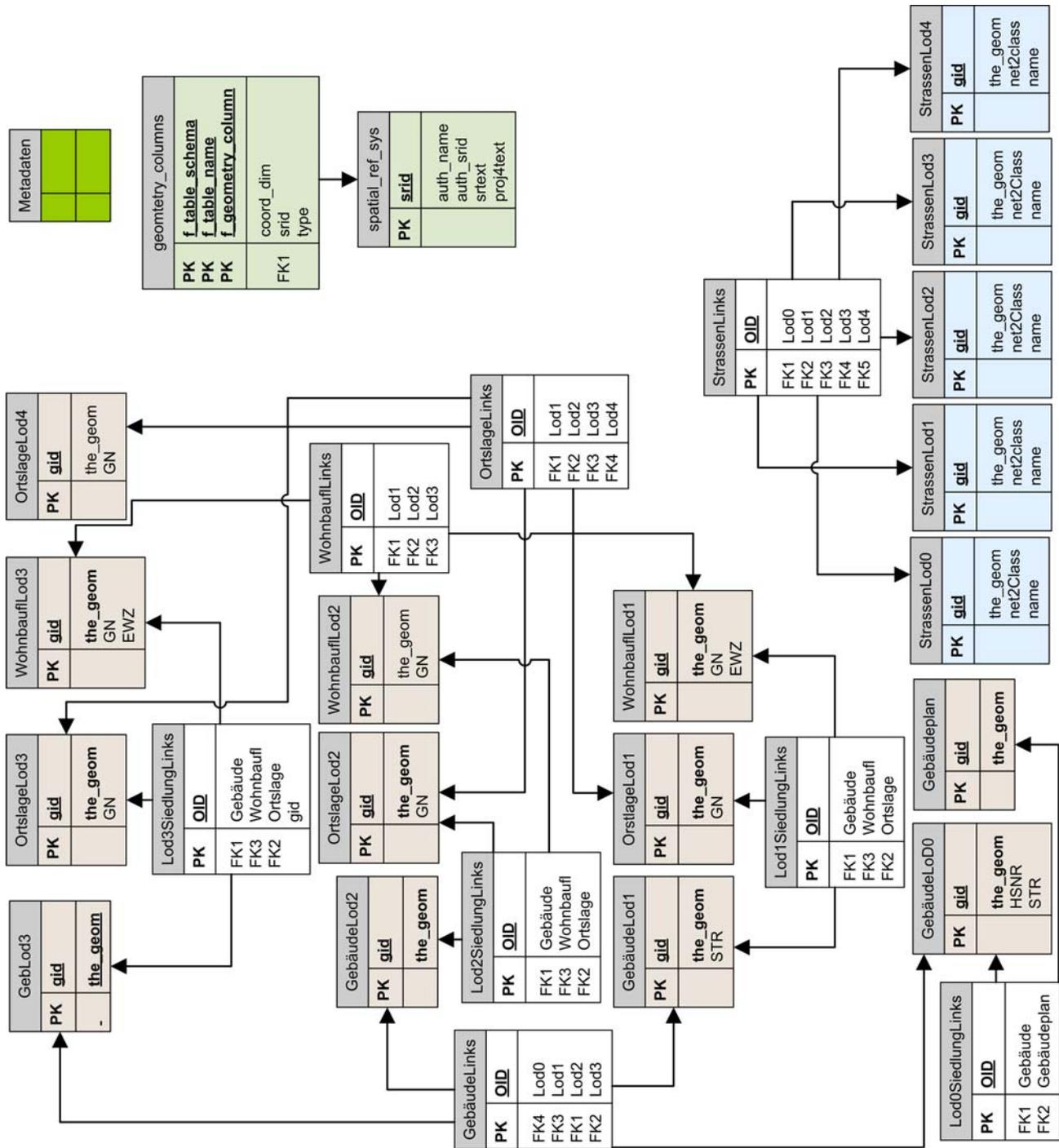


Abb. 5.12: Datenbankschema der MRDB: Datentabellen Siedlung (braun), Verkehr (blau), Link-Tabellen (weiß), Metadaten (grün)

und als Open Source deklarierten Projektes *JUMP*¹⁶ zurückgegriffen. Hinter diesem Begriff verbirgt sich eine Sammlung freier Bibliotheken und erweiterbarer APIs¹⁷ sowie ein grafisches Interface (siehe Abbildung 5.13). Die API ist dreigeteilt und besteht aus der JTS¹⁸, JCS¹⁹ und JUMP²⁰. JTS bildet die Grundlage des Systems und bietet geometrische Objekttypen, Funktionalitäten sowie geometrische Operationen, welche allesamt konform zu dem räumlichen Objektmodell des OGC (OGC 1999) sind. JTS stellt somit eine komplette, konsistente und robuste Implementierung grundlegender räumlicher 2D Algorithmen zur Verfügung. Die JCS stellt darüber hinaus Funktionalitäten zur Verknüpfung und Anpassung benachbarter Datensätze bereit.

JUMP beinhaltet schließlich eine erweiterbare GIS-Oberfläche zur Visualisierung und Bearbeitung räumlicher Daten. Durch eine entsprechende API können eigene Applikationen zur Bearbeitung der räumlichen Daten in die Oberfläche integriert werden. So werden beispielsweise Generalisierungsfunktionalitäten integriert und deren Ergebnisse visualisiert, bearbeitet und über eine JDBC²¹ Schnittstelle in der Datenbank abgelegt. JUMP bietet die Möglichkeit, über diese Schnittstelle mit der *PostGIS/PostgreSQL* Datenbank zu kommunizieren, um Daten auszulesen oder Geometrien in der Datenbank abzulegen.

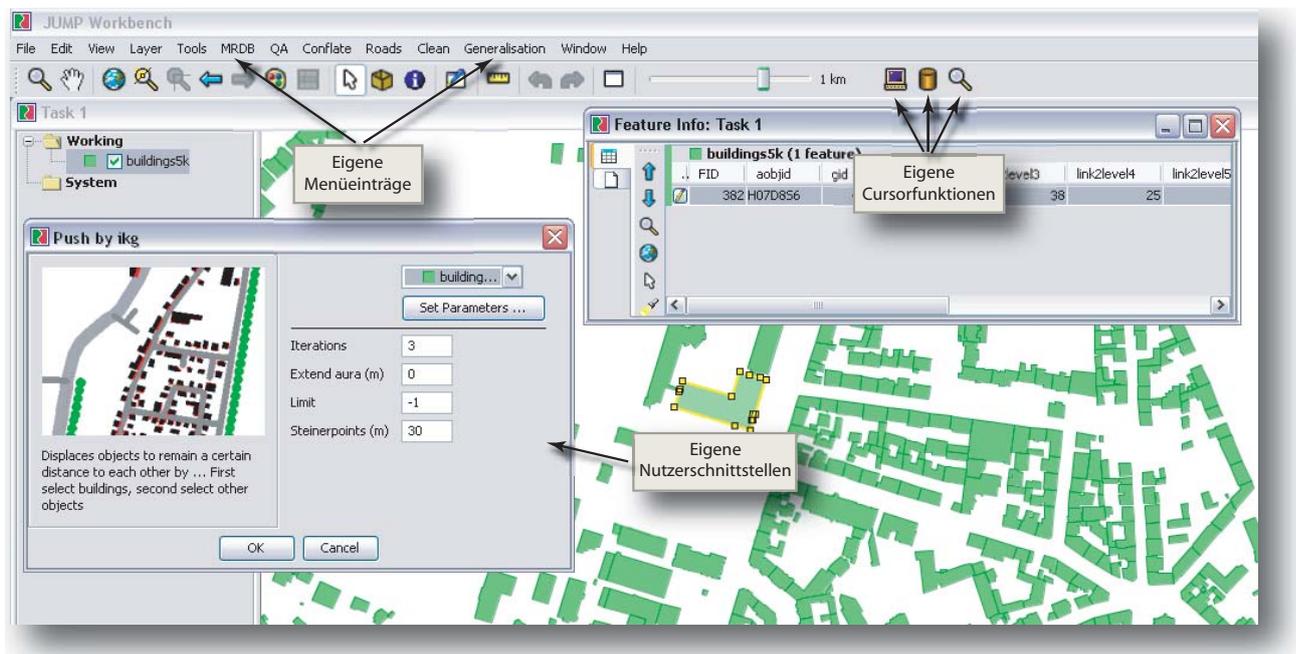


Abb. 5.13: *JUMP-GUI: Möglichkeit zur Visualisierung geometrischer Datensätze sowie Integration eigener Funktionalitäten zur Bearbeitung der Daten in die Oberfläche*

5.3.2 Generalisierung der Datensätze

Die Darstellung des konzeptionellen Schemas (Abbildung 5.9) lässt erkennen, dass einige Auflösungsstufen nicht vorhanden sind und durch entsprechende Generalisierungsverfahren erzeugt werden müssen. Hierzu wird zum Teil auf vorhandene Verfahren zurückgegriffen, fehlende Prozeduren sollen durch eigene Implementierungen ergänzt werden. Darüber hinaus wird auf diese Weise der Weg zur Ableitung der Verknüpfungen direkt aus dem Generalisierungsprozess aufgezeigt und untersucht. Des Weiteren werden die automatisch ablaufenden Prozesse im späteren Verlauf der Arbeit dazu genutzt, ein System zur Online-Generalisierung zu simulieren, um dieses dann mit dem Konzept einer MRDB vergleichen zu können (vgl. Kapitel 7).

Eine Einschränkung muss den nachfolgend beschriebenen Verfahren an dieser Stelle zugesprochen werden. Die einzelnen Verfahren werden jeweils isoliert genutzt und in keiner Weise innerhalb eines Generalisierungs-Workflows

¹⁶<http://www.jump-project.org>

¹⁷Application Programming Interfaces

¹⁸JTS Topology Suite

¹⁹JCS Conflation Suite

²⁰JUMP Unified Mapping Platform

²¹Java Database Connectivity

miteinander kombiniert. Auch die einzelnen Objektarten werden zum größten Teil für sich betrachtet, ohne Rücksicht auf umliegende Objekte anderer Objektarten. Somit sollen die hier vorgestellten Verfahren nicht als ein Satz zur automatisierten Generalisierung vollständiger Datensätze betrachtet werden, sondern es besteht vielmehr der Anspruch, exemplarisch einige Generalisierungsschritte zu implementieren, um die zuvor beschriebenen Punkte behandeln zu können.

Entsprechend dem konzeptionellen Schema gilt es, die Objektarten 'Gebäude', 'Ortslage', 'Siedlungsfläche' und 'Straßennetzwerk' aus den Basisdaten für die Maßstäbe 1:10k, 1:25k, 1:50k sowie 1:100k zu generieren. Hierzu werden Generalisierungsverfahren zur Gebäudevereinfachung, -zusammenfassung, -typifizierung, Verdrängung, Flächengeneralisierung, Auswahl sowie Linienvereinfachung genutzt. Zur Durchführung der Verfahren kann teilweise auf bereits vorhandenen Programmcode zurückgegriffen werden.

Gebäudevereinfachung

Die Gebäudevereinfachung verändert die Kontur einer Gebäudegeometrie, indem Details, deren Größe unterhalb einer definierten Minimaldimension liegt, eliminiert werden.

Ansatz Der hier verwendete Ansatz wurde entwickelt von Sester (2000) und ist in der zitierten Literatur ausführlich beschrieben. Der im weiteren Verlauf beschriebene Prozess umfasst dabei nur einen Teil der vorgeschlagenen Generalisierungsprozedur. In einem zweiten Schritt wird das Ergebnis durch einen Ausgleichungsansatz verbessert, indem das generalisierte Gebäude der Originalgeometrie angenähert wird. Auf diesen zweiten Schritt soll in der hier beschriebenen Implementierung verzichtet werden.

Es werden in einem ersten Schritt zunächst alle Kanten einer Gebäudegeometrie auf ihre Länge hin untersucht. Unterschreitet eine Kante die vorgegebene Minimallänge, so wird in einem nächsten Schritt bestimmt, welchen der in Abbildung 5.14 aufgezeigten Fälle diese Kante beschreibt. Die Idee beruht auf der Annahme, dass jede

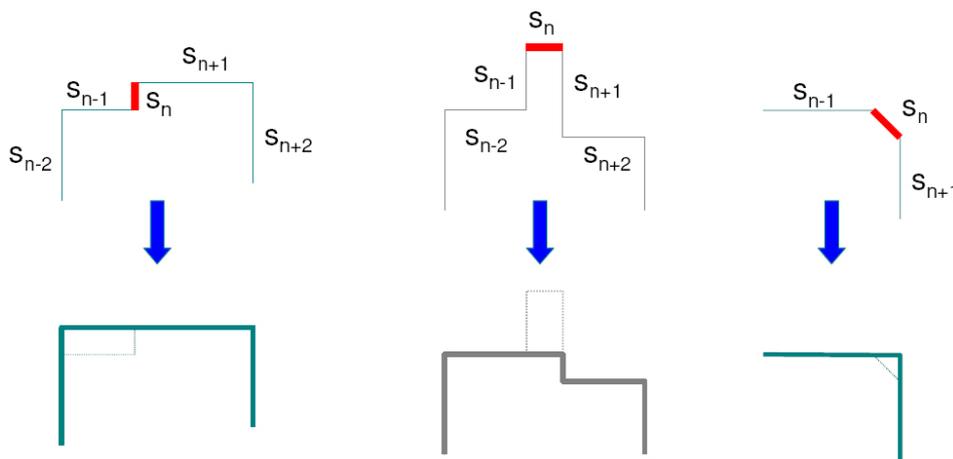


Abb. 5.14: Gebäudevereinfachung durch eliminieren zu kurzer Kanten in den Fällen Versatz (links), Vorsprung (Mitte), Ecke (rechts) (aus Sester 2000)

Situation der Gebäudevereinfachung auf einen der dargestellten Fälle übertragen werden kann. Welcher Fall aktuell vorliegt, wird durch die Untersuchung der Richtungen der benachbarten Kanten bestimmt. Jeder der drei Fälle löst dann eine entsprechende Folgeaktion aus (vgl. Abbildung 5.14):

1. Vorgänger- (S_{n-1}) und Folgekante (S_{n+1}) besitzen die gleiche Richtung \Rightarrow Fall a (Versatz): Die Kante S_n wird eliminiert, außerdem:
 - a) S_{n-1} ist kürzer und wird ebenfalls eliminiert $\Rightarrow S_{n+1}$ verbleibt, wird verlängert und mit der verbleibenden Vorgängerkante S_{n-2} verschnitten oder
 - b) S_{n+1} ist kürzer und wird ebenfalls eliminiert $\Rightarrow S_{n-1}$ verbleibt, wird verlängert und mit der verbleibenden Folgekante S_{n+2} verschnitten
2. Vorgänger- (S_{n-1}) und Folgekante (S_{n+1}) verlaufen in entgegengesetzter Richtung \Rightarrow Fall b (Aus- oder Einbuchtung): Die Kante S_n wird eliminiert, außerdem:
 - a) S_{n+1} ist kürzer und wird ebenfalls eliminiert $\Rightarrow S_{n+2}$ wird mit S_{n-1} verschnitten oder
 - b) S_{n-1} ist kürzer und wird ebenfalls eliminiert $\Rightarrow S_{n-2}$ wird mit S_{n+1} verschnitten

3. Vorgänger- (S_{n-1}) und Folgekante (S_{n+1}) bilden einen rechten Winkel \Rightarrow Fall c (Ecke): Die Kante S_n wird eliminiert, Vorgänger- (S_{n-1}) und Folgekante (S_{n+1}) werden verschritten

Implementierung und Bewertung Um entscheiden zu können, ob Kanten in die gleiche oder entgegengesetzte Richtung verlaufen oder einen rechten Winkel bilden, werden die Differenzen der Richtungswinkel bestimmt. Dabei müssen die Winkeldifferenzen nicht exakt 0° , 90° oder 180° aufweisen. Vielmehr werden diese um einen gewissen Toleranzbereich erweitert. Des Weiteren gilt es, zuvor Punkte in einer nahezu gerade verlaufenden Linie zu eliminieren, da diese ansonsten in den Fällen a) und b) Schnitte zweier nahezu parallel verlaufenden Linien verursachen.

Nachdem zu kurze Kanten einer Geometrie eliminiert wurden, muss die neu gebildete Geometrie erneut auf zu kurze Kanten hin untersucht und eventuell angepasst werden. Um zu verhindern, dass Gebäude mit unerlaubter Geometrie, zum Beispiel durch unerlaubte Selbstschnitte, entstehen, wird die gesamte Geometrie nach der Vereinfachung auf Konformität hin überprüft. Gebäude mit weniger als vier verbleibenden Kanten oder einer zu geringen Fläche werden ebenfalls eliminiert. Die Abbildung 5.15 (unten) zeigt Ergebnisse des Verfahrens und



Abb. 5.15: Ergebnisse der Konturenvereinfachung. Oben: Originaldatensatz (links) und Datensatz nach der Generalisierung mit Schwellwert 3,5 m (rechts); Unten: Originaldatensatz (links) und Datensatz nach der Generalisierung mit Schwellwert 10 m, eine Lückenbildung ist erkennbar (rechts)

deckt einen Nachteil auf: Durch das Eliminieren von Gebäudeteilen oder ganzen Gebäuden entstehen Lücken im Datenbestand. Gebäude, welche vorher eine Häuserreihe bildeten, stellen sich nun unter Umständen als einzeln stehende Objekte dar. Dieser Effekt ist aus kartografischer Sicht nicht korrekt. In dicht bebauten Gebieten sollte daher entweder vor der Konturenvereinfachung eine Verschmelzung angrenzender Objekte erfolgen oder aber die entstandenen Lücken sollten nach der Generalisierung wieder geschlossen werden. Des Weiteren kann es passieren, dass benachbarte Gebäude nach der Generalisierung überlappende Geometrien besitzen. Aus demselben Grund können Gebäude, welche vorher eine Linie bildeten, nach der Generalisierung Versätze aufweisen, so dass auch hier eine manuelle Nachbearbeitung erforderlich ist.

Um zu vermeiden, dass Objekte, welche sehr lang und gleichzeitig sehr schmal sind, eliminiert werden, kann als zusätzlicher Parameter eine minimale Fläche angegeben werden. Objekte beziehungsweise Objektteile, welche größer sind, werden nicht eliminiert, sondern bleiben durch Verlängern der Kante erhalten.

Alle genannten Probleme können durch manuelle Nachbearbeitung gelöst werden. Ein automatisches Verfahren müsste zur Lösung der Konflikte jeweils die Lage der Nachbargebäude bei der Generalisierung mit berücksichtigen, damit keine Lücken, Überlappungen oder Versprünge entstehen. Des Weiteren kann bei der Generalisierung neben der geometrischen auch die semantische Information, in diesem Fall beispielsweise der Gebäudetyp, also Wohnhaus oder öffentliches Gebäude, und daraus folgend die Wichtigkeit einzelner Objekte berücksichtigt werden. Hierdurch wird vermieden, dass wichtige Objekte gelöscht werden.

Für Papierkarten gilt eine Minimaldimension von 0,25 mm für Vorsprünge und Kanten (siehe Tabelle 5.1). Für Bildschirmkarten ergeben sich an dieser Stelle weitere Einschränkungen, da die Lesbarkeit von Bildschirmkarten

weiterhin von dem Typ, der Auflösung und Größe des Bildschirms sowie daraus folgend der Größe der Pixel und der Orientierung der angezeigten Objekte abhängt. Bollman & Koch (2002) empfehlen für Bildschirmkarten daher die Vergrößerung der Minimaldimensionen um den Faktor zwei bis vier, verglichen mit Papierkarten. Somit ergibt sich eine minimale Größe der Vorsprünge und Kanten von 0,5 mm.

Neben der neuen Geometrie wird auch die Verknüpfung zwischen den beiden Geometrien in der Datenbank abgelegt. Diese zusätzliche Funktion lässt sich sehr einfach in den Generalisierungsalgorithmus integrieren, da lediglich ergänzende Informationen in die Datenbank geschrieben werden müssen.

Neben dem hier vorgestellten Programm existieren weitere Produkte zur automatischen Vereinfachung von Gebäudegeometrien, wie das Programmsystem CHANGE der Universität Hannover. Da die Intention des beschriebenen Algorithmus jedoch unter anderem darin lag, die Ableitung der Verknüpfungen in den Generalisierungsprozess zu integrieren, wurde das Programm CHANGE lediglich dazu genutzt, die Generalisierungsergebnisse des eigenen Verfahrens zu validieren.

Zusammenfassung von Gebäuden

Unterschreitet die Distanz zweier Objekte einen bestimmten Wert, so gilt es, diesen Abstand entweder auf ein bestimmtes über der Minimaldimension liegendes Maß zu erhöhen, indem Objekte verschoben werden, oder die Lücke zu eliminieren, indem die benachbarten Objektgeometrien miteinander verschmolzen werden. Für den letztgenannten Fall soll das im Folgenden beschriebene Verfahren genutzt werden.

Verfahren Der Ansatz besteht aus vier Schritten. Zunächst werden alle Objekte um den halben Betrag des Schwellwertes vergrößert. Anschließend wird geprüft, ob sich nun Überlappungen zwischen einzelnen Objekten ergeben haben. In dem nächsten Schritt werden alle sich überlappenden Geometrien zu einer einzelnen verschmolzen. Danach wird die Vergrößerung durch eine Verkleinerung um denselben Betrag wieder zurückgenommen. Abbildung 5.16 verdeutlicht dieses Verfahren. Dieses Vergrößern und anschließende Verkleinern bewirkt zusätzlich eine Vereinfachung der Geometrien (siehe Abbildung 5.16). Diese morphologischen Operationen stammen ursprünglich aus dem Bereich der Segmentierung von Rasterbildern und werden von Umbaugh (1998) wie folgt beschrieben: „Dilatation erlaubt Objekten sich auszudehnen, wodurch potenziell kleine Löcher gefüllt und getrennte Objekte verbunden werden. Erosion schrumpft Objekte durch wegradieren (erodieren) ihrer Ränder. Diese beiden grundlegenden Operationen, Dilatation und Erosion, können zu komplexeren Sequenzen kombiniert werden. *Schließen* besteht aus einer Dilatation gefolgt von einer Erosion und kann dazu benutzt werden, um Löcher und kleine Lücken zu füllen.“ Das letztgenannte Prinzip des *Schließens* wird hier verwendet, um kleine Freiflächen in den Gebäudegeometrien, wie zum Beispiel Innenhöfe, zu eliminieren (s. Abbildung 5.17).

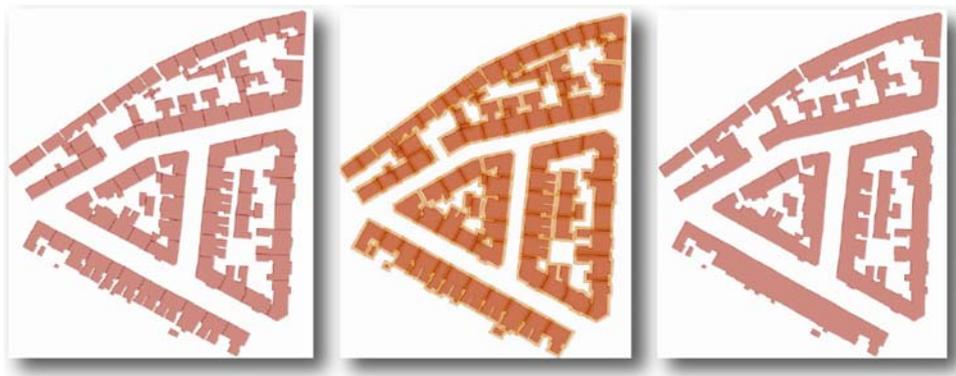


Abb. 5.16: Schritte der Gebäudeverschmelzung: Ausgangsdatensatz (links), Pufferbildung (Mitte), Verschmelzung und Schrumpfen (rechts)

Implementierung und Bewertung Durch unterschiedliche geometrische Operationen werden aus einer oder mehreren Geometrien neue gebildet. In diesem Fall wird zunächst durch eine Pufferfunktion (buffer) jede der vorliegenden Gebäudegeometrien um einen bestimmten Betrag nach außen hin vergrößert. Der Puffer ist definiert als die Geometrie, welche „[...] alle Punkte repräsentiert, deren Abstand zu der ursprünglichen Geometrie kleiner oder gleich d ist“ (OGC 1999).

In einem nächsten Schritt gilt es nun, die neu entstandenen Geometrien auf ihrer gegenseitige Lage hin zu untersuchen. In dem hier beschriebenen Fall, in dem zwei Gebäudeflächen betrachtet werden, gilt es herauszufinden, ob nach der Vergrößerung einer der Fälle *equals*, *touches* oder *overlaps* auftritt.

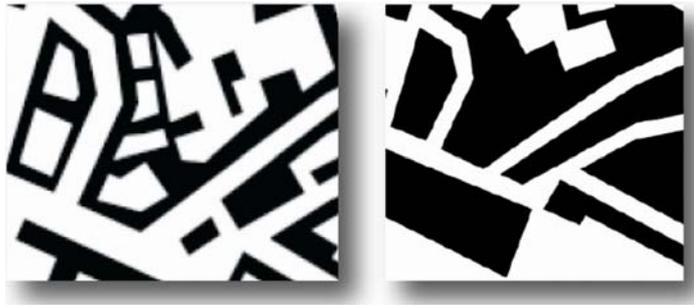


Abb. 5.17: Schließen von Lücken als Folge der Dilatation und Erosion

- *Equals*: Geometrien sind räumlich identisch
- *Touches*: Geometrien haben einen Randpunkt gemeinsam, aber keinen im Inneren
- *Overlaps*: Geometrien teilen einige, aber nicht alle Punkte und die Schnittfläche hat dieselbe Dimension, wie die Geometrien

Um herauszufinden, ob einer dieser Fälle vorliegt, müssen theoretisch alle Objekte mit allen verglichen werden, was einen enormen Zeit- und Rechenaufwand bedeutet. Die Zahl möglicher Kombinationen ergibt sich aus $\frac{1}{2} * (n^2 - n)$. In dem dargestellten, 2 km x 4 km großen Testgebiet mit 1.250 Gebäuden ergeben sich insgesamt 780.625 Vergleichsoperationen. Um deren Zahl zu minimieren, wird die Tatsache genutzt, dass Objekte niemals über eine Straße hinweg miteinander verschmolzen werden dürfen. Somit müssen nur die Objekte miteinander verglichen werden, welche sich innerhalb einer Straßenmasche befinden. Daher werden zunächst aus dem aus Linienzügen bestehenden Straßennetz mit Hilfe eines Polygonisierungsverfahrens flächenhafte Objekte gebildet (Abbildung 5.18), um auf dieser Basis Gebäude gruppieren zu können, welche verschmolzen werden dürfen. Eine Gruppe entsteht aus Gebäuden, welche sich innerhalb einer Polygonfläche befinden. Bei 50 Flächen und

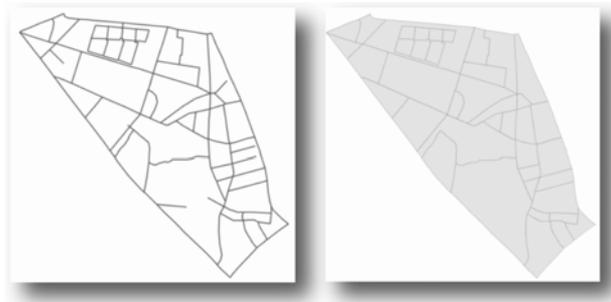


Abb. 5.18: Polygonisierung des Straßennetzwerkes: Ausgangsdaten, Linienzüge (links); resultierende Polygonflächen (rechts)

einer durchschnittlichen Dichte von 25 Gebäuden pro Fläche ergeben sich 300 Vergleichsoperationen pro Fläche, insgesamt also 15.000. Hinzu kommt eine Vorverarbeitung, um jedes Gebäude einer Fläche zuordnen zu können. Hier sind maximal 62.500 ($1250 * 50$) Vergleiche durchzuführen. Dieser Prozess der Zuordnung kann ausgelassen werden, falls in der Datenbank bereits Verknüpfungen zwischen Siedlungsflächen und Gebäuden bestehen. Diese Information kann dazu genutzt werden, nur Gebäudegeometrien zu verschmelzen, welche mit derselben Siedlungsfläche verknüpft sind. Dieser Vorteil wird teilweise kompensiert durch den Zeitverlust auf Grund der Kommunikation mit der Datenbank. Mit einer steigenden Zahl möglicher Verknüpfungskombinationen steigt jedoch der Vorteil, diese Informationen aus der MRDB abzufragen. Des Weiteren wird weniger Rechenleistung zur Generalisierung erforderlich, so dass diese oder ähnliche Prozesse auch auf der Client-Seite, beispielsweise auf mobilen Geräten, ausgeführt werden können.

Wird eine Überlagerung der Geometrien festgestellt, so wird durch die geometrische Funktion *Union* aus den sich überlappenden Geometrien eine einzige gebildet. „Union bildet eine Geometrie, welche die vereinigte Punktmenge beider Geometrien darstellt“ (OGC 1999). Anschließend wird die Vergrößerung wieder umgekehrt, indem ein Puffer mit negativem Abstand desselben Betrages wie bei der Vergrößerung gebildet wird.

Nach der Tabelle 5.1 gelten für Papierkarten Minimalabstände von 0,20 mm. Entsprechend den Aussagen des vorangegangenen Abschnittes muss für digitale Karten diese Dimension mindestens um den Faktor zwei erhöht werden, so dass sich Mindestabstände von 0,40 mm ergeben. Somit ergeben sich für den Maßstab 1:10k der

Parameter 4 m, bei 1:25k 10 m. Der Prozess der Verschmelzung wird der Gebäudevereinfachung vorangestellt, um zu vermeiden, dass künstliche Lücken zwischen den Gebäuden durch die Vereinfachung entstehen. Die Kombination aus Dilatation und Erosion wird darüber hinaus genutzt, um in dicht bebauten Gebieten ab dem Maßstab 1:50k Löcher zu füllen, wie es Abbildung 5.17 zeigt. Außerdem werden durch diese Vorgehensweise die Konturen der Objekte vereinfacht.

Auch bei diesem Verfahren wird neben den resultierenden Geometrien zusätzlich die Information über die Verknüpfungen zwischen den Ausgangsdaten und den resultierenden Geometrien in der Verknüpfungstabelle der Datenbank abgelegt.

Auswählen

Welche Objekte während der Generalisierung eliminiert werden und welche erhalten bleiben, hängt von diversen geometrischen und semantischen Faktoren ab. Geometrische Faktoren umfassen unter anderem die Größe, respektive den Flächeninhalt, oder Ausdehnung eines Objektes sowie deren Nachbarschaften. Einzelne stehende Objekte bleiben eher erhalten als Teile einer Gruppe. Semantische Informationen reflektieren die Bedeutung der Objekte. Gebäude können unterteilt werden in öffentliche und nicht-öffentlich, Straßensegmente werden einer bestimmten Kategorie zugeordnet, welche hierarchisch gegliedert sind.

Zur Generalisierung der Siedlungsflächen werden Polygone, welche eine bestimmte Größe unterschreiten, einer Nachbarfläche zugeschlagen, sofern diese nicht durch eine Straße voneinander getrennt sind.

Das Straßennetzwerk wird bei kleiner werdendem Maßstab stückweise reduziert. Hierzu greift der Algorithmus auf das Attribut 'Netclass2' zurück, um auf diese Weise die Straßensegmente geringerer Kategorie schrittweise zu eliminieren (siehe Abbildung 5.19).



Abb. 5.19: Generalisierung des Straßennetzwerkes durch Auswählen bestimmter Straßenkategorien

Darüber hinaus werden die in dem folgenden Absatz aufgezeigten Verfahren verwendet, um deren Linienverläufe zu vereinfachen.

Typifizierung, Verdrängung und Liniengeneralisierung

Zusätzlich zu den oben genannten Generalisierungsverfahren sollen die in dem folgenden Abschnitt beschriebenen Algorithmen zur Generalisierung angewendet werden. Diese bestehenden Generalisierungslösungen sollen ebenfalls zur Produktion unterschiedlich aufgelöster Datensätze genutzt werden. Da der Quellcode dieser Verfahren nicht verändert werden kann, stehen hier lediglich die Ausgangsdaten und der Ergebnisdatensatz zur Verfügung, so dass die Verknüpfungen der unterschiedlichen Auflösungsstufen nachträglich bestimmt werden müssen.

Eine Typifizierung wird notwendig, wenn die Anzahl bestimmter Objekte, in der Regel Gebäude, auf Grund von Platzmangel reduziert werden muss auf eine bestimmte Untermenge. Gleichzeitig werden die verbleibenden Objekte entsprechend vergrößert und so angeordnet, dass diese die typische Siedlungsstruktur wiedergeben. Während in dicht bebauten Gebieten Lücken zum Beispiel durch die oben beschriebenen Verfahren der Dilatation und Erosion geschlossen werden, so dass einzelne Baublöcke entstehen, wird die Typifizierung eher in Gebieten lockerer Bebauung eingesetzt. Das Testgebiet Hannover entspricht einer geschlossenen Bauweise, so dass hier keine Typifizierung verwendet werden muss. Trotzdem soll an dieser Stelle auch die Typifizierung in die Reihe



Abb. 5.20: Testdaten Flensburg als Gebiet lockerer Bebauung

der zum Aufbau einer MRDB notwendigen Generalisierungsverfahren mit aufgenommen werden. Daher wird an dieser Stelle ein alternatives Testgebiet mit lockerer Bebauung eingeführt (Abbildung 5.20).

Hierzu soll ein von Sester (2001) entwickeltes und implementiertes Verfahren (Typify) eingesetzt werden. Innerhalb der JUMP-GUI wird eine Schnittstelle zum Aufruf des Algorithmus sowie zur Abfrage der Parameter entwickelt. Aus derselben Quelle stammt ein Verfahren zur Verdrängung konkurrierender Objekte (Push), in dem vorliegenden Fall von Straßen und Gebäuden. Die Ergebnisse von Typifizierung und Verdrängung sind in Abbildung 5.21 zu erkennen.



Abb. 5.21: Typifizierung eines Testgebietes (1500m x 1500m) (oben): Verdrängung und Typifizierung für Zielmaßstab 1:10k (unten links); Zielmaßstab 1:50k (unten rechts) (aus Sester et al. (2004))

Da die Verdrängung keinen Einfluss auf die Verknüpfungen hat, kann die Zuordnung der Objekte vor der Verdrängung erfolgen, beispielsweise zu dem Zeitpunkt der Zusammenfassung oder Konturenvereinfachung. Die Typifizierung dagegen eliminiert einzelne Objekte. Somit muss nach der Typifizierung ein Matchingverfahren zur Bestimmung der Korrespondenzen durchgeführt werden.

Des Weiteren gilt es, das Straßennetzwerk nicht nur auszudünnen, sondern auch den Linienverlauf entsprechend dem jeweiligen Maßstab zu vereinfachen. Hierzu soll das Douglas-Peucker-Verfahren zur Linienvereinfachung (Douglas & Peucker 1973) verwendet werden. Die Verknüpfung der Objekte erfolgt auf Basis der Geometrie, in diesem Fall also dem Linienzug, nicht jedoch auf Basis einzelner Liniensegmente. Da auch die Linienvereinfachung und -glättung für jeden Linienzug, also von Kreuzungspunkt zu Kreuzungspunkt (vgl. Unterabschnitt 5.2.2),

separat durchgeführt wird, hat diese Generalisierung keinen Einfluss auf die Verknüpfungen. Somit kann, ähnlich der Gebäudevereinfachung und Verdrängung, jedem vereinfachten Linienzug der ursprüngliche Linienzug als Partner zugeordnet werden.



Abb. 5.22: *Linienvereinfachung nach Douglas-Peucker*

5.3.3 Verknüpfung vorhandener Datensätze durch Matching

An dieser Stelle ist die MRDB mit allen notwendigen Repräsentationen gefüllt. Allerdings sind nicht alle Objekte in das System integriert. Einige Datensätze existieren von Beginn an nebeneinander, wie zum Beispiel die Gebäude und Siedlungsflächen. Werden zur Generalisierung Softwareprodukte eingesetzt, welche innerhalb des Generalisierungsprozesses keinen Zugriff auf die einzelnen Elemente erlauben, so liegen die generalisierten Daten zunächst auch getrennt von den Ausgangsdaten vor. Die fehlenden Beziehungen sollen durch entsprechende Matchingverfahren hergestellt werden. Des Weiteren können nachträglich neue Objektarten oder Auflösungsstufen hinzugefügt werden, welche dann durch geeignete Prozeduren zu integrieren sind.

Unterabschnitt 3.4.5 beschreibt die Möglichkeiten und Bedingungen zur Verknüpfung von Objekten. Diese erfolgt für räumliche Daten größtenteils über geometrische oder semantische Verfahren. Bei den geometrischen Methoden können Abstände, Lagebeziehungen, bei linearen Objekten Winkelinformationen und bei linien- oder flächenhaften Objekten Formmerkmale verglichen werden.

Matching Straßennetzwerke unterschiedlicher LoDs

Die Zuordnung der einzelnen Straßensegmente erfolgt nach der Ausdünnung und vor der Linienvereinfachung. Die Ausdünnung bedeutet ein einfaches Löschen einiger Objekte, die übrigen werden übernommen, so dass hier eine einfache 1:1 Zuordnung in der Datenbank erfolgen kann. Diese Zuordnung wird auch durch die Linienvereinfachung, welche segmentweise vorgeht, nicht beeinflusst.

Matching Gebäude - Siedlungsflächen

Die fehlende Verknüpfung zwischen den ALK-Gebäuden sowie deren generalisierten Repräsentationen und den flächenhaften Objekten 'Siedlungsfläche' und 'Ortslage' des ATKIS-DLM sowie wiederum deren generalisierten Repräsentationen soll über eine einfache geometrische Zuordnung erfolgen. Unter der Voraussetzung gleicher Koordinatensysteme werden durch geometrische Funktionen die Gebäudegeometrien bestimmt, welche sich innerhalb einer Siedlungsfläche, respektive Ortslage befinden. Ergibt der Lagevergleich zwischen den Gebäude- und den Siedlungsgeometrien ein `within=true`, so kann eine Verknüpfung zwischen beiden Objekten in der Datenbank abgelegt werden. Die verbleibenden Gebäude werden auf ein `overlap=true` hin untersucht, was bedeutet, dass diese nur teilweise innerhalb einer zuzuordnenden Fläche liegen. Ragen Gebäude in mehr als eine Fläche (vgl. Abbildung 5.23), so werden diese derjenigen Fläche zugeordnet, welche die größte flächenmäßige Überlappung darstellt. Auf diese Weise können die Verknüpfungen zwischen den Gebäuden und den Siedlungsflächen beziehungsweise den Ortslagen abgeleitet werden. Dasselbe Verfahren wird angewendet, um die Siedlungsflächen mit der Ortslage zu verknüpfen.



Abb. 5.23: Zuordnung der Gebäude und Siedlungsflächen; einige Gebäude liegen in zwei Siedlungsflächen (siehe Markierung), die Zuordnung erfolgt über der Fläche mit der größten Überlappung

Matching: Gebäude - typifizierte Gebäude

Unter der Voraussetzung, dass beide zu verknüpfenden Datensätze in demselben Koordinatensystem vorliegen, ist davon auszugehen, dass Objekte, welche einen gewissen räumlichen Abstand nicht überschreiten und sich gleichzeitig kein alternativer Matching-Kandidat findet, dasselbe Realweltobjekt repräsentieren. In dem Fall der Typifizierung lässt sich argumentieren, dass ein bestimmtes Gebäude in dem Ausgangsdatsatz durch jene typifizierte Geometrie repräsentiert wird, welche dieser Ausgangsgeometrie am nächsten liegt (Abbildung 5.24). Das Ziel der Typifizierung liegt in der Reduktion der Daten bei gleichzeitigem Erhalt der Objektdichte und -verteilung. Daher kann jedes typifizierte Objekt als Stellvertreter seiner direkten Nachbarn interpretiert werden.

Somit soll zur Verknüpfung der typifizierten Gebäude mit den Ausgangsdaten jeder ursprünglichen Gebäudegeometrie der Kandidat zugeordnet werden, welcher dieser am nächsten liegt. Um die Suche nach den geeigneten Kandidaten zu beschleunigen, werden mit Hilfe eines Abstandsparameters d alle typifizierten Objekte gesucht, welche sich innerhalb dieses Umkreises um das Ausgangsobjekt befinden. Wird kein Kandidat gefunden, wird der Abstand schrittweise erhöht. Befinden sich mehrere Matchingkandidaten innerhalb des Umkreises, so wird aus dieser begrenzten Menge das nächstliegende Objekt gewählt und die Bezeichner beider Objekte werden als Paar gespeichert und am Ende des Matchingprozesses in die Datenbank geschrieben. Eine exemplarische Zuordnung ist in Abbildung 5.24 zu erkennen. Somit ergeben sich n:1 Relationen, welche in der entsprechen Datenbanktabelle abgelegt werden können.



Abb. 5.24: Ergebnis des Matchingverfahrens: Zuordnung der Ausgangsdaten (farbig) zu den typifizierten Objekten (schwarzer Rahmen)

Auf diese Weise steht an dieser Stelle eine integrierte multiskalige Datenbank zur Verfügung, welche fünf Auflösungsstufen der Objektbereiche 'Siedlung' und 'Verkehr' besitzt (siehe Tabelle 5.3 und Abbildung 5.25).

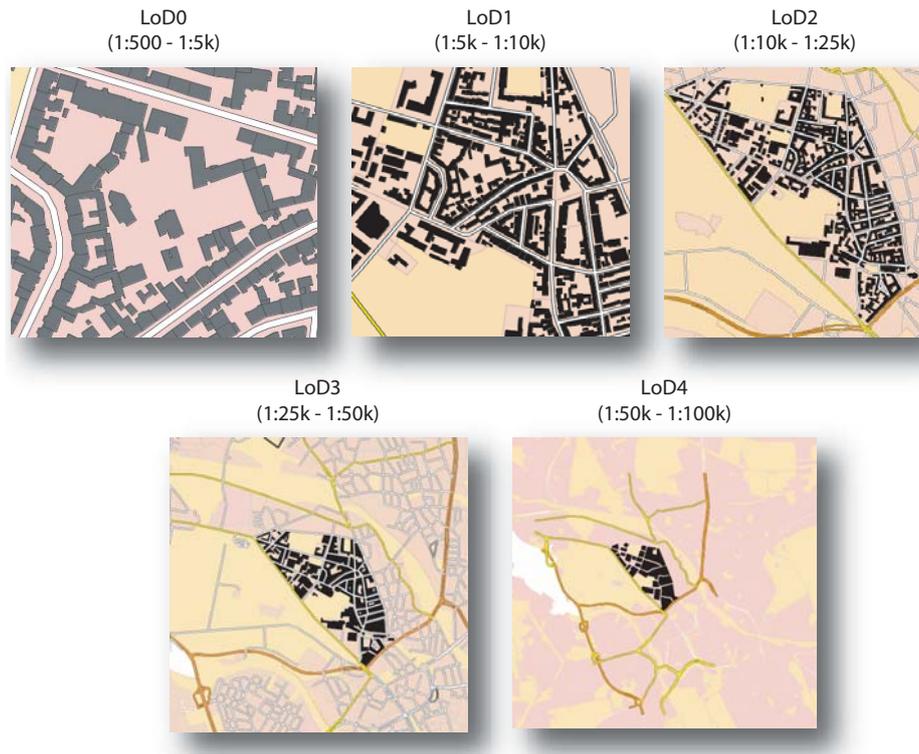


Abb. 5.25: Durch Generalisierung entstandene Auflösungsstufen der MRDB (LoD0 - LoD4); Gebäude- und Straßendaten sind nur für einen kleineren Kartenausschnitt verfügbar; Darstellung in etwa maßstäblich

Level	Maßstab	Objektbereich Siedlung	Objektbereich Verkehr
Level0	Daten höchster Auflösung	ALK-Gebäude, Gebäudepläne	MultiNet™ Straßendaten (Straßenkategorien 1, 2, 3, 4, 6), Verdrängung
Level1	bis 1:10.000	Aus Level0 abgeleitete Gebäude: Generalisierung durch Vereinfachung; Siedlungsfläche, Ortslage: Originaldaten aus BasisDLM	Generalisierte Straßendaten: Semantische Auswahl (Straßenkategorien 1, 2, 3, 4, 6)
Level2	1:25.000	Generalisierte Gebäude: Vereinfachung, Auswahl, Zusammenfassung; Siedlungsfläche, Ortslage: Originaldaten aus BasisDLM	Generalisierte Straßendaten: Semantische Auswahl (Straßenkategorien 1, 2, 3, 4), Verdrängung
Level3	1:50.000	Generalisierte Gebäude: Zusammenfassen, Vereinfachen, Lücken füllen (tlw. manuell) bei geschlossener Bebauung/Typifizierung bei lockerer Bebauung; Verdrängung; Siedlungsfläche (tlw. Zusammenfassen)	Generalisierte Straßendaten: Linienvereinfachung durch Douglas-Peucker, semantische Auswahl (Straßenkategorien 1, 2, 3), Verdrängung
Level4	1:100.000	Generalisierte Gebäude: Zusammenfassen, Vereinfachen, Lücken füllen (tlw. manuell) bei geschlossener Bebauung/Typifizierung bei lockerer Bebauung; Verdrängung; Siedlungsfläche (tlw. Zusammenfassen)	Generalisierte Straßendaten: Linienvereinfachung durch Douglas-Peucker und semantische Auswahl (Straßenkategorien 1, 2), Verdrängung

Tab. 5.3: Level der MRDB

6 MRDB-Webdienst: Funktionalitäten und Schnittstellendefinition

Die in dem Abschnitt 4.1 aufgestellten Anwendungsszenarien verdeutlichen, dass die Notwendigkeit besteht, Daten aus einer MRDB per Webservice abfragen zu können, damit der (mobile) Nutzer von den Inhalten und Vorzügen dieser profitieren kann. Ein *Service* beschreibt einen bestimmten Teil der Funktionalität, welcher durch eine Instanz über Schnittstellen angeboten wird (Lieberman 2003) (vgl. Abschnitt 2.3). Somit sollten gewisse Funktionalitäten über eine definierte Schnittstelle bereitgestellt werden, die es dem Anwender erlauben, auf die Inhalte der MRDB zuzugreifen. Im Folgenden werden daher zunächst sinnvolle und essenzielle Funktionalitäten eines derartigen Dienstes zusammengetragen, um diese anschließend zu implementieren und durch definierte Schnittstellen über das Internet zur Verfügung zu stellen.

Das OGC hält einen umfassenden Katalog bereit zur Vernetzung verschiedener Geodatenquellen (vgl. Unterabschnitt 2.3.2). Ziel ist es, einen interoperablen Rahmen zum Abruf, der Integration, Analyse, Nutzung und Visualisierung von Online-Geodatenquellen zu ermöglichen. Innerhalb dieses Netzwerkes, dem OWS, finden sich zahlreiche Spezifikationen für die unterschiedlichsten Bereiche eines Webdienstes. Unter anderem sind in diesem Zusammenhang auch Schnittstellen für den Bereich der Datenservices zum Zugriff auf Geodatenquellen, beispielsweise der WFS, beschrieben. Aufbauend auf diesem System ist es somit möglich, über eine WFS-Instanz auf die Daten einer MRDB zuzugreifen und diese entsprechend zu verarbeiten.

Der folgende Abschnitt fasst zunächst notwendige Funktionalitäten eines Webservices zum Abruf multiskaliger Daten zusammen. Anschließend sollen diese Funktionalitäten denen eines OGC-WFS gegenübergestellt werden, um herauszuarbeiten, ob eine Instanz des OGC-WFS diesen Anforderungen nachkommen kann, beziehungsweise, an welcher Stelle dieser auf Probleme und Grenzen stößt und somit Ergänzungsbedarf besteht. Der darauffolgende Abschnitt 6.3 gibt schon an dieser Stelle preis, dass die Möglichkeiten eines OGC-WFS hier nicht ausreichend sind. Der entsprechende Abschnitt führt eine Erweiterung des OGC-WFS ein, welche die Möglichkeiten einer MRDB voll unterstützten und an den Nutzer weitergeben soll.

6.1 Erforderliche Funktionalitäten eines MRDB-Services

Eine MRDB bietet ein Informationspotenzial, welches über das einer herkömmlichen Geo-Datenbank hinausgeht. Die Spezialität dieser liegt in der Speicherung mehrerer zu einem bestimmten Phänomen gehörigen Repräsentationen. In dem hier vorgestellten Prototypen werden geografische Objekte in unterschiedlicher geometrischer Auflösung vorgehalten. Neben den Objektinformationen besitzt die MRDB zusätzliche Angaben über die Beziehungen zwischen diesen alternativen Repräsentationen. Hierdurch entstehen ergänzende Anforderungen an einen Datenservice, den MRDB-Service, welche in diesem Abschnitt beschrieben werden und gleichzeitig neue Möglichkeiten für den Nutzer, der mit diesen Daten arbeitet (vgl. Abschnitt 3.3 und Kapitel 8). Die in Unterabschnitt 4.1.1 beschriebenen Szenarien helfen bei der Aufstellung der erforderlichen Funktionalitäten.

Die Anforderungen an einen Datenservice zur Abfrage monorepräsentativer Geodaten werden durch die Verwendung eines OGC-WFS abgedeckt. Der *Basic-WFS* umfasst hierzu folgende Operationen (OGC 2005c):

- *GetCapabilities*: Ein WFS ist in der Lage, seine Fähigkeiten zu beschreiben. Insbesondere zeigt er an, welche Objekttypen dieser bereitstellt und welche Operationen für jeden Objekttyp unterstützt werden
- *DescribeFeatureType*: Ein WFS ist in der Lage, auf Anfrage die Struktur jedes bereitgestellten Objekttypen zu beschreiben
- *GetFeature*: Ein WFS ist in der Lage, eine Anfrage nach Objektinstanzen zu bedienen. Zusätzlich hat der Client die Möglichkeit zur Auswahl relevanter Objektteile (Attribute) durch Verwendung entsprechender Filtermethoden

Ein *Transactional-WFS* verarbeitet außerdem Transaktionen zur Manipulation von Datenbankobjekten mit Hilfe von Operationen wie *create*, *insert* oder *update*. Zusätzliche Filterfunktionen werden durch die *Filter Encoding*

Specification (OGC 2005a) festgelegt. Diese beschreibt Operatoren zur Filterung räumlicher Daten, wie räumliche Operatoren, Vergleichsoperatoren, logische Operatoren, Objektidentifikatoren, arithmetische Operatoren etc.

Neben den grundlegenden Erfordernissen eines Datendienstes finden sich in den szenarienbasierten Untersuchungen (vgl. Abschnitt 4.1) zusätzliche spezifische Anforderungen, welche nur durch eine multiskalige Datenbasis erfüllt werden können. Daher werden von einem Datendienst, welcher als Schnittstelle zwischen dem Anwender und der MRDB fungiert und die Vorteile dieser an den Nutzer weitergeben möchte, zusätzliche Funktionalitäten erwartet. In den in Abschnitt 4.1 beschriebenen Szenarien finden sich folgende Anforderungen an einen Datenservice beziehungsweise den darauf aufbauenden Webservice (Kartendienst, Informationsservice o.ä.):

- Abfrage von Karten unterschiedlicher Maßstäbe
 - Kleiner Maßstab: Zur Fahrzeugnavigation, Übersicht über größere Gebiete und Strecken (Stadt oder mehrere Städte, Verkehrsnetz etc.)
 - Mittlerer Maßstab: Fußgängernavigation, Übersicht über mittelgroße Gebiete und kürzere Strecken (Stadtteile, Siedlungsstruktur etc.)
 - Großer Maßstab: Detailansicht zur Identifikation von Objekten (Gebäuden), möglichst viele Details der Objekte
- Abfrage aller zu einem Phänomen (z.B. Gebäude, Siedlungsfläche) verfügbaren Informationen
 - Grafische Informationen (alternative Repräsentationen, Gebäudepläne etc.)
 - Semantische Informationen (Gefahrgüter, Ansprechpartner, Einwohnerzahl, Mietspiegel, Bevölkerungsstruktur etc.)
- Informationsdarstellung für mobile Nutzer, angepasst an kleine Displays

Formal ergeben sich somit folgende, *erweiterte Anforderungen an einen MRDB-Datenservice*:

1. Kataloginformationen zur Beschreibung der Fähigkeiten und Inhalte des Dienstes
 - a) Reflexion der hierarchischen Datenstruktur: Jede Objektart beinhaltet mehrere Auflösungsstufen. Somit muss unterschieden werden zwischen der Objektart und einer Repräsentation einer Objektart
 - b) Beschreibung des Multiplizitätsparameters (hier: geometrische Auflösung, Maßstabsbereiche), durch den sich die Repräsentationen unterscheiden lassen
2. Abfrage von Objektinstanzen
 - a) Trennung zwischen Objektart und Repräsentationsstufe im Ergebnisdatensatz
 - b) Gezielte Abfrage spezifischer Objektarten und Repräsentationen unter Vorgabe von Auswahlkriterien (z.B. Objektart, Maßstab)
3. Abfrage der mit einem Objekt verknüpften Instanzen
 - a) Abfrage der vertikal verknüpften Objekte (Abfrage der mit einem Objekt verlinkten Objekte mit höherer/niedrigerer/bestimmter Auflösung)
 - b) Horizontal verknüpft (Abfrage der mit einem Objekt verlinkten Objekte gleicher Auflösung, z.B. Siedlungsfläche, welche mit dem Haus verknüpft ist)
 - c) Abfrage aller verknüpften Objekte (Abfrage aller alternativen Repräsentationen zu diesem Gebäudeobjekt, sowohl die unterschiedlichen Auflösungen als auch die alternativen Sichtweisen innerhalb einer Auflösungsstufe)
 - d) Abfrage der Verknüpfungsinformationen (Abfrage der IDs der mit einem Objekt verlinkten Objekte, damit z.B. eine Applikation die Zuordnungen intern speichern kann)
4. Abfrage der Attribute einer Instanz sowie der Attribute der hiermit verknüpften Instanzen
5. Filterung der Abfrageergebnisse
 - a) Ausweitung des Filters auf alternative/verlinkte Repräsentationen (z.B.: Filter sucht nach einem bestimmten Objekt im Ergebnisdatensatz (z.B. Gebäude), welches mit dem Objekt einer anderen Auflösungsstufe (z.B. einem generalisierten Gebäude) verlinkt ist)

6.2 Probleme bei Verwendung des OGC-WFS in einer multirepräsentativen Umgebung

Bei dem Versuch, die hier aufgeführten, desiderablen Funktionalitäten mit Hilfe eines OGC-WFS abzudecken, wird man auf Probleme stoßen (Hampe & Intas 2006). Derartige Anfragen lassen sich entweder nur umständlich durch

Hinzunahme eigener Funktionalitäten, wie die Kombination mehrfacher Abfragen zu einem Ergebnisdatensatz realisieren. Oder diese sind gar nicht möglich, weil bestimmte Informationen über einen OGC-WFS nicht abzufragen sind, wie beispielsweise die Verknüpfungen. Diese Probleme sollen durch die folgenden Ausführungen verdeutlicht werden.

Die erste oben aufgeführte Anforderung zur Abfrage der Kataloginformationen (1(a) und 1(b)) ermöglicht einer Applikation, welche Informationen über den Webservice bezieht, zuvor Informationen über die verfügbaren Daten einzuholen. Die Abfrage der Datenbankinhalte beruht auf diesen Informationen. In der Initialisierungsphase fragt beispielsweise die Applikation auf dem PDA von Christa (Szenario 2) Informationen über die verfügbaren Objektarten sowie die unterschiedlichen Auflösungsstufen ab. Somit kann je nach Situation die gewünschte Repräsentation aufgerufen werden. Vergrößert oder verkleinert der Nutzer den Kartenausschnitt, wird eine alternative Auflösungsstufe von dem Service angefordert. Der OGC-WFS bietet hier eine Anfrage `<GetCapabilities>`. Als Antwort wird ein XML-Dokument gesendet, welches unter anderem die durch den Service verfügbaren Objektklassen und deren Attribute darstellt (siehe Beispiel 6.2, S. 128). Es ist allerdings an keiner Stelle ein Hinweis zu finden, welche Objektklasse (GebäudeLoD0, GebäudeLoD1, GebäudeLoD2 etc.) zu welcher Objektart (Gebäude, Siedlungsfläche, Straße) und welcher Auflösungsstufe (LoD0, LoD1, LoD2 oder LoD3) gehört. Der OGC-WFS bietet hier also keine Möglichkeit, derartige Informationen abzufragen und die Bedingung 1(a) oder 1(b) zu erfüllen.

Eine Anfrage der Daten selbst erfolgt bei dem OGC-WFS über die Funktion `<GetFeature>`. Der Service liefert als Ergebnis die gewünschten Daten in Form eines GML-Dokumentes. Die Anforderungen 2 bis 4 können jedoch nicht mehr von dem OGC-WFS erfüllt werden. Da der OGC-WFS nicht zwischen den alternativen Repräsentationen unterscheiden, sondern lediglich unterschiedliche Objektklassen (`featureTypes`) spezifizieren kann, ist es nicht möglich, alle oder bestimmte Repräsentationen derselben Objektart abzufragen. Hier kann sich der Nutzer nur behelfen, indem er eine Tabelle anlegt, welche ihm anzeigt, über welchen `featureType` welche Objektart in welcher Auflösungsstufe aufzurufen ist. Soll ein Gebäude mit höchster Auflösung abgefragt werden, so wird zum Beispiel der `featureType=GebäudeLoD0` benötigt. Alternativ können diese Informationen der Metadatentabelle der MRDB entnommen werden. Dies widerspricht jedoch dem Prinzip der Datenkapselung. Der Nutzer sollte keinen direkten Zugriff auf die Datenbank erhalten, sondern lediglich einen gesteuerten Zugang über die Webservice-Schnittstelle.

Desweiteren ist keine Abfrage der verknüpften Objekte über den OGC-WFS möglich. Der Nutzer kann somit keine alternative Repräsentation zu einem bestimmten Objekt erhalten. Um dies zu ermöglichen, müssten die Verknüpfungsinformationen direkt an dem abgefragten Objekt als Attribut angehängt sein. Dies ist allerdings in dem Datenbankschema nicht vorgesehen (vgl. Unterabschnitt 5.2.3).

Um die Abfrageergebnisse vor der Ausgabe zu filtern bietet der OGC-WFS die in einer speziellen Filter-Spezifikation definierten, weitreichenden Filtermethoden an. So ist es beispielsweise möglich, nur Objekte eines bestimmten Gebietes (Bounding-Box) in das Ergebnis zu übernehmen oder nur Objekte mit bestimmten Attributwerten zu filtern. Der OGC-WFS bietet jedoch nur die Möglichkeit, das aktuelle Anfrageergebnis zu filtern. Alternative Repräsentationen werden hier nicht berücksichtigt. Benötigt der Anwender eine bestimmte Repräsentation für ein bestimmtes Objekt (z.B. aus dem LoD0), dessen ID nur in einer anderen Auflösungsstufe (z.B. LoD1) bekannt ist, so lässt sich dieses Objekt auf diese Weise nicht filtern.

6.3 Erweiterung des OGC-WFS zur Abfrage multirepräsentativer Daten

Die Untersuchungen des vorangegangenen Abschnittes zeigen, dass es einer Erweiterung der Funktionalitäten des OGC-WFS bedarf, wenn dieser die in den Anwendungsszenarien ermittelten Anforderungen an einen MRDB-Service erfüllen soll. Dabei ist der OGC-WFS nicht komplett zu ersetzen, sondern es wird versucht, diesen lediglich durch ein zusätzliches Modul um bestimmte MRDB-spezifische Funktionen zu erweitern. Hier zeigt sich der in Abschnitt 2.1 beschriebene Vorteil einer Webservice-Architektur. Einzelne Servicekomponenten, wie der hier beschriebene MRDB-Service, können auf bestehende Dienste, wie den OGC-WFS, über definierte Schnittstellen zugreifen und für ihre Zwecke nutzen.

6.3.1 Architektur des MRDB-Service

Die grundlegende Architektur des OGC-WFS lässt sich in Abbildung 6.1 ablesen. Neben dem Client ist hier besonders der für den Nutzer unsichtbare Datenspeicher hervorzuheben. Alle Informationen erhält der

Anwender über den WFS, er hat keinen direkten Zugriff auf den Datenspeicher, was durchaus sinnvoll ist, da der Serviceanbieter das kostbare Datengut nicht ungeschützt der Nutzerschicht preisgeben möchte. Weiterhin ist somit trotz unterschiedlicher Datenbankschemata der Zugriff auf die Daten über eine einheitliche Schnittstelle sichergestellt. Änderungen des Schemas haben keine Auswirkungen auf die Kommunikation mit dem Client. Die Datenbank sollte aus diesem Grund losgelöst von der Nutzerschicht sein. Lediglich dem Service ist ein Zugriff auf die Datenbank erlaubt. Die Erweiterung des OGC-WFS, im Folgenden als MR-WFS¹ bezeichnet, ist in der

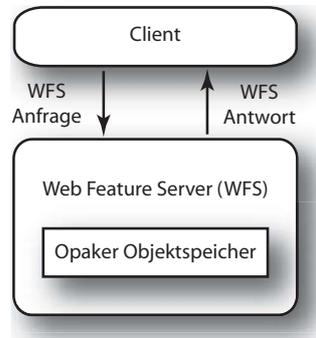


Abb. 6.1: OGC-WFS Architektur (aus OGC 2005c)

Abbildung 6.2 abgebildet. Auch hier soll das Prinzip der Kapselung nicht durchbrochen werden. Der Anwender kommuniziert über das HTTP in Form einer POST-Anfrage² ausschließlich mit dem MR-WFS. Dieser greift wiederum auf die MRDB zurück, um Meta- und Verknüpfungsinformationen zu erhalten. Die Datenabfrage selbst erfolgt über eine OGC-WFS-Implementierung, dem *Geoserver*³, der konform zu der OGC-Spezifikationen WFS 1.0 (OGC 2005c) entwickelt wurde. Im einfachsten Fall können die Anfragen ohne Änderung direkt vom MR-WFS an den *Geoserver* weitergereicht werden. Wird dagegen eine der oben genannten Funktionalitäten benötigt, so sammelt der MR-WFS die entsprechenden Daten auf den im Folgenden beschriebenen Wegen, bereitet diese auf und sendet das Ergebnis in Form eines GML-Datensatzes an den Client.

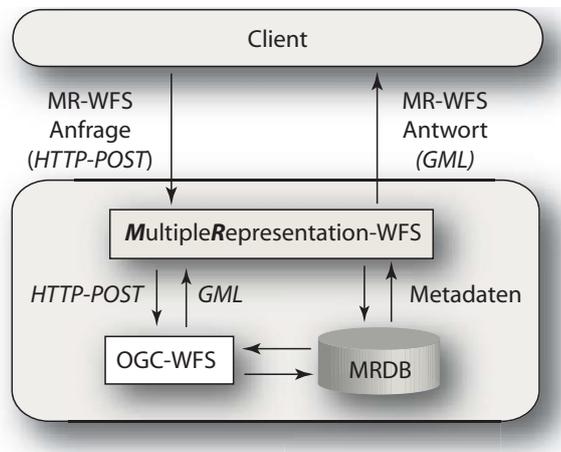


Abb. 6.2: Architektur des erweiterten WFS für Datenquellen mit multiplen Repräsentationen

6.3.2 Funktionen und Implementierung eines MRDB-Service

Die Logik- oder Applikationsebene, in die auch der MR-WFS einzuordnen ist, wird, wie in Abschnitt 2.2 detailliert nachzulesen ist, über Web- und Applikationsserver realisiert. Der Webserver, in dem hier beschriebenen Fall der *Apache Tomcat*⁴, nimmt die Anfrage des Clients entgegen, leitet diese an den Applikationsserver weiter und

¹Multiple Representation Web Feature Service

²Kommuniziert der Client mit dem Server über die POST-Methode, so wird die Anfrage nicht über die URI (wie bei GET), sondern im *Body* der Anfrage übermittelt.

³<http://docs.codehaus.org/display/GEOS/Home>

⁴<http://tomcat.apache.org/>

sendet den Ergebnisdatensatz an den Client zurück. Der Applikationsserver, hier die *Java EE 5 SDK*⁵, ermöglicht die Entwicklung entsprechender Funktionalitäten. In diesem Fall werden die Operationen des MR-WFS also in Form von *Java-Servlets* realisiert.

Die Anfragen werden zunächst im XML-Format über HTTP-POST an ein sogenanntes *Dispatcher-Servlet* gesendet, welches die Anfrage analysiert und an die jeweils verantwortlichen Servlets verteilt. Um die herausgearbeiteten Erfordernisse zu realisieren, wurden die im Folgenden vorgestellten Operationen implementiert. Die jeweiligen Anfragen an den MR-WFS sowie resultierende Ausgaben werden durch einige Beispiele dargestellt und den jeweiligen Entsprechungen des OGC-WFS gegenübergestellt. Die Unterschiede zwischen beiden Schnittstellen sind in den Beispielen durch Variation der Schriftstärke markiert.

GetCapabilities

Die Operation `<GetCapabilities>` soll die oben aufgestellte Bedingung 1 erfüllen. Diese beschreibt, genau wie im OGC-WFS, die Eigenschaften und Fähigkeiten des Dienstes. Der MR-WFS soll diese Funktion des OGC-WFS nicht vollständig ersetzen. Vielmehr besteht das erklärte Ziel darin, MRDB-spezifische Informationen zu ergänzen. Daher wird die Anfrage an die Implementierung des OGC-WFS weitergeleitet. Gleichzeitig werden ergänzende Informationen aus der Metadatentabelle der MRDB entnommen und auf die unten beschriebene Weise in das Ergebnis des OGC-WFS integriert.

Um die hierarchische Struktur der MRDB zum Ausdruck zu bringen, wird der Teil des Ergebnisdatensatzes, welcher die verfügbaren Objekttypen beschreibt (`<FeatureTypeList>`), durch eine zusätzliche Hierarchiestufe (`<Representation>`) ergänzt, welche die jeweiligen Repräsentationen einer Objektart darstellt. Eine Karten-Applikation benötigt eine Liste der verfügbaren Objektarten, um dem Nutzer eine entsprechende Auswahlliste der verfügbaren Kartenebenen anbieten zu können. Gleichzeitig benötigt diese Applikation jedoch in dem Fall einer multiskaligen Datenbasis eine Liste der verfügbaren Auflösungsstufen jeder Objektart und ihrer Eigenschaften. So bietet die hier beschriebene MRDB beispielsweise die Objektarten 'Gebäude', 'Siedlungsfläche', 'Ortslage' und 'Straßen'. Die Objektart 'Gebäude' besitzt die Repräsentationen 'GebäudeLoD0', 'GebäudeLoD1', 'GebäudeLoD2', 'GebäudeLoD3' und 'GebäudeLoD4'. 'GebäudeLoD0' besitzt dabei eine Auflösung, welche für den Maßstabsbereich 1:1k bis 1:5k empfohlen wird. Genau diese Informationen benötigt die Applikation, um eine Karte eines bestimmten Maßstabes generieren zu können.

Beispiel 6.2 zeigt das Resultat einer `<GetCapabilities>` Anfrage, wie es der OGC-WFS liefert. Problematisch ist das Element `<FeatureTypeList>`, welches eine Liste aller durch den Service verfügbaren Objektarten anzeigt. Es fehlt hier eine hierarchische Ebene, welche die verschiedenen Repräsentationen derselben Objektart zusammenfasst. Daher wird die Ausgabe durch den MR-WFS in der Weise modifiziert, als dass nun das Element `<FeatureType>` zusätzlich ein weiteres Unterelement `<Representation>` enthält, welches die verfügbaren Repräsentationen in jeder Objektart auflistet. In diesem Fall wird der `<FeatureType>` zu einem abstrakten Objekt, die in der Datenbank gespeicherten Instanzen finden sich in dem Element `<Representation>`.

Das Beispiel 6.3 zeigt einen Ausschnitt der Antwort des MR-WFS auf eine Anfrage `<GetCapabilities>`. Die Informationen, welche Repräsentation welcher übergeordneten Objektart zuzuordnen ist, entnimmt der Service der MRDB-Metadatentabelle. Zusätzlich sind Merkmale zur Unterscheidung der verschiedenen Repräsentationen ('ScaleMin', 'ScaleMax') in dem resultierenden Dokument zu finden.

Der MR-WFS stellt zunächst eine Anfrage `<GetCapabilities>` an den OGC-WFS. Anschließend selektiert der MR-WFS aus der Metadatentabelle die Spalten 'Repräsentation', 'Objektart', 'ScaleMin' und 'ScaleMax' und ordnet die Elemente der `<FeatureTypeList>`, beruhend auf diesen Informationen, neu an, indem als `<Name>` nun der Name der Objektart eingesetzt wird und ein neues Unterelement `<Representation>` gebildet wird. Alle übrigen Elemente werden aus dem resultierenden Dokument der OGC-WFS Anfrage übernommen.

Die Anwenderapplikation, welche eine `<GetCapabilities>` Anfrage an den MR-WFS gestellt hat, kann die entsprechenden Informationen über die verfügbaren Objektarten und Auflösungsstufen dem resultierenden Dokument entnehmen, indem nach den Schlüsselwörtern `<Name>`, `<FeatureType>` und `<Representation>` gesucht wird. Die Beispiele 6.1, 6.2 und 6.3 stellen eine Anfrage `<GetCapabilities>` und die jeweiligen Antworten eines OGC-WFS sowie des MR-WFS dar.

⁵<http://java.sun.com/javaee>

Bsp. 6.1 Anfrage *GetCapabilities*

```
<getCapabilities></getCapabilities>
```

Bsp. 6.2 Antwort des OGC-WFS auf *GetCapabilities*

```
<WFS_Capabilities version="1.0.0" ...>
  <Service>...</Service>
  <Capability>...</Capability>
  <FeatureTypeList>
    <Operations>...</Operations>
    <FeatureType>
      <Name>GebäudeLoD0</Name>
      ...
    </FeatureType>
    <FeatureType>
      <Name>GebäudeLoD1</Name>
      ...
    </FeatureType>
    <FeatureType>
      <Name>GebäudeLoD2</Name>
      ...
    </FeatureType>
    ...
    <FeatureType>
      <Name>StrassenLoD0</Name>
      ...
    </FeatureType>
    ...
  </FeatureTypeList>
  ...
</WFS_Capabilities>
```

Bsp. 6.3 Antwort des MR-WFS auf *GetCapabilities*

```
<MR-WFS_Capabilities version="1.0.0" ...>
  <Service>...</Service>
  <Capability>...</Capability>
  <FeatureTypeList>
    <FeatureType>
      <Name>Gebäude</Name>
      <Representation>
        <ReprName>GebäudeLoD0</ReprName>
        <Scalemin>1000</Scalemin>
        <Scalemax>10000</Scalemax>
        ...
      </Representation>
      <Representation>
        <ReprName>GebäudeLoD1</ReprName>
        <Scalemin>10001</Scalemin>
        <Scalemax>25000</Scalemax>
        ...
      </Representation>
    </FeatureType>
    <FeatureType>Strassen</FeatureType>
    <Representation>
      ...
    </Representation>
    ...
  </FeatureTypeList>
  <ogc:Filter_Capabilities>...</ogc:Filter_Capabilities>
</MR-WFS_Capabilities>
```

GetAlternativeFeature

Die Operation `<GetAlternativeFeature>` ist ausschließlich über den MR-WFS verfügbar und soll den oben genannten Anforderungen 2 bis 4 gerecht werden. Die Intention dieser Operation liegt in der gezielten Abfrage einer Repräsentation über ein bestimmtes Merkmal. In dem hier beschriebenen Fall enthält die MRDB unterschiedliche Auflösungsstufen der einzelnen Objektarten, um unterschiedliche Maßstäbe bedienen zu können. Daher wird der empfohlene Maßstabsbereich als Parameter eingeführt, um spezifische Repräsentationen für den aktuellen Zweck abfragen zu können. Die Operation `<GetAlternativeFeature>` entspricht dem OGC-WFS Pendant `<GetFeature>`, ergänzt die Abfrage allerdings um den Parameter `'scale'`.

Die Abfrage `<GetFeature>` des OGC-WFS erlaubt es, bestimmte Objektklassen, welche in monoskaligen Datenbanken jeweils eine Objektart repräsentieren, mit Hilfe des Parameters `'typename'` gezielt aus der Datenbank

abzufragen. In dem Fall einer multirepräsentativen Datenbank dagegen gilt es nicht nur, die verschiedenen Objektarten, sondern zusätzlich deren unterschiedlichen Repräsentationen abfragen zu können. Jede Repräsentation hat bestimmte Eigenschaften, wodurch diese sich von den übrigen Repräsentationen unterscheidet. In diesem Fall liegen hier Daten unterschiedlicher Auflösung vor, welche sich zur Darstellung in verschiedenen Maßstäben eignen. Somit wird an dieser Stelle der zusätzliche Parameter 'scale' eingeführt, durch welchen die gewünschte Repräsentation bestimmt wird.

Von dem MR-WFS wird im einfachsten Fall lediglich eine Übersetzung der Anfrage vorgenommen. Mit den Parametern 'scale=9000' und 'typename=Gebäude' sucht der MR-WFS in der Metadaten-tabelle der MRDB den Namen der Klasse der entsprechende Repräsentationen (hier 'GebäudeLoD1'). Daraufhin wird eine entsprechende Anfrage <GetFeature> an den OGC-WFS mit dem Parameter 'typename=GebäudeLoD1' gestellt (siehe Abbildung 6.3).

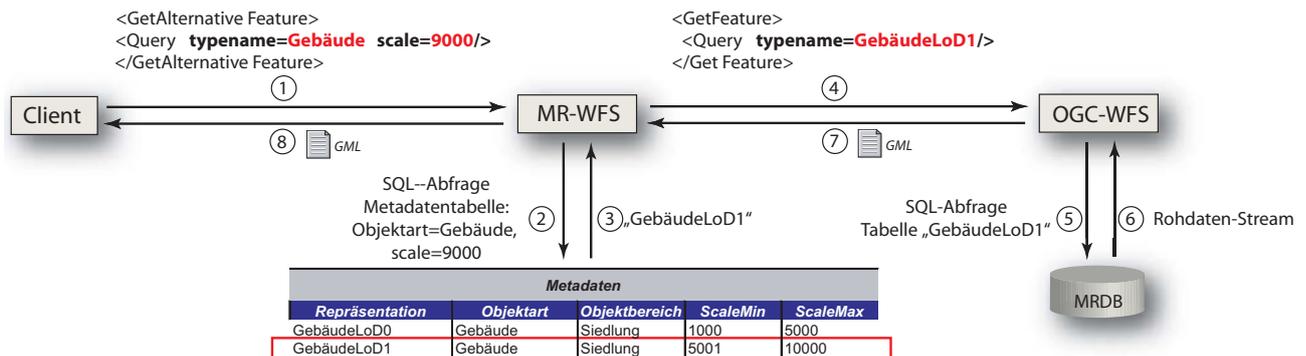


Abb. 6.3: Ablauf bei einer einfachen Anfrage <GetAlternativeFeature> an den MR-WFS

Eine derartige Übersetzung kann von dem Anwender auch selbst vorgenommen werden, da die hierzu notwendigen Informationen aus dem Anfrageergebnis <GetCapabilities> entnommen werden können. Somit wird an dieser Stelle dem Anwender lediglich eine derartige Übersetzung abgenommen, wodurch die Anfrage der gewünschten Daten für diesen wesentlich einfacher wird. Gleichzeitig kann der Anwender sicher sein, dass seine Anfrage korrekt beantwortet wird, auch wenn die Auflösungsstufen in der Datenbank in der Zwischenzeit überarbeitet wurden. So ist es durchaus denkbar, dass nach einer Aktualisierung der Datenbank die Objektklasse 'GebäudeLoD1' nicht mehr dem Maßstab 1:9k entspricht.

Bsp. 6.4 Spezifische Anfrage GetAlternativeFeature des MR-WFS

```
<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="9000"></Query>
</GetAlternativeFeature>
```

Das Beispiel 6.4 zeigt eine entsprechend formulierte Anfrage an den MR-WFS. Diese Anfrage liefert alle Daten der Objektart 'Gebäude', welche zur Darstellung in dem Maßstab 1:9k gekennzeichnet sind. Sind diese Daten nicht verfügbar, so resultiert hieraus eine entsprechende Fehlermeldung. Beispiel 6.5 zeigt das resultierende Ergebnis ausschnittsweise. Außerdem ist die Ausgabe in der Art modifiziert, als dass auch hier, ähnlich der Anfrage <GetCapabilities>, eine zusätzliche Hierarchiestufe eingeführt wird, welche die Objektart (z.B. MRDB:Gebäude) beschreibt, zu der das jeweilige Objekt gehört. Dies ist notwendig, wenn in einer Anfrage mehr als eine Objektart abgefragt wird (siehe Beispiel 6.6).

Darüber hinaus ist es möglich, innerhalb einer <GetAlternativeFeature> Operation mehr als eine Objektart, aber auch mehr als eine Repräsentation, für eventuelle Vergleichsanalysen, abzufragen (Beispiel 6.6). Das Ergebnis dieser Anfrage stellt sich ähnlich dem Beispiel 6.5 dar, wobei in dem resultierenden GML-Dokument dann zusätzlich Gebäude für den Maßstab 1:50k und Straßenobjekte (<MRDB:Straßen>) für den Maßstab 1:9k zu finden sind.

Somit wird hier die 3. Anforderung erfüllt, welche die Möglichkeit zur Abfrage der zu einer bestimmten Objektart und Repräsentation gehörigen Instanz unter Vorgabe spezifischer Auswahlkriterien fordert. Auf diese Weise kann eine Applikation nun über <GetAlternativeFeature> eine dem Maßstab angepasste Visualisierung generieren. Möchte Christa (Szenario 2) sich die Lage der Wohnungen in einer Übersichtskarte anzeigen lassen, um beurteilen zu können, wie zentral diese positioniert sind, so berechnet die Applikation den notwendigen Maßstab, um den gesamten Stadtteil auf dem PDA darstellen zu können. Anschließend wird eine Anfrage an den Service gestellt

Bsp. 6.5 Antwort des MR-WFS auf GetAlternativeFeature

```

...
<MRDB:Gebäude>
<Scalemin>5001</Scalemin>
<Scalemax>10000</Scalemax>
  <gml:featureMember>
    <MRDB:GebäudeLoD1 fid="GebäudeLoD1.1">
      <MRDB:the_geom>...</MRDB:the_geom>
      <MRDB:HSNR>5</MRDB:HSNR>
      <MRDB:STR>Hauptstraße</MRDB:STR>
    </MRDB:GebäudeLoD0>
  </gml:featureMember>
</MRDB:Gebäude>
<MRDB:Gebäude>
<Scalemin>5001</Scalemin>
<Scalemax>10000</Scalemax>
  <gml:featureMember>
    <MRDB:GebäudeLoD1 fid="GebäudeLoD1.2">
      <MRDB:the_geom>...</MRDB:the_geom>
      <MRDB:HSNR>7</MRDB:HSNR>
      <MRDB:STR>Hauptstraße</MRDB:STR>
    </MRDB:GebäudeLoD0>
  </gml:featureMember>
</MRDB:Gebäude>
<MRDB:Gebäude>
<Scalemin>5001</Scalemin>
<Scalemax>10000</Scalemax>
  <gml:featureMember>
    <MRDB:GebäudeLoD1 fid="GebäudeLoD1.3">
      <MRDB:the_geom>...</MRDB:the_geom>
      <MRDB:HSNR>9</MRDB:HSNR>
      <MRDB:STR>Hauptstraße</MRDB:STR>
    </MRDB:GebäudeLoD0>
  </gml:featureMember>
</MRDB:Gebäude>
...

```

Bsp. 6.6 GetAlternativeFeature des MR-WFS mit mehreren Anfragen

```

<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="9000"></Query>
  <Query typeName="Gebäude" scale="50000"></Query>
  <Query typeName="Straße" scale="9000"></Query>
</GetAlternativeFeature>

```

<GetAlternativeFeature> mit den Parametern, typename='Gebäude' und scale='48000', um automatisch eine adäquate Darstellung anbieten zu können.

Besitzt eine MRDB verschiedene Repräsentationen, welche sich nicht durch den zu verwendenden Maßstabsbereich, sondern durch andere Merkmale (z.B. Erfassungszeitraum) unterscheiden lassen, so ist das System entsprechend anzupassen. Die zur Unterscheidung herangezogenen Merkmale werden in jedem Fall in der Metadatentabelle zu finden sein.

Filterfunktionen

Das Ergebnis der Anfrage eines OGC-WFS kann mit Hilfe von Filterfunktionen entsprechend der *Filter Encoding Implementation Specification* (OGC 2005a) reduziert werden. Der MR-WFS ermöglicht die Einbindung aller Filtermethoden, welche von der OGC-WFS Implementierung unterstützt werden, in die <GetAlternativeFeature> Umgebung. In dem Beispiel 6.7 werden Gebäude für den Maßstab 1:10k angefragt, welche die gid='10' oder '20' tragen.

Bsp. 6.7 GetAlternativeFeature Operation inklusive Filterfunktion des MR-WFS

```

<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="10000">
    <Filter>
      <Or>
        <FeatureId gid="10" />
        <FeatureId gid="20" />
      </Or>
    </Filter>
  </Query>
</GetAlternativeFeature>

```

Motiviert durch die Forderung, eine bestimmte Gruppe von Objekten oder einzelne Objekte durch alternative Repräsentationen zu ersetzen, muss es möglich sein, nicht nur eine bestimmte Auflösungsstufe wählen zu können, wie es `<GetAlternativeFeature>` ermöglicht. Vielmehr sollte eine Funktion bereitstehen, um nur bestimmte Objekte einer bestimmten Repräsentation abfragen zu können. Der Feuerwehrmann in dem Szenario 1 möchte eine möglichst detaillierte Darstellung eines bestimmten Gebäudes für den Maßstab 1:1k erhalten. Er klickt hierzu auf das generalisierte Gebäude 'GebäudeLoD2.100'⁶. Die Applikation stellt eine Anfrage `<GetAlternativeFeature>` mit den Parametern 'scale=1000' und 'typename=Gebäude' an den MR-WFS. Dieser sucht alle Gebäudeobjekte der Objektklasse 'GebäudeLoD0' aus der Datenbank.

Der verfügbare Filter kann an dieser Stelle jedoch nur die Daten des angefragten Datensatzes untersuchen. Er kann auf diese Weise nicht angeben, dass hier das Objekt gesucht ist, welches mit dem Objekt 'GebäudeLoD2.100' verlinkt ist, da kein Attribut Informationen hierüber liefert. Die Applikation hat desweiteren auch keinen Zugriff auf die entsprechenden Linktabellen. Daher ist es Aufgabe der Zugriffsschnittstelle, also des MR-WFS, eine Filterung verlinkter Objekte zu ermöglichen. Es muss an dieser Stelle also ein Filterelement eingeführt werden, welches es dem Nutzer erlaubt, aus dem Ergebnisdatsatz diejenigen Objekte herauszufiltern, welche mit einem bestimmten Objekt verlinkt sind. Daher wird an dieser Stelle das Filterelement `<ObjectRepresentation>` eingeführt, welches ein zu filterndes Objekt einer beliebigen Auflösungsstufe spezifiziert. Beispiel 6.8 soll diese Funktion verdeutlichen. Als Ergebnis dieser Anfrage werden diejenigen Objekte ausgegeben, welche für den

Bsp. 6.8 Filterfunktion des MR-WFS mit dem Element `<ObjectRepresentation>` zur Nutzung alternativer Repräsentationen innerhalb des Filters

```
<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="1000">
    <Filter>
      <Or>
        <ObjectRepresentation>gebäudeLoD2.100</ObjectRepresentation>
      </Or>
    </Filter>
  </Query>
</GetAlternativeFeature>
```

Maßstab 1:1k klassifiziert wurden und mit dem Objekt 'GebäudeLoD2.100' verknüpft sind. Somit kann dem Feuerwehrmann durch eine derartige Abfrage eine hochauflösende Repräsentation des Gebäudes dargestellt werden.

Der MR-WFS greift an dieser Stelle zunächst auf die Metadatentabelle der MRDB zurück, um herauszufinden, in welcher Tabelle die Verknüpfungen zu der Repräsentation 'GebäudeLoD2' zu finden sind. In dem Fall sind diese in der Tabelle 'GebäudeLinksV' gespeichert. Anschließend bestimmt der Service anhand der Metadaten, dass die Gebäude-Repräsentationen für den Maßstab 1:1k durch 'GebäudeLoD0' dargestellt werden. In der Verknüpfungstabelle wird nun das Objekt 100 in der Spalte 'GebäudeLoD2' gesucht und in der entsprechenden Zeile der Wert der Spalte 'GebäudeLoD0' ausgelesen. Da das Objekt 100 mit mehreren Objekten verknüpft ist, taucht es in verschiedenen Zeilen auf. Die entsprechenden Einträge der Spalte 'GebäudeLoD0' lauten beispielsweise '1,2,3,4,5'. Mit diesen Werten kann nun eine Anfrage an den OGC-WFS gesendet werden, welcher den 'typename=GebäudeLoD0' anfragt und aus dem Ergebnis die Objekte mit 'gid=1,2,3,4,5' filtert. Diese Geometrieobjekte können anschließend an den PDA des Feuerwehrmannes gesendet werden.

Oftmals ist es gewünscht, nicht nur bestimmte Objekte, sondern einen definierten räumlichen Bereich aus dem Datensatz herauszufiltern. Hierzu stellt der OGC-WFS das Filterobjekt Bounding-Box (BBOX) zur Verfügung. Zwei Punkte definieren hierbei ein Rechteck, welches den Filterbereich aufspannt. In den beschriebenen Szenarien findet dieser Filter beispielsweise Anwendung, wenn Christa eine detailliertere Darstellung eines Kreuzungsbereiches erhalten möchte. So sollen an Stelle der Siedlungsflächen einzelne Gebäudegrundrisse in einem Rechteck von 500 m x 500 m dargestellt werden. Es ist problemlos möglich, über `<GetAlternativeFeature>` inklusive eines Bounding-Box-Filters hochauflösende Gebäude, zum Beispiel für den Maßstab 1:5k, abzufragen.

Der Filter arbeitet derart, dass alle Objekte, welche ganz oder teilweise innerhalb des Rechteckes liegen, in das Ergebnis übernommen werden. Die Applikation auf dem PDA verwendet dasselbe Rechteck, um die Repräsentationen zu bestimmen, welche durch die höher aufgelösten Objekte ersetzt werden sollen, und somit gelöscht werden können. Abbildung 6.4 verdeutlicht die Problematik dieser Situation. Die Lücke, die durch die gelöschten Siedlungsflächen entsteht, ist größer als die resultierenden Gebäude abdecken. Es ist an dieser Stelle notwendig, dem Service mitzuteilen, dass die Bounding-Box zunächst auf die gering aufgelösten Objekte

⁶Objekte in einem GML/SVG-Datensatz werden bezeichnet mit dem Namen der Objektklasse und der eigentlichen ID, getrennt durch einen Punkt

Bsp. 6.9 Ursprüngliches Filterelement BBOX, Ergebnis (Abbildung 6.4, Mitte)

```

<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="5000">
    <Filter>
      <BBOX>
        ...
      </BBOX>
    </Filter>
  </Query>
</GetAlternativeFeature>

```

anzuwenden ist, um anschließend die hochauflösenden Objekte filtern zu können, welche mit diesen verknüpft sind. Der Standardfilter liefert nur das in der Abbildung in der Mitte dargestellte Ergebnis, da nur die hochauflösenden Gebäude innerhalb des Rechteckes herausgefiltert werden. Aus diesem Grund muss der Parameter 'scale' auch innerhalb der Filterumgebung 'BBOX' verfügbar sein (Beispiel 6.10). Hierdurch wird es dem Anwender ermöglicht, anzugeben, auf Grundlage welcher Repräsentation die Filterung vorgenommen wird. Die in der Abbildung 6.4 unten dargestellte Lösung lässt sich nur durch den genannten Parameter erzeugen.



Abb. 6.4: Teile der Karte werden durch höher aufgelösende Daten ersetzt. Oben: Bounding-Box und betroffene Objekte geringer Auflösung. Mitte: Bounding-Box, angewendet auf Objekte hoher Auflösung, der betroffene Bereich ist kleiner, eine Kombination der Daten lässt Lücken entstehen. Unten: BBox-Filter wird auf Daten geringer Auflösung bezogen und durch Abfrage der verlinkten Objekte auf die hohe Auflösung übertragen

Das Beispiel 6.10 zeigt eine Abfrage, welche das in der Abbildung unten gezeigte Ergebnis liefert. Es wird ein Gebäudedatensatz für den Maßstab 1:9k benötigt, der BBOX-Filter soll sich aber auf die Daten der Repräsentation 1:100k beziehen.

Bsp. 6.10 Filterelement BBOX des MR-WFS mit zusätzlichem Parameter scale, um den Filter auf eine alternative Repräsentation zu lenken (Abbildung 6.4 oben)

```

<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="9000">
    <Filter>
      <Or>
        <BBOX scale='100000'>
          ...
        </BBOX>
      </Or>
    </Filter>
  </Query>
</GetAlternativeFeature>

```

Der Service nutzt hierzu die Verknüpfungen zwischen den einzelnen Repräsentationen. Es werden zunächst die Objekte für den Maßstab 1:100k über den OGC-WFS abgefragt, wobei gleichzeitig die Bounding-Box als Filter

eingesetzt wird. Die resultierenden Daten werden auf ihre ID hin untersucht. Anschließend wird eine Anfrage `<GetAlternativeFeature>` formuliert, wobei nun das Filterelement `<ObjectRepresentation>` genutzt wird. Das System fragt die Gebäude (`'featuretype=Gebäude'`) für den Maßstab 1:9k (`'scale=9000'`), wobei ein Filter (`<ObjectRepresentation>`) eingefügt wird, der die IDs der vorher bestimmten Objekte geringer Auflösung enthält. Auf diese Weise werden die hiermit verknüpften Objekte hoher Auflösung abgefragt und an den Nutzer geschickt. Es resultiert hieraus das in der Abbildung unten dargestellte Ergebnis. Wird der Parameter innerhalb der `<BBOX>` Umgebung nicht verwendet, so bezieht sich der Filter auf die Objekte der angefragten Auflösung (hier 1:9k).

Abfrage der Verknüpfungen

Schließlich ist es für bestimmte Anwendungen interessant, nicht die verknüpften Objekte selbst abzufragen, sondern lediglich das Wissen über die Verknüpfungen. Hat die Applikation des Feuerwehrmannes zum Beispiel mehrere hochauflösende Gebäudedarstellungen über `<GetAlternativeFeature>` abgefragt, welche nun in dem Speicher des Gerätes liegen, so muss diese Anfrage nicht erneut gestellt werden, wenn der Anwender ein weiteres Mal auf das generalisierte Gebäude klickt. Wurden zu den hochauflösenden Gebäuden auch die IDs der verknüpften Objekte festgehalten, so kann diese Information nun geprüft und die entsprechende Geometrie aus dem Speicher des Gerätes aufgerufen werden. Für diesen Zweck wird das Element `<Links>` eingeführt (Beispiel 6.11), welches in die Anfrage `<GetAlternativeFeature>` integriert werden kann.

Dabei sind hier zwei Verknüpfungsrichtungen zu unterscheiden. Der vertikale Link verknüpft Objekte unterschiedlicher Auflösungsstufen, geeignet zur Darstellung in verschiedenen Maßstäben. Die Abfrage alternativer Repräsentationen wurde an diesem Parameter festgemacht. Die horizontale Verknüpfung setzt unterschiedliche Objektarten in Beziehung, welche teilweise denselben Teil der Realität repräsentieren, so zum Beispiel die Gebäude und die Siedlungsflächen. Auf dieser Ebene ist der Austausch von Attributdaten interessant, da unterschiedliche Objektarten auch unterschiedliche Attribute besitzen, welche jedoch auch für die jeweils andere Objektart von Interesse sind. So möchte Christa zum Beispiel die Einwohnerzahl des Stadtteiles erfahren, in dem ein bestimmtes Gebäude steht. Somit werden hier zwei Parameter eingeführt. `<Scale>` gibt Informationen über verknüpfte Objekte in dem genannten Maßstab und `<Objektart>` gibt Informationen zu den verknüpften Objekten der jeweiligen Objektart. In dem Beispiel 6.11 ruft die Applikation eine Gebäuderepräsentation für den Maßstab 1:8k

Bsp. 6.11 Element `<Links>` des MR-WFS zur Abfrage der IDs verknüpfter Objekte

```
<GetAlternativeFeature>
  <Query typeName="Gebäude" scale="8000">
    <Links>
      <Scale>15000</Scale>
      <Scale>35000</Scale>
      <Objektart>Siedlungsfläche</Objektart>
    </Links>
    <Filter>
      <Or>
        <ObjectName>GebäudeLoD0.13</ObjectName>
      </Or>
    </Filter>
  </Query>
</GetAlternativeFeature>
```

auf. Dabei wird diejenige Repräsentation herausgefiltert, welche mit dem Objekt `'GebäudeLoD0.13'` verknüpft ist. Es wird eine entsprechende Gebäudegeometrie geliefert. Zusätzlich bewirkt die Angabe `<Scale>15000</Scale>`, `<Scale>35000</Scale>` und `<Objektart>Siedlungsfläche</Objektart>`, dass an die Gebäudebeschreibung Informationen zu den verknüpften Objekten in den Maßstäben 1:15k und 1:35k sowie der Objektart `'Siedlungsfläche'` in dem Maßstab 1:8k angehängt werden.

Es ergibt sich hieraus die Antwort des MR-WFS in Beispiel 6.12. Es wird das Objekt `'GebäudeLoD1.10'` ausgegeben und gleichzeitig die Information, dass dieses mit den Objekten `'GebäudeLoD2.112'`, `'GebäudeLoD3.34'` sowie `'SiedlungsflächeLoD1.95'` verknüpft ist.

Das entsprechende Servlet des MR-WFS greift hierzu zunächst wieder auf die Metadatentabelle zurück, um herauszufinden, in welcher Tabelle die vertikalen Verknüpfungen zu den Objekten der Repräsentation `'GebäudeLoD0'` zu finden sind (hier `'GebäudeLinks'`). Außerdem werden in der Metadatentabelle die entsprechenden Repräsentationen für die Maßstäbe 1:8k (`'GebäudeLoD1'`), 1:15k (`'GebäudeLoD2'`) und 1:35k (`'GebäudeLoD3'`) gesucht.

Anschließend wird in der entsprechenden Link-Tabelle in der Spalte `'GebäudeLoD0'` der Attributwert `'13'` gesucht. In der entsprechenden Zeile werden die Werte für `'GebäudeLoD1'` gesucht, da der Zielmaßstab 1:8k

Bsp. 6.12 Antwort des MR-WFS auf eine Anfrage inklusive *<Links>* Element

```

<MRDB:Gebäude>
  <gml:featureMember>
    <MRDB:GebäudeLoD1 gid="GebäudeLoD1.10">
      <MRDB:the_geom>
        .
        .
        .
      </MRDB:the_geom>
      <MRDB:LinkedWith ScaleMin="10001" ScaleMax="25000">
        <MRDB:GebäudeLoD2 gid="GebäudeLoD2.112" />
      </MRDB:LinkedWith>
      <MRDB:LinkedWith ScaleMin="25001" ScaleMax="50000">
        <MRDB:GebäudeLoD3 gid="GebäudeLoD3.34" />
      </MRDB:LinkedWith>
      <MRDB:LinkedWith Objektart="Siedlungsfläche">
        <MRDB:SiedlungsflächeLoD1 gid="SiedlungsflächeLoD1.95" />
      </MRDB:LinkedWith>
    </MRDB:GebäudeLoD1>
  </gml:featureMember>
</MRDB:Gebäude>

```

beträgt. Es wird das Objekt 'GebäudeLoD1.1' gefunden. Außerdem werden die Werte der Spalten 'GebäudeLoD2' und 'GebäudeLoD3' ausgelesen, hier also '112' und '34'. Danach müssen die horizontalen Verknüpfungen zu der Objektklasse 'GebäudeLoD1' gesucht werden. Der MR-WFS greift erneut auf die Metadatentabelle zurück und ermittelt die horizontale Verknüpfungstabelle 'SiedlungLinksLoD1' für die Objektklasse 'GebäudeLoD1'. In dieser Tabelle wird in der Spalte 'Gebäude' die ID '10' gesucht und in der entsprechenden Zeile der Wert der Spalte 'Siedlungsfläche' ausgelesen, in diesem Beispiel '95'.

Anschließend wird der Teil des XML-Dokumentes generiert und in den Ergebnisdatensatz integriert. Somit erhalten die resultierenden Objekte zusätzliche Attribute zur Beschreibung der Verknüpfungen (siehe Beispiel 6.12).

Wird der Parameter *<Scale>* oder *<Objektart>* mit dem Wert 'all' aufgerufen, so werden alle Verknüpfungsinformationen gesammelt.

Zusätzliches Erweiterungspotenzial für den OGC-WFS

Die oben beschriebenen und implementierten Funktionalitäten des MR-WFS sind ausreichend, um die in der Bedarfsanalyse bestimmten Erfordernisse zu erfüllen. Darüber hinaus bietet der OGC-WFS jedoch weitere Funktionalitäten als Webservice-Schnittstelle.

So bietet der *Transactional-WFS* (WFS-T) beispielsweise weitere Operationen zur Bearbeitung der Daten in der Datenbank. Diese Operationen eines WFS-T sind in gleichem Maße von der besonderen Struktur einer MRDB berührt. Der OGC-WFS-T unterstützt zusätzlich zu den Basisfunktionalitäten die Operation *<Transaction>*. Eine Transaktion besteht aus einzelnen Operationen, welche Datenbankobjekte modifizieren, *<Insert>*, *<Update>* und *<Delete>*. Ähnlich den Operationen des Basic-WFS gehen auch diese von einer monorepräsentativen Datenstruktur aus.

Das Element *<Insert>* wird verwendet, um neue Objektinstanzen zu erzeugen und in der Datenbank abzulegen. Das zu bildende Objekt muss hierbei dem Objekt-Schema entsprechen, welches über *<DescribeFeatureType>* abgefragt werden kann. Die resultierende Beschreibung kann als Muster für den Insert-Prozess genutzt werden. Liegen nun multirepräsentative Daten zugrunde, so muss *<DescribeFeatureType>*, ähnlich der Operation *<GetCapabilities>*, das multirepräsentative Schema beschreiben, um das neue Objekt entsprechend einordnen zu können. Somit sollte das Element *<Insert>* die Option bereithalten, sowohl die jeweilige Objektart als auch die Auflösungsstufe zu benennen, um den Datensatz in das Datenbankschema integrieren zu können. Außerdem sollte es möglich sein, mehrere Objekte gleichzeitig einzupflegen und diese als alternative Repräsentationen eines Phänomens zu kennzeichnen.

Die Operation *<Update>* wird verwendet, um Eigenschaften eines Objektes zu ändern. Die Änderungen können auf Attribute oder aber die Geometrie eines Objektes angewendet werden. Zusätzlich besteht die Möglichkeit, mit Filtern zu arbeiten, um beispielsweise alle Objekte, welche einen bestimmten Attributwert besitzen, zu verändern. Auch hier gilt es, zusätzlich den Zugriff auf eine bestimmte Auflösungsstufe zu erlauben. In ähnlicher Weise sollte schließlich die Operation *<Delete>* ergänzt werden. Des Weiteren besteht der Bedarf, die Verknüpfungen zwischen Objekten über *<Insert>*, *<Update>* oder *<Delete>* bearbeiten zu können.

Neben dem WFS existieren einer Reihe weiterer Webservice-Spezifikationen zur Arbeit mit Geodaten. So lassen sich zum Beispiel auch die Funktionen eines OGC-WMS anpassen an eine multirepräsentative Datenbasis, um Darstellungen bestimmter Auflösungsstufen oder multiskalige Karten zu erhalten.

7 MRDB-Webdienst: Applikations-Ebene

Die Applikations-Ebene beinhaltet über das Internet verfügbare spezifische Funktionalitäten und Lösungen für bestimmte Anwendungsfälle. Der hier beschriebene Prototyp demonstriert einen auf multiskaligen Daten basierenden Geoinformationsdienst für mobile Nutzer. In dem folgenden Kapitel wird auf der einen Seite demonstriert, welche neuen Nutzungsmöglichkeiten durch die Verwendung einer MRDB innerhalb einer Webumgebung entstehen. Es ergeben sich alternative Techniken zur Bereitstellung und Visualisierung von Geoinformationen für mobile Clients. Auf der anderen Seite soll ein Testbett aufgebaut werden, welches den Informationsfluss von der Datenbank bis zum Client nachbildet, um auf diese Weise Schwächen im System aufzudecken und diese entsprechend zu korrigieren. Gleichzeitig sollen durch Vergleiche mit alternativen Workflows, wie dem Zugriff über einen OGC-WFS sowie dem Einsatz von Online-Generalisierungsverfahren, die Vorzüge des multirepräsentativen Systems aufgezeigt werden.

7.1 Systemarchitektur der Applikations-Ebene

Der MRDB-Service stellt eine Informationsbasis bereit, auf die der Client über die MR-WFS-Schnittstelle (Kapitel 6) zugreifen kann, sofern dieser über eine Internetverbindung verfügt. Die von dem Service gelieferten GML-Daten können von dem jeweiligen Anwender für seine Zwecke eingesetzt oder weiterverarbeitet werden.

Unterabschnitt 2.3.3 beschreibt eine Reihe von Anwendungen aus den Bereichen *Auskunft*, *Daten* und *Lösungen*. Das Auskunftssystem bereitet die Daten weiter auf, bevor sie an den Endkunden gesendet werden. Hier zu nennen sind beispielsweise Karten- oder Routingdienste. Daten-Services senden die Informationen direkt an das System des Kunden, welcher die Daten für eigene Analysen oder Planungsarbeiten einsetzt. Die dritte Kategorie umfasst maßgeschneiderte Lösungen für ein individuelles Problem des Kunden.

Der hier beschriebene Prototyp deckt die beiden Bereiche *Auskunft* und *Daten* ab. Der Datendienst in Form des MR-WFS wird in dem vorangegangenen Kapitel beschrieben. Der Auskunftsdienst, eine Art Mapservice, stellt für den Kunden eine Reihe von Funktionalitäten bereit, um diesen mit Karten und Informationen zu bestimmten Objekten zu versorgen. Dieser Teil des Systems arbeitet nach dem Prinzip des *Thin Client*. Es wird versucht, die Hauptarbeit auf die Seite des Servers zu verlagern, um die rechentechnisch schwachen Geräte der Klienten zu entlasten. Der Kartendienst greift seinerseits auf einen Datendienst, den MR-WFS, zurück. In dem hier vorgestellten Prototypen (vgl. Abbildung 7.1) bedient der Mapservice vorrangig mobile Geräte. Diese besitzen Beschränkungen bezüglich ihrer Rechenleistung, der Größe und Auflösung ihrer Displays sowie der Bandbreite der Internetanbindung. Daher besteht das Ziel dieses Dienstes darin, den Client rechentechnisch zu entlasten, das Kartenbild an das kleine Display anzupassen sowie der geringeren Bandbreite der Internetverbindung durch eine entsprechende Reduktion der Datenmengen gerecht zu werden. Wenn in dem genannten Szenario von einem Client die Rede ist, so ist damit immer die Applikation gemeint, welche auf dem mobilen Gerät installiert ist, mit dem der Mapservice kommuniziert und auf das der Endkunde Zugriff hat.

Eine zweite Anwendung, welche Funktionalitäten zur Informationssuche und -analyse bieten soll, wird in ein bestehendes GIS (JUMP-GIS) integriert und ruft die benötigten Daten direkt über die MR-WFS Schnittstelle ab. Die Funktionalitäten zur Bearbeitung der Daten sind in diesem *Thick-Client*-Szenario in das GI-System des Anwenders implantiert.

Die Client-Ebene mit ihren unterschiedlichen Plattformen und Funktionalitäten wird in dem nachfolgenden Kapitel 8 beschrieben. In diesem Kapitel erfolgt zunächst die Darstellung eines exemplarischen Kartendienstes, welcher bestimmte Funktionalitäten für mobile Nutzer bereithält.

7.2 Ein Webmapservice für mobile Clients

7.2.1 Architektur eines Webmapservice

Die Architektur des gesamten hier beschriebenen Prototypen ist in der Abbildung 7.1 skizziert. Der Aufruf eines Webdienstes erfolgt, wie im Abschnitt 2.2 ausführlich dargelegt wurde, über einen sogenannten *HTTP-Request*.

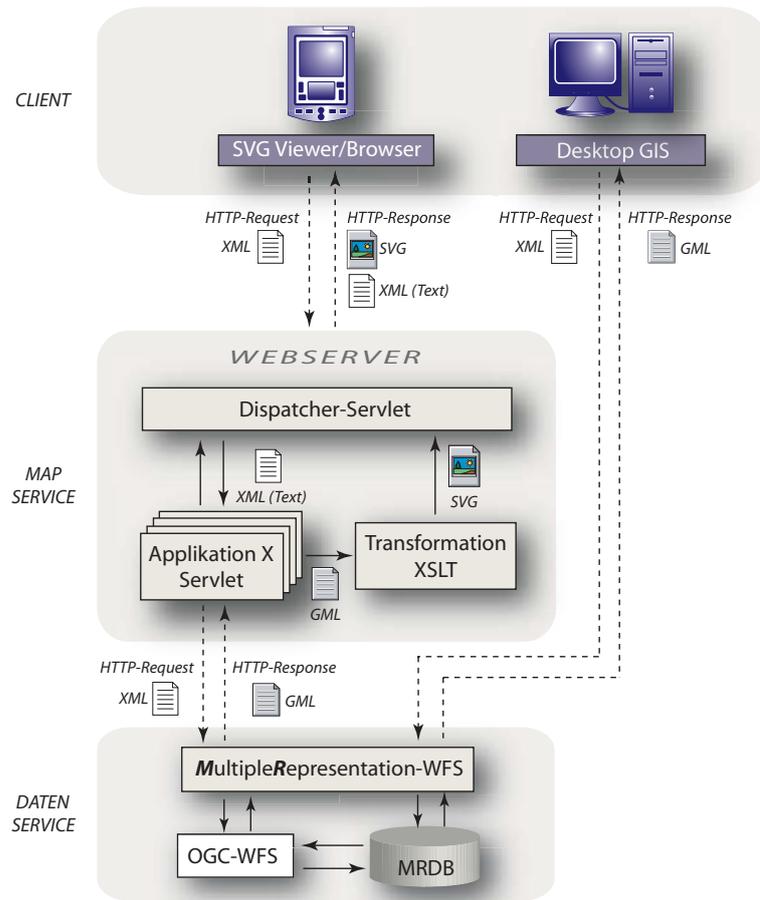


Abb. 7.1: Systemarchitektur und Datenfluss der Applikations-Ebene

In dem hier beschriebenen Prototypen eines Kartendienstes ruft der mobile Nutzer, also zum Beispiel die Applikation des Feuerwehrmannes oder der Studentin in den beiden Szenarien, mit Hilfe eines HTTP-Requests das entsprechende Servlet inklusive der benötigten Parameter auf, wie zum Beispiel:

```
http://www.myMapService.de/serviceDispatcher?application=getmap&scale=1000&layer=Gebäude
```

Dieser Aufruf enthält, wie in dem Beispiel abzulesen ist, die Adresse des Dienstes. Hinter der Adresse werden, gekennzeichnet durch ein '?', die notwendigen Parameter zum Aufruf des Dienstes angehängt. Die Anfrage erfolgt hier also über HTTP-GET. In diesem Fall wird die Applikation 'getmap' gewünscht, welche die Parameter 'scale' und 'layer' erfordert.

Dieser Aufruf wird von dem Webserver interpretiert und an das entsprechende Servlet des Applikations-Servers weitergeleitet. Ein sogenanntes *Dispatcher-Servlet* nimmt die Anfrage entgegen und entscheidet auf Grund der in dem Aufruf enthaltenen Parameter, welche der vorhandenen Funktionalitäten beansprucht werden. Das für diese Funktion verantwortliche Servlet baut eine Verbindung zum MR-WFS auf, sammelt durch eine entsprechende Anfrage an den Service die benötigten Daten und bereitet diese auf. Nach Fertigstellung der Prozesse übergibt das Dispatcher-Servlet sein Resultat an den Webserver, welcher dieses in sein *HTTP-Response-Dokument* integriert, dieses als *XML-Mime-Type* kennzeichnet und die Antwort an den aufrufenden Client sendet. Die Clientapplikation kann die resultierende SVG-Karte oder das XML-Dokument interpretieren und auf dem Display des mobilen Gerätes darstellen oder die Informationen für einen späteren Aufruf zwischenspeichern.

Der hier beschriebene Webmap-Service steht stellvertretend für die in den Szenarien geforderten Stadt- und Feuerwehrinformationsdienste, welche die höchste Stufe einer durch neue Technologien erreichbaren Erleichterung der beschriebenen Tätigkeiten repräsentieren. Der Service führt bestimmte Aufgaben und Prozesse im Auftrag des Client durch, so dass diesem Arbeit abgenommen wird. Der Client, in diesem Fall die Applikation auf dem PDA des Feuerwehrmannes oder von Christa, muss lediglich die entsprechende, in der Situation relevante Funktion aufrufen und den Ergebnisdatensatz darstellen.

7.2.2 SVG und XSLT

Der Service soll den Client mit digitalen Karten unterschiedlicher Maßstäbe sowie zusätzlichen Informationen zu einzelnen Objekten versorgen. Eine Codierung der Karten in Form von Vektordaten wird diesen Ansprüchen gerecht. Die Vorteile der Vektordaten, im Gegensatz zu Rasterbildern, liegen zum einen in der hohen grafischen Qualität, welche sich auch beim Vergrößern der Darstellung nicht verschlechtert. Zum anderen besteht bei Daten im Vektorformat die Möglichkeit, einzelne Objekte auszuwählen, um Informationen zu diesen abfragen zu können.

Um nun Karten im Vektorformat an den Client übertragen zu können wird das SVG-Format genutzt. Die SVG-Spezifikation (Ferraiolo & Jackson 2003) wurde im Jahre 2001 erstmals von dem W3C als Empfehlung vorgestellt. Dieses auf XML-Syntax aufbauende Format wurde zur Darstellung von 2D-Vektorgrafiken in Form von Pfaden, Bildern oder Texten entwickelt. Die SVG-Bilder sind interaktiv, der Nutzer kann also mit einzelnen Objekten durch Mausklicks interagieren, und dynamisch, so dass Animationen in die Graphik eingebunden werden können. Das SVG-DOM¹ ist kompatibel und konsistent zum HTML-DOM, so dass sich der SVG-Code optimal in den HTML-Code integrieren lässt. SVG kann in Verbindung mit Skriptsprachen eingesetzt werden, um anspruchsvolle Applikationen zu realisieren. Weiterhin ist die Darstellung der SVG-Graphik in einem Webbrowser möglich. Die aktuellen Versionen von *Firefox*, *Opera*, *Safari* und *Camino* bringen native Unterstützung für SVG. Andere Browser werden durch ein zusätzliches Plugin SVG-fähig. *Opera-Mini*, ein Webbrowser für mobile Geräte, bietet ebenfalls eine, wenn auch eingeschränkte, Unterstützung zur Darstellung von SVG-Grafiken. Alternativ hierzu existieren eine Reihe von reinen SVG-Viewern².

Da auch die von dem MRDB-Service ausgegebenen GML-Daten auf einer XML-Syntax aufbauen, lässt sich der GML-Datensatz durch Transformation in einen SVG-Datensatz umformen. Hierzu wird ein entsprechendes Stylesheet definiert, basierend auf der XSL³, welches Informationen darüber enthält, auf welche Weise die Objekte des GML-DOMs in das zukünftige SVG-DOM übersetzt werden sollen. Somit lässt sich in diesem Fall mit Hilfe der XSL die Darstellung, also Farbe, Form, Opazität etc., manipulieren. Es ist auch möglich, mehr als ein XSL-Dokument vorzuhalten, um je nach Vorlieben des Client oder je nach Inhalt der Karte ein bestimmtes Layout zu wählen. Durch die XSLT wird, den Vorschriften des XSL-Dokumentes folgend, der GML-Baum in einen SVG-Baum umgewandelt, indem GML-Objekte durch entsprechende SVG-Objekte ausgetauscht werden. Das resultierende SVG-Dokument wird schließlich von dem aufgerufenen Servlet als Ergebnisdatensatz an den Client zurückgegeben.

7.2.3 Funktionalitäten

In diesem Abschnitt folgt eine nähere Betrachtung dessen, was in der Abbildung 7.1 allgemein als „Applikation X Servlet“ gekennzeichnet ist. Die einzelnen Funktionalitäten sind aus den Bedürfnissen heraus motiviert, welche die Anwendungsszenarien hervorgebracht haben (vgl. Unterabschnitt 4.1.1, S.83). Grundsätzlich besteht der Bedarf, die in der MRDB gesammelten und über den MR-WFS abzurufenden Daten in entsprechender Weise zur Visualisierung aufzubereiten. Daher ist zunächst die Abfrage einer einfachen Kartendarstellung notwendig, deren Inhalt sich durch bestimmte Parameter steuern lässt. Dies soll durch die Funktion 'GetMap' realisiert werden. Desweiteren werden übersichtliche Visualisierungen der Geoinformation auf dem kleinen Display des mobilen Gerätes gefordert. So benötigt Christa zum Beispiel zur Orientierung eine angemessene Darstellung der gesamten Wegstrecke und gleichzeitig eine detaillierte Darstellung ihrer näheren Umgebung. Um verschiedene Auflösungsstufen in einer Karte zu kombinieren wird die Funktion 'MultiscaleMap' eingeführt. Um dabei jedoch nicht nur in einem bestimmten Bereich den Detailreichtum zu erhöhen, sondern eine Karte zu erhalten, die zusätzlich den Maßstab variiert, so dass eine Art Lupeneffekt entsteht, soll die Funktion 'Lupe' zusätzlich eine lokale Vergrößerung des Kartenbildes berechnen. Schließlich wird in beiden Szenarien eine Funktionalität gefordert, welche es erlaubt, entlang der in der MRDB vorliegenden Verknüpfungen nach Informationen zu suchen. Das 'InformationDrilling' soll dem Nutzer erlauben, Repräsentationen und Informationen verknüpfter Objekte abfragen zu können. Christa benötigt Informationen über die Wohngegend, welche sie über das Gebäude abfragen möchte. Der Feuerwehrmann benötigt den Gebäudeplan zu einem Haus oder möchte wissen, wieviele Einwohner der Stadtteil besitzt, in dem das brennende Haus steht.

Der hier beschriebene Service liefert grundsätzlich auf Anfrage Geodaten in Form eines SVG-Datensatzes oder Zusatzinformationen in Form eines XML-Textdokumentes. Die entsprechenden Daten werden durch Zugriff auf den MR-WFS gesammelt und aufbereitet.

¹Document Object Model

²Eine Liste verfügbarer SVG-Viewer: <http://www.svgi.org/>

³Extensible Stylesheet Language

GetMetaData

Der Client kommuniziert in dem vorgestellten Szenario ausschließlich mit dem Kartendienst, auch wenn dieser die Möglichkeit besitzt, Informationen direkt von dem MR-WFS Datendienst abzurufen. Zur Abfrage von Metainformationen kann der Anwender beispielsweise direkt die Funktion `<GetCapabilities>` des MR-WFS aufrufen. Der Kartendienst bietet jedoch unter anderem den Vorteil, dass lediglich die Informationen gefiltert und aufbereitet werden, welche für den Nutzer relevant sind. Der Client kann die an ihn gesendeten Informationen direkt verwenden, ohne selbst eine Aufbereitung der Daten vornehmen zu müssen.

Eine auf einem mobilen Gerät installierte Applikation möchte dem Nutzer je nach Anwendungsfall bestimmte Kartenebenen, also eine Auswahl bestimmter Objektarten, sowie Werkzeuge zur Navigation in der Karte, wie Zoom (Vergrößern/Verkleinern) und Pan (Verschieben), bereitstellen. Daher benötigt diese zunächst Informationen über die verfügbaren Kartenlayer sowie die empfohlenen Maßstabsbereiche der einzelnen Repräsentationen. Auf diese Weise können die Datensätze in den entsprechenden Maßstabsbereichen verwendet und neue Datensätze nachgeladen werden, falls der Bereich durch Zoom-Aktivitäten verlassen wird (vgl. Kapitel 8). Die Applikation sendet daher folgende Anfrage an den Mapservice:

```
http://www.myMapService.de/serviceDispatcher?application=getmetadata
```

Der Mapservice sendet seinerseits eine Anfrage `<GetCapabilities>` an den MR-WFS zur Abfrage der Daten aus der MRDB, filtert die Informationen zu den verfügbaren Ebenen (Gebäude, Siedlungsfläche, Ortslage und Straßen) sowie die jeweiligen Maßstabsgrenzen `'scaleMin'` und `'scaleMax'` jeder Repräsentation aus dem Ergebnisdatensatz und bereitet diese Informationen in einem neuen XML-Dokument (siehe Beispiel 7.1) auf, welches an die Client-Applikation gesendet wird.

Bsp. 7.1 HTTP-Response des Webservices auf die Anfrage `GetMetaData`

```
<?xml version="1.0" encoding="UTF-8"?>
<MetaData xmlns="http://www.mapservice.de/service">
  <LayerList>
    <Layer>
      <name>Gebäude</name>
      <Representation>
        <scalemin>1000</scalemin>
        <scalemax>5000</scalemax>
      </Representation>
      <Representation>
        <scalemin>5001</scalemin>
        <scalemax>10000</scalemax>
      </Representation>
      <Representation>
        ...
      </Representation>
    </Layer>
    <Layer>
      <name>Straßen</name>
      <Representation>
        <scalemin>1000</scalemin>
        <scalemax>10000</scalemax>
      </Representation>
      <Representation>
        ...
      </Representation>
    </Layer>
    ...
  </LayerList>
</MetaData>
```

Das Dokument enthält Informationen über die verfügbaren Objektarten sowie die Auflösungsstufen. Diese Informationen können in den Funktionsaufrufen (z.B. `GetMap`) verwendet werden.

GetMap

Die essenzielle Funktion des hier beschriebenen Webmapdienstes zur Abfrage einer Karte findet sich in `GetMap`, welche mit den Parametern `scale`, `layers`, `bbox`, `feature`, `linkedwith` aufgerufen wird, wobei der erstgenannte Parameter `scale` obligatorisch ist, die übrigen sind optional. Somit lässt sich im einfachsten Fall durch Angabe eines Maßstabes (`scale`) eine Karte mit allen verfügbaren Objektarten abfragen. Zusätzlich kann die Ausgabe beschränkt werden auf eine oder mehrere Objektarten durch Angabe des Parameters `layers`. Nachdem der Client den Service mit den Parametern `'application=getmap'`, `'scale=9000'` und `'layers=Gebäude'` aufgerufen

hat, liest das Dispatcher-Servlet, welches alle Anfragen entgegennimmt, die Anfrageparameter aus und übergibt diese an die entsprechende Javaklasse, in diesem Fall die Klasse 'GetMap'. Diese stellt nun ein XML-Dokument zusammen, welches als Anfrage per HTTP POST an den MR-WFS gesendet wird. In diesem Fall werden Gebäudedaten für den Maßstab 1:9k benötigt. Daher wird folgende Anfrage zusammengestellt:

```
<GetAlternativeFeature>
  <Query typeName=Gebäude scale=9000>
  </Query>
</GetAlternativeFeature>
```

Das von dem MR-WFS zurückgelieferte GML-Dokument wird von einer separaten Klasse in ein SVG-Dokument transformiert und an das Dispatcher-Servlet übergeben, welches dieses dann an den Client zurücksendet. Auch durch die Verwendung eines OGC-WFS oder WMS lassen sich Kartendaten abfragen und nutzen. Der Nutzer erhält auf diese Weise jedoch keine Informationen über die verfügbaren Auflösungsstufen und findet darüberhinaus keine Möglichkeit, auf alternative Repräsentationen zugreifen zu können.

Eine weitere Filterung erfolgt durch Aufruf einer begrenzenden Bounding-Box (BBOX), wobei hier neben den vier Koordinatenwerten für das entsprechende Rechteck ein fünfter Wert angegeben werden kann, welcher angibt, auf welchen Datensatz (Maßstab) sich diese Bounding-Box beziehen soll (vgl. Abbildung 6.4 und Beispiel 6.8, S.131). Ein Problem mobiler Geräte besteht in der geringen Speicherkapazität. Daher ist es sinnvoll, jeweils nur den aktuell benötigten Kartenausschnitt in dem Speicher vorzuhalten und bei Bedarf einen neuen Ausschnitt nachzuladen. Mit Hilfe des Parameters 'bbox' ist es nun möglich, anhand des aktuellen Maßstabes sowie der Bildschirmgröße den benötigten Kartenausschnitt zu berechnen und diesen mit Hilfe der Bounding-Box abzufragen.

Durch Verwendung des Parameters 'feature' zusammen mit den IDs der Objekte, die von Interesse sind, erhält der Client eine Repräsentation dieser Objekte in dem mit 'scale' definierten Maßstab. Der Feuerwehrmann arbeitet beispielsweise derzeit in einer Übersichtskarte mit dem Maßstab 1:40k und möchte mehr Details zu bestimmten Gebäuden erhalten. Er wählt durch Anklicken mehrere generalisierte Gebäude aus, deren IDs die Applikation in einer Liste speichert. Der Client benötigt nun einen Service, welcher eine Abfrage hochauflösender Darstellungen dieser Gebäude ermöglicht. Hier kann wiederum der Map-Service mit seiner Applikation 'GetMap' aufgerufen werden. In diesem Fall wird zusätzlich der Parameter 'feature=GebäudeLoD3.100,GebäudeLoD3.101' angegeben, welcher die Anfrage auf die beiden genannten Objekte beschränkt. Der Mapservice kann seinerseits auf die Funktion <GetAlternativeFeature> des MR-WFS zurückgreifen, welcher die Filterfunktion <ObjectRepresentation> bietet, mit Hilfe derer einzelne Objekte des Zielmaßstabes gefiltert werden können, welche mit den angegebenen Objekten verlinkt sind. Somit stellt der Kartendienst folgende Anfrage an den MR-WFS:

```
<GetAlternativeFeature>
  <Query typeName=Gebäude scale=9000>
    <Filter>
      <Or>
        <ObjectRepresentation>gebäudeLoD3.100</ObjectRepresentation>
        <ObjectRepresentation>gebäudeLoD3.101</ObjectRepresentation>
      </Or>
    </Filter>
  </Query>
</GetAlternativeFeature>
```

In diesem Fall werden also die Gebäude für den gewünschten Maßstab zurückgegeben, welche mit den Objekten 100 und 101 des LoD3 verlinkt sind und daher eine alternative Repräsentation darstellen.

Schließlich lassen sich zusätzlich zu den eigentlichen Objektdaten auch Informationen zu den IDs alternativer Repräsentationen abfragen durch Benutzung des Parameters 'linkedwith' und der zusätzlichen Angabe der gewünschten Maßstäbe (z.B. 'linkedwith=50000'). Die IDs der verknüpften Objekte werden zusammen mit dem eigentlichen Objekt in dem SVG-Datensatz abgelegt. Diese Information ist somit als zusätzliches Attribut zu einem bestimmten Objekt codiert und kann vom Nutzer jederzeit abgerufen werden. Diese Information kann zum Beispiel von einer Applikation verwendet werden, um zwei parallel vorgehaltene Repräsentations-Ebenen in Beziehung zu setzen (vgl. Unterabschnitt 8.1.1).

MultiscaleMap

Die in den folgenden Abschnitten beschriebenen Funktionalitäten 'MultiscaleMap' und 'Lupe' zeigen die Möglichkeiten multiskaliger Datensätze zur Verbesserung des Kartenbildes gerade auf kleinen Displays. In dem Abschnitt 2.5 wurden Verfahren zur Aufwertung der Geodatenvisualisierung auf mobilen Geräten durch Adaption des Karteninhaltes beschrieben. Dadurch, dass die Karte in dem Moment generiert wird, in dem diese benötigt wird, ergeben sich hier neue Chancen zur optimalen Anpassung dieser. Auch die Verwendung unterschiedlicher

Auflösungsstufen bietet, wie in dem Abschnitt 2.5 und Unterabschnitt 3.3.1 angedeutet wurde, eine Möglichkeit, dem Informationsüberschuss herkömmlicher Karten entgegenzuwirken, um die kleine Darstellungsfläche mobiler Geräte zu entlasten.

Die Verfügbarkeit unterschiedlich aufgelöster Datensätze ermöglicht nun, geringer aufgelöste Repräsentationen für Objekte von geringem Interesse zu verwenden und gleichzeitig hochauflösende Daten zur Darstellung von Objekten einzusetzen, welche den Nutzer aktuell interessieren. Auf diese Weise wird die dem Informationskanal angemessene Datenmenge produziert, indem überflüssige Informationen, respektive Details, entfernt werden. Die Möglichkeit, Objekte unterschiedlich stark zu generalisieren, um wichtige Objekte hervorzuheben, wird von Sester (2002) beschrieben. So sollen durch unterschiedliche Verwendung von Generalisierungsoperationen, wie Verkleinern, Vereinfachen, Zusammenfassen und Verdrängen, unwichtige Objekte optisch in den Hintergrund gedrängt und wichtige Objekte betont werden. Die Funktion 'MultiscaleMap' stellt Objekte in einem bestimmten Umkreis um eine vorgegebene Position mit einer höheren Auflösung dar als die übrigen Objekte. Diese Position betrifft üblicherweise entweder den aktuellen Standort des Nutzers oder aber das Zentrum der Karte. Objekte in direktem Umkreis um diesen Punkt sind üblicherweise von höherem Interesse für den Nutzer.

Abbildung 7.2 (links) zeigt eine solche multiskalige Karte, die der Mapservice nach folgender Anfrage liefert:

```
http://www.myMapService.de/serviceDispatcher?application=multiscalemap&scale1=1000&scale2=25000&position=x,y&radius=100&layer=Gebäude
```

Die Funktion bedingt die Parameter `scale1` und `scale2`, welche die beiden Auflösungsstufen festlegen, `position` und `radius` definieren den Kreis, innerhalb dessen die hochauflösenden Daten zu finden sind. Zusätzlich besteht die Möglichkeit, ähnlich wie bei der Funktion `GetMap`, die Ausgabe durch eine Bounding-Box (`bbox`), Objektfilter (`feature`) oder Objektartenfilter (`layers`) zu reduzieren.

So benötigt Christa (Szenario 2) zur Navigation eine Karte, welche ihr die gesamte Wegstrecke übersichtlich darstellt. Diese Darstellung erfolgt daher mit weniger Details, um die gesamte Strecke in einem kleinen Maßstab darstellen zu können. Gleichzeitig benötigt sie jedoch an bestimmten Stellen zur Orientierung mehr Details. Daher sollte die Applikation eine Karte generieren, welche in der näheren Umgebung um die aktuelle Position mehr Details darstellt als in der übrigen Karte. Hierzu wird ein Service benötigt, welcher eine derartige Karte generieren kann sowie eine Datenbasis, welche mehrere Auflösungsstufen vorhält. Beides ist hier vorhanden. Die mobile Applikation auf dem PDA ermittelt jeweils die aktuelle Position und sendet eine Anfrage an den Mapservice mit den Parametern 'application=multiscalemap', 'scale1=1000', 'scale2=25000', 'layers=Gebäude', 'position=x,y' sowie 'radius=50'.

Das Mapservice generiert hierzu zwei Anfragen an den MR-WFS. In einem ersten Aufruf werden die geringer aufgelösten Daten abgefragt, beispielsweise durch:

```
<GetAlternativeFeature>
  <Query typeName=Gebäude scale=25000>
  </Query>
</GetAlternativeFeature>
```

Durch eine geometrische Operation *touches* werden anschließend alle Objekte bestimmt, welche sich innerhalb des angegebenen Radius um die definierte Position befinden. Die IDs dieser Objekte werden ausgelesen, bevor diese dann aus dem Datensatz gelöscht werden. Finden sich z.B. die Objekte 10, 20, 30 und 40 innerhalb des Radius, so wird ein zweiter Datensatz mit höherer Auflösung (1:1k) angefordert, wobei die IDs der gelöschten Objekte geringerer Auflösung verwendet werden (in dem Beispiel 10, 20, 30, 40), um die mit diesen verknüpften Objekte höherer Auflösung abzufragen:

```
<GetAlternativeFeature>
  <Query typeName=Gebäude scale=1000>
  <Filter>
    <Or>
      <ObjectRepresentation>GebäudeLoD2.10</ObjectRepresentation>
      <ObjectRepresentation>GebäudeLoD2.20</ObjectRepresentation>
      <ObjectRepresentation>GebäudeLoD2.30</ObjectRepresentation>
      <ObjectRepresentation>GebäudeLoD2.40</ObjectRepresentation>
    </Or>
  </Filter>
</Query>
</GetAlternativeFeature>
```

Der Ergebnisdatensatz wird mit den verbliebenen Objekten der ersten Anfrage in dem GML-Dokument kombiniert und in ein SVG-Dokument transformiert (siehe Abbildung 7.2, links).

Neben dem Austausch von Objekten innerhalb eines bestimmten Umkreises um eine Position ist es auch denkbar, dem Ansatz von Sester (2002) zu folgen und einzelne Objekte auszuwählen, um diese durch höher aufgelöste

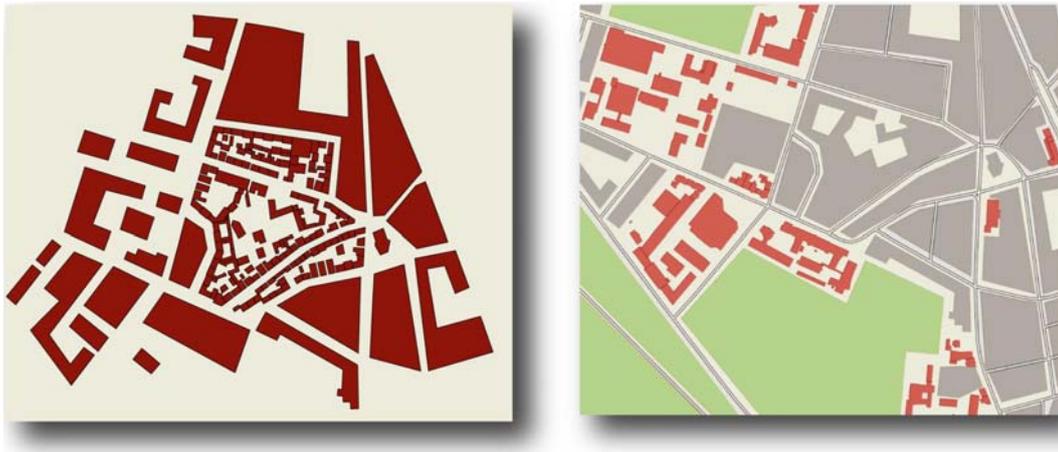


Abb. 7.2: Kombination von Daten unterschiedlicher geometrischer Auflösung in einer Karte: Zur Visualisierung von Objekten in der näheren Umgebung (links) oder zur Visualisierung von wichtigen Objekten, z.B. Unigebäuden (rechts)

Repräsentationen zu ersetzen. Je nach Thema der Karte können somit wichtige Objekte hervorgehoben werden und gleichzeitig mehr Informationen liefern als andere Objekte. So kann beispielsweise eine Standortkarte der Universität die zu dieser gehörigen Gebäude mit einer höheren Auflösung darstellen. Somit ist es neben einer Betonung dieser Objekte auch leichter, diese Gebäude in der Realität zu identifizieren (vgl. Abbildung 7.2, rechts) (Hampe et al. 2005). Diese Form der Darstellung kann auf die gleiche Weise zur Visualisierung von Landmarken, also markanten Objekten, in Karten zur Routenführung verwendet werden (Elias et al. 2005). Der Kartenservice muss lediglich die IDs der zu betonenden Objekte ermitteln und kann dann, entsprechend der oben aufgeführten Anfrage, alternative Repräsentationen aus der MRDB abfragen und in das Kartenbild integrieren. Sollen einzelne Objekte betont werden, wie in der Abbildung 7.2 (rechts) zu sehen ist, so genügt es jedoch nicht, die Daten mit geringem LoD durch die Daten mit hohem LoD zu ersetzen, da die Objekte mit geringem LoD (z.B. Gebäudeblöcke) mehr als ein Objekt der höheren Auflösungsstufe (z.B. Einzelgebäude) repräsentieren. Sollen daher nur ein oder wenige hochauflösende Objekte dargestellt werden, so müssen an dieser Stelle die Objekte geringer Auflösung mit Hilfe der übrigen, nicht dargestellten hochauflösenden Objekte neu gebildet werden. Die Verknüpfungen helfen hier, diese verbleibenden Objekte zu finden. Enthält die Verknüpfung darüber hinaus Informationen über die zur Generierung des niedrigen LoD notwendigen Verfahren (hier: Vereinfachen und Zusammenfassen) und Parameter, so lassen sich derartige Darstellungen gemischter Auflösung problemlos aus der MRDB ableiten.

Lupe

Die in dem vorangegangenen Abschnitt beschriebenen Karten erfüllen streng genommen noch nicht den Begriff einer multiskaligen Karte, da hier zwar Daten unterschiedlicher Auflösung dargestellt werden, der Maßstab der Karte jedoch konstant bleibt. Ähnlich dem vorher beschriebenen Ansatz versuchen sogenannte *polyfokale* Darstellungen die wichtigen, oftmals zentralen Teile der Karte, vergrößert darzustellen. Durch diese Verzerrung des Kartennetzes entsteht eine Art Lupeneffekt, bei dem ein Teil der Karte vergrößert dargestellt wird (vgl. Abschnitt 2.5). Während vorangegangene Arbeiten alternative Projektionen verwenden, um geografische Koordinaten in die Ebene abzubilden (z.B. Fairbairn & Taylor 1995), verwenden Harrie et al. (2002) einen Algorithmus, welcher bestehende ebene Koordinaten verwendet und diese entsprechend verändert. Auf diese Weise können die Daten der oben beschriebenen Karte (Abbildung 7.2, links) verwendet werden, um aus diesen eine Darstellung zu berechnen, wie sie in Abbildung 7.3 zu erkennen ist.

Der Kartenleser erhält auf der einen Seite eine vergrößerte Darstellung im Zentrum der Karte und auf der anderen Seite eine verkleinerte Übersicht zum Rand der Karte hin. Auf diese Weise können auch auf kleinen Displays alle notwendigen Informationen in der Karte dargestellt werden. Durch eine Verkleinerung des Maßstabes zum Kartenrand hin mit gleichzeitiger Verringerung der Details erhält der Nutzer eine Übersicht über ein größeres Gebiet und zugleich genügend Details, um sich in seiner näheren Umgebung orientieren zu können. Die entsprechenden Verfahren zur Berechnung einer solchen Verzerrung sind in Harrie et al. (2002) sowie Hampe et al. (2004) beschrieben.

Die Funktion 'Lupe' des Mapservices wird über dieselben Elemente gesteuert, wie die zuvor beschriebene Funktion 'MultiscaleMap'. Zusätzlich wird ein zweiter Radius (`radius2`) benötigt, welcher den Beginn des

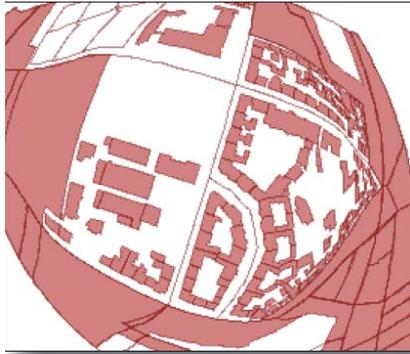


Abb. 7.3: Karte mit Daten unterschiedlicher Auflösung und zusätzlicher polyfokaler Verzerrung (aus (Hampe et al. 2004))

geringeren Maßstabes angibt. Das Kartenbild wird in drei Bereiche unterteilt, wobei in einem bestimmten Umkreis (**radius1**) um den definierten Standort die Daten in einem großen Maßstab dargestellt werden. Zwischen diesem Umkreis und dem zweiten definierten Umkreis (**radius2**) mit demselben Mittelpunkt ist der Bereich definiert, in dem ein linearer Übergang zwischen dem großen und dem kleinen Maßstab stattfindet. Außerhalb des zweiten Umkreises ist dann der konstante, kleinere Maßstab zu finden.

Der Mapservice berechnet nach Fertigstellung der multiskaligen Karte nach dem obigen Verfahren eine Verzerrung der Karte. Auf diese Weise erhält *Christa* nicht nur mehr Details der aktuellen Umgebung. Vielmehr lassen sich diese auch besser erkennen als bei einer bloßen Erhöhung der geometrischen Informationsmenge. Idealerweise verfolgt die Applikation die Bewegung des Nutzers und fragt nach einer gewissen Wegstrecke eine neue Karte mit neuem Zentrum der Lupe ab.

InformationDrilling

Die in einer MRDB vorhandenen multiplen Repräsentationen eignen sich sehr gut zu Analyse Zwecken, wie in Unterabschnitt 3.3.4 nachzulesen ist. Bernier et al. (2005) beschreiben beispielsweise das Verfahren des „Information Drillings“, also dem Bohren nach Informationen, indem die Attribute verknüpfter Objekte abgefragt werden. Auf diese Weise wird es zum Beispiel möglich, zusätzliche Informationen zu einem Objekt abzufragen. So fragt *Christa* in dem zweiten Szenario nach Zusatzinformationen zu einem bestimmten Gebäude. Die Gebäude sind in der MRDB mit Siedlungsflächen und Ortslagen verknüpft, welche Attribute über Einwohnerzahlen enthalten. Andere statistische Informationen sind denkbar, sind jedoch in der prototypischen MRDB nicht vorhanden.

Auch diese Funktionalität kann der hier beschriebene Prototyp eines Mapservices dank der Funktionalitäten des MR-WFS realisieren. Die Funktion `'application=informationdrilling'` wird zusammen mit einer Objekt-ID sowie einem interessierenden Attribut (`feature`), wie beispielsweise der Einwohnerzahl oder dem geografischen Namen, aufgerufen:

```
http://www.myMapService.de/serviceDispatcher?application=informationdrilling&
object=GebäudeLoD0.1&feature=EWZ,GN
```

Die Datenbank des hier beschriebenen Prototypen enthält beispielsweise die Objektklasse 'Siedlungsfläche', welche die für *Christa* interessante Informationen 'EWZ=Einwohnerzahl' und 'GN=geografischer Name' beinhaltet. *Christa* klickt hierzu auf das sie interessierende Gebäude und tippt auf den Info-Button. Die Applikation ruft daraufhin die hier beschriebene Funktion des Mapservice auf und übergibt diesem als Parameter den Namen des Gebäudes `'Object=GebäudeLoD0.1'`. Der Kartendienst ruft nun die Funktion `<GetAlternativeFeature>` des MR-WFS auf, wobei hier vor allen Dingen der Parameter `<Links>` interessant ist, welcher es erlaubt, Verknüpfungsinformationen zu sammeln.

Der Service ruft zunächst die mit dem relevanten Objekt verknüpften Daten ab:

```
<GetAlternativeFeature>
<Query typeName="Gebäude" scale="10000">
  <Links>
    <Objektart>all</Objektart>
  </Links>
  <Filter>
    <Or>
      <ObjectName>GebäudeLoD0.1</ObjectName>
    </Or>
  </Filter>
</Query>
</GetAlternativeFeature>
```

Der MR-WFS sammelt die IDs der verknüpften Objekte innerhalb der LoD1 auf die Weise, wie in dem Beispiel 6.11 auf Seite 133 ausführlich beschrieben wurde. In einem nächsten Schritt werden die resultierenden Siedlungsflächen und Ortslagen über eine einfache Anfrage `<GetAlternativeFeature>` abgefragt, deren Attribute 'EWZ' und 'GN' ausgelesen und in einem XML-Dokument zusammengefasst werden. Dieses Dokument wird an die aufrufende Applikation zurückgesendet, welche das Ergebnis in Form einer Tabelle darstellen kann. Auf diese Weise erfährt Christa zum Beispiel, dass die Wohnung in der 'Südstadt' liegt und dass die Südstadt 15.000 Einwohner hat.

Probleme treten auf, wenn die Siedlungsflächen oder Ortslagen keine relevanten Informationen besitzen. In diesem Fall müsste der Mapservice sich durch wiederholtes Aufrufen des MR-WFS mit jeweils verändertem Parameter 'scale' durch die verschiedenen Auflösungsstufen bewegen, um auf diese Weise die Informationen mehrerer Auflösungsstufen zu sammeln. Der Nutzer kann sich auf diese Weise beliebig horizontal oder vertikal in der Datenstruktur bewegen, um Informationen zu sammeln. Durch eine Variation des Parameters 'scale' kann der Nutzer vertikal in der Datenstruktur navigieren. Durch Verwendung der Elemente `<Links>` und `<Objektart>` innerhalb einer `<GetAlternativeFeature>` Anfrage ist eine horizontale Bewegung möglich.

An dieser Stelle ist natürlich eine Erweiterung des Systems denkbar. Zum einen können über eine Anfrage `<DescribeFeatureType>` zunächst die verfügbaren Attribute der verknüpften Objekte bestimmt werden. Die resultierende Liste kann dann mit der angefragten Information des Nutzers abgeglichen werden. Ein in das System integrierter Thesaurus kann eine Zuordnung zwischen den angefragten und den verfügbaren Begriffen herstellen, um zum Beispiel die Frage nach *Größe* mit dem Attribut *Flächeninhalt* oder *Fläche* beantworten zu können. Der hier beschriebene Prototyp wurde nicht in dieser Tiefe implementiert.

Weitere Möglichkeiten

Durch die Verwendung eines OGC-WFS oder OGC-WMS lassen sich zwar Kartendaten aus einer entsprechenden Datenbank abrufen. Informationen über den jeweiligen Maßstabsbereich jedes Datensatzes müsste der Mapservice sich jedoch aus einem von dem Datenanbieter veröffentlichten Metadatenkatalog erfragen und in sein System integrieren. Spätestens bei der Abfrage alternativer, verknüpfter Repräsentationen scheitern jedoch die Bemühungen des Serviceanbieters, da dieser keinen Zugriff auf die in der Datenbank gespeicherten Verknüpfungen erhält.

Der MR-WFS nimmt dem Serviceanbieter somit einen großen Teil des Implementierungsaufwandes ab, da dieser auf vorhandene Funktionen zurückgreifen kann. Er stellt zudem sicher, dass die benötigten Metainformationen, wie die verfügbaren Objektarten und Auflösungsstufen, aktuell sind und somit die Anfragen das erwartete Resultat liefern. Schließlich ermöglicht der MR-WFS Zugriff auf alle expliziten und impliziten Informationen einer MRDB, welche dem Nutzer ansonsten teilweise verborgen bleiben, wie beispielsweise die Attribute verknüpfter Objekte. Weiterhin bleibt der Mapservice unberührt von Änderungen innerhalb der MRDB, da der Zugriff in jedem Fall über die definierte Schnittstelle des MR-WFS erfolgt.

Die hier vorgestellten Funktionalitäten erfüllen keineswegs die vollständigen Ansprüche eines Stadt- oder Feuerwehrinformationsdienstes. Vielmehr werden hier die grundsätzlichen Bedürfnisse durch die Nutzung multiskaliger Daten befriedigt, wie es bislang in keinem Webservice realisiert wurde. Auf dem Weg zu einem vollständigen Service fehlen neben einer breiteren Datenbasis sowie der verbesserten Informationssuche, wie sie im vorangegangenen Abschnitt angedeutet wurde, vor allen Dingen die Möglichkeiten, durch Nutzerprofile die Bedürfnisse des einzelnen Nutzers stärker zu fokussieren. Die in dem beschriebenen Prototypen umgesetzten Funktionen sollen hier nur eine erste Idee geben und deren Realisierbarkeit demonstrieren.

7.3 MRDB-Service im Vergleich zur Online-Generalisierung

Neben der Nutzung multirepräsentativer Daten zur Realisierung multiskaliger Karten oder zu Analysezwecken können die verschiedenen, vorgegeneralisierten Stufen die Notwendigkeit einer Online-Generalisierung ersetzen. Bei einem Maßstabswechsel kann der Client Daten für den neuen Maßstab aus der Datenbank abfragen, deren Auflösung an die entsprechende Darstellung angepasst ist. Es stellt sich hier die Frage, ob es überhaupt notwendig ist, Daten für diesen Fall vorzubereiten und zu speichern, wenn alternativ automatische Verfahren zur Generierung dieser Daten bereitstehen.

In einem solchen Alternativszenario sind nur die am höchsten aufgelösten Ausgangsdaten bereitzuhalten und notwendige Generalisierungen zu dem Zeitpunkt der Anfrage automatisch durchzuführen. Die in Unterabschnitt 5.3.2 beschriebenen Generalisierungsverfahren zur Gebäudevereinfachung, -verschmelzung, -typifizierung

und -verdrängung sollen dazu verwendet werden, um eine Aussage darüber treffen zu können, ob eine Vorverarbeitung der Daten aus den folgenden Gründen sinnvoll ist:

- Das Verfahren ist grundsätzlich zu zeitaufwändig
- Das Verfahren ist für eine bestimmte Datenmenge zu zeitaufwändig
- Die Ergebnisse bedingen eine manuelle Nachbearbeitung

Hierzu wird ein 1,5 km x 1,5 km großer Ausschnitt des Testgebietes *Hannover-Nordstadt* (vgl. Abbildung 5.1) beziehungsweise für die Typifizierung ein 4 km x 4 km großer Ausschnitt des Testgebietes *Flensburg* (vgl. Abbildung 5.20) verwendet. Die Gebäudevereinfachung und -verschmelzung wird für die Maßstäbe 1:10k bis 1:30k durchgeführt. Die Größe des Testgebietes deckt in dem Maßstab 1:30k in etwa die Größe des Displays eines mobilen Gerätes ab (5 cm x 5 cm). Die Typifizierung wird, da dieses Verfahren vor allen Dingen in mittleren Maßstäben einzusetzen ist, für die Maßstäbe 1:10k bis 1:75k getestet. Auch hier ergibt sich eine Darstellung des 4 km x 4 km großen Testgebietes auf etwa 5 cm x 5 cm.

7.3.1 Test Gebäudevereinfachung

Inhalt	Minimaldim. (m)	Gebäude	Punkte	Punkte/Gebäude	Punkte eliminiert	Laufzeit (msek)	Laufzeit/Objekt (msek)	Laufzeit/Punkt (msek)	Zielmaßstab
Gebäude höchster LoD	3,0				9901	125	0,10	0,007	1:5k-1:10k
	5,0	1281	17131	13	11287	125	0,10	0,007	1:10k-1:20k
	10,0				13971	109	0,09	0,006	1:20k-1:25k
Zusammengefasste Gebäude	3,0	212	20986	99	10125	127	0,81	0,008	1:25k-1:30k
	10,0	326	24212	74	6048	205	0,63	0,008	1:30k-1:40k

Tab. 7.1: Laufzeittests der Gebäudevereinfachung bei originären und zusammengefassten Gebäude

Aus den Ergebnissen der Tabelle 7.1 lässt sich ablesen, dass die Prozessierungszeit des Algorithmus zur Gebäudevereinfachung in der Hauptsache von der Anzahl der Punkte im Datensatz abhängt. Bei der Messung ergeben sich maximal 0,008 Millisekunden pro Punkt, unabhängig davon, ob der Punkt nun eliminiert werden muss oder nicht. Insgesamt benötigt der Algorithmus zur Generalisierung der Daten etwa 0,1 bis 0,2 Sekunden. Folgt man der in Unterabschnitt 5.3.2 beschriebenen Empfehlung, benachbarte Gebäude erst zusammenzufassen, bevor diese vereinfacht werden, so ergeben sich ähnliche Werte, wie in den letzten beiden Zeilen der Tabelle zu erkennen ist. Die Zeit zur Verschmelzung der Gebäude muss hier natürlich noch addiert werden.

7.3.2 Test Gebäudezusammenfassung

Tabelle 7.2 zeigt die Ergebnisse der Gebäudezusammenfassung desselben Testdatensatzes, wie in den zuvor beschriebenen Tests zur Gebäudevereinfachung. Der Prozess wird stark beschleunigt durch die Restriktion, dass nur Objekte innerhalb einer Straßenmasche verschmolzen werden dürfen. Bei kleiner werdendem Maßstab werden mehr Gebäude verschmolzen, so dass sich eine höhere Laufzeit ergibt, da der Prozess der Geometriverschmelzung zusätzliche Rechenzeit beansprucht. Insgesamt ergibt sich eine Laufzeit von 10 bis 12 Sekunden für den Testdatensatz. Bei steigender Zahl von Objekten innerhalb einer Masche steigt auch die Prozessierungszeit. Liegen alle 1281 Gebäude innerhalb einer Masche, so steigt die Rechenzeit auf über drei Minuten an.

7.3.3 Test Typifizierung und Verdrängung

Tabelle 7.3 zeigt die Ergebnisse der Laufzeitmessung zur Typifizierung und Verdrängung. In den Tests werden die Reduktionsrate sowie die Größe des Testgebietes variiert. Der Testdatensatz umfasst ein Gebiet lockerer Bebauung (siehe Abbildung 5.21) und besitzt in den ersten Tests eine Größe von 4 km x 4 km und in einem

Inhalt	Minimalabst. (m)	Straßenmaschen	Gebäude vor der Gen.	Punkte vor der Gen.	Gebäude nach der Gen.	Punkte nach der Gen.	Laufzeit (msek)	Laufzeit/Objekt (msek)	Zielmaßstab
Gebäude höchster LoD	3,0	1	1281	17131	320	9489	180125	140,61	1:10k
	3,0	25	640	8156	163	4258	6324	9,74	1:10k
	3,0				326	9685	12359	9,65	1:10k
	5,0	51	1281	17131	212	8365	11797	9,21	1:15k
	10,0				148	5257	13203	10,31	1:20k
	15,0					102	3715	15141	11,82

Tab. 7.2: Laufzeittests der Gebäudeverschmelzung

reduzierten Datensatz eine Größe von 1,5 km x 1,5 km. Die Zeit zur Prozessierung des größeren Datensatzes umfasst etwa eine Minute. Der reduzierte Datensatz kann innerhalb von 6 Sekunden berechnet werden, stellt allerdings im Maßstab 1:75k lediglich einen Kartenausschnitt von 2 cm x 2 cm dar (vgl. Abbildung 7.4).



Abb. 7.4: Durch Typifizierung und Verdrängung innerhalb von 6 Sekunden erstellter Kartenausschnitt (2 cm x 2 cm)

7.3.4 Beurteilung der Testergebnisse

Die Ergebnisse der beschriebenen Laufzeitversuche der automatisierten Generalisierungsverfahren sollen dazu genutzt werden, zu entscheiden, ob diese in einem Kartenservice zur Aufbereitung der Daten für einen bestimmten Maßstab in Echtzeit geeignet sind oder ob es unerlässlich ist, die Daten vorher zu erzeugen und in einer MRDB zu speichern. Ein Echtzeit-System wird im engeren Sinne bezeichnet als ein System, welches auf eine Anfrage ohne feststellbare Verzögerung antwortet. Wird der Begriff weiter ausgelegt, so definiert dieser sich nach der Zeit, die ein Nutzer bereit ist, um auf eine Antwort vom System zu warten. Aus der gesammelten Erfahrung heraus wird ein Internetnutzer nach etwa zehn Sekunden ungeduldig und spricht nach etwa 30 Sekunden von einer „schlechten Systemleistung“ und bricht die Anfrage ab (Bouch et al. 2000). Dabei verbraucht die Übertragung der Daten über das Netzwerk allein schon einen großen Zeitvorrat, gerade bei mobilen Anwendungen, so dass wenig Spielraum für die Prozessierung der Daten verbleibt.

Die beschriebenen Verfahren erfüllen diese Anforderungen unter der Voraussetzung, dass die Datenmenge reduziert wird. Somit empfiehlt sich bei größeren Datenmengen auf jeden Fall die Vorverarbeitung der Daten. Zum anderen bedingen die genannten Verfahren streng genommen eine manuelle Nachbearbeitung, um qualitativ ansprechende Kartenbilder zu erzeugen, obgleich dieses wenigstens die Einhaltung der Minimaldimensionen sicherstellen können.

An dieser Stelle soll noch einmal die geringe Bandbreite gerade mobiler Internetanbindungen (vgl. Unterabschnitt 2.4.2) aufgegriffen werden. Durch die Generalisierung wird in der Regel auch die Datenmenge, respektive Punktmenge, reduziert, so dass ein Service, welcher generalisierte Datensätze bietet, immer im Vorteil gegenüber einem Dienst ist, welcher nur hochauflösende Daten bereitstellt. Um dies zu verdeutlichen und zu bestätigen, soll ein Vergleichstest dreier möglicher Szenarien durchgeführt werden. Als Testbett für diesen Test dient die in dem EU-Projekt *GiMoDig* entwickelte Systemarchitektur (Details hierzu finden sich in Lehto 2003). Das verwendete System greift auf einen *Datenservice (WFS)* zurück und überträgt diese Daten dann an den *Datenprozessierungs-Service (DPS)*, welcher bei Bedarf eine Generalisierung durchführt. Die Daten werden anschließend an den *Portal-Service (PSL)* weitergegeben, welcher eine Transformation der Daten in ein für den Client lesbares Format, beispielsweise SVG, vornimmt und die Daten an diesen weiterleitet.

<i>Inhalt</i>	<i>Objekte vorher</i>	<i>Straßensegmente</i>	<i>Reduktionsrate</i>	<i>Objekte nachher</i>	<i>Laufzeit (msek)</i>	<i>Laufzeit/Objekt (msek)</i>	<i>Zielmaßstab</i>	
Großer Datensatz (4x4 km), Gebäude reduziert, typifiziert und gegenseitig und gegenüber Straßennetz verdrängt			0,8	1595	69939	35,92		
			0,9	1794	66775	34,29	1:10k	
			1,0	1946	73055	37,52		
		1947	539	0,5	997	61638	31,65	1:25k
				0,7	1401	65413	33,60	
				0,6	966	58713	30,15	1:50k
Kleiner Datensatz (1,5x1,5 km), Gebäude reduziert, typifiziert und gegenseitig und gegenüber Straßennetz verdrängt			0,5	638	54487	27,60	1:75k	
			1,0	262	6200	23,66	1:10k	
		262	64	0,7	193	5658	21,60	1:25k
				0,6	72	5708	21,75	1:50k
				0,5	91	5477	20,90	1:75k

Tab. 7.3: Laufzeittests Typifizierung und Verdrängung

Innerhalb dieser Architektur soll nun der Weg der Daten von der Datenbank zum Client verfolgt werden, wobei folgende Szenarien unterschieden werden sollen

1. Der Service greift auf hochauflösende Daten zurück und leitet diese ohne Generalisierung weiter an den Client (Tabelle 7.4)
2. Der Service greift auf hochauflösende Daten zurück, generalisiert diese online (Gebäudevereinfachung) und sendet das Ergebnis an den Client (Tabelle 7.5)
3. Der Service greift auf die entsprechend aufgelösten Daten der MRDB zurück und leitet diese ohne Generalisierung weiter an den Client (Tabelle 7.6)

Die Tabellen 7.4 bis 7.6 fassen die Ergebnisse dieses Testes zusammen. In den Tabellen kann unterschieden werden zwischen der Prozessierungszeit innerhalb der entsprechenden Ebene sowie der Zeit zur Übertragung der Daten von einer Ebene an die nächste.

<i>Quellmaßstab</i>	<i>Online- Generalisierung</i>	<i>Punkte vorher/nachher</i>	<i>Objekte vorher/nachher</i>	<i>Zielmaßstab</i>
1:5k	ohne	15528/15527	1155/1154	1:5k
<i>System-Ebene</i>	<i>Prozessierung (msek)</i>	<i>Übertragung (msek)</i>	<i>Gesamtzeit (msek)</i>	<i>Datenvolumen (kB)</i>
Datenservice (WFS)	2348	5446	7794	
DPS (Online-Gen.)	20	223	243	228
PSL (XSLT)	1108	1171	2277	
Summe	5456	6840	10314	

Tab. 7.4: Abfrage hochauflösender Daten und Weiterleitung durch das System zum Client ohne Generalisierung

Das Ergebnis zeigt, dass der Gesamtprozess in Verbindung mit einer MRDB (Fall 3) nur 6,69 Sekunden in Anspruch nimmt, während die Übertragung und Generalisierung hochauflösender Daten (Fall 2) 9,96 Sekunden benötigt und eine Übertragung hochauflösender Daten ohne Generalisierung (Fall 1) sogar noch mehr, nämlich 10,32 Sekunden bedarf (siehe Abbildung 7.5). Zwei Gründe sind hier anzuführen:

Quellmaßstab	Online- Generalisierung	Punkte vorher/nachher	Objekte vorher/nachher	Zielmaßstab
1:5k-1:10k	Gebäudevereinf. (5,0m)	15528/5262	1155/914	1:20k
System-Ebene	Prozessierung (msek)	Übertragung (msek)	Gesamtzeit (msek)	Datenvolumen (kB)
Datenservice (WFS)	2579	5464	8043	118
DPS (Online-Gen.)	142	299	441	
PSL (XSLT)	602	875	1477	
Summe	3323	6638	9961	

Tab. 7.5: Abfrage hochauflösender Daten, anschließende Online-Generalisierung und Weiterleitung zum Client

Quellmaßstab	Online- Generalisierung	Punkte vorher/nachher	Objekte vorher/nachher	Zielmaßstab
1:20k	ohne	5265/5265	917/917	1:20k
System-Ebene	Prozessierung (msek)	Übertragung (msek)	Gesamtzeit (msek)	Datenvolumen (kB)
Datenservice (WFS)	1428	3906	5334	124
DPS (Online-Gen.)	24	0	24	
PSL (XSLT)	515	817	1332	
Summe	1967	4723	6690	

Tab. 7.6: Abfrage von Daten für den Maßstab 1:20k aus einer MRDB

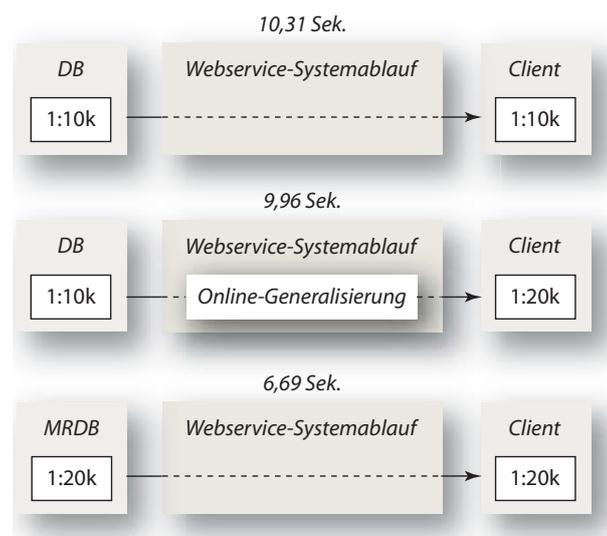


Abb. 7.5: Vergleich der Laufzeiten ohne Generalisierung (oben), mit Online-Generalisierung (Mitte) und mit MRDB-Daten (unten)

1. Durch das Auslassen von Prozessen zur Online-Generalisierung kann Zeit gespart werden
2. Werden zuvor generalisierte Daten zwischen den Diensten übermittelt, so wird die Übertragungsmenge und -zeit reduziert

Dabei nimmt die Online-Generalisierung (Gebäudevereinfachung) an sich nur 142 Millisekunden in Anspruch, andere Verfahren benötigen hier wesentlich mehr Zeit. Entscheidend ist, dass alle Prozesse, welche von der Datenmenge beeinflusst werden, wie die Abfrage des Daten aus der Datenbank über den WFS (5334 ms zur Abfrage generalisierter Daten aus der MRDB gegenüber 8043 ms zur Abfrage hochaufgelöster Daten), die Transformation der Daten in ein SVG-Bild in der Portalebene (PSL) und die anschließende Übertragung an den Nutzer, mehr Zeit benötigen, wenn eine größere Datenmenge vorliegt. Dies wird auch dadurch bestätigt, dass die Daten, welche auf halben Wege online generalisiert wurden, im Endeffekt schneller beim Nutzer ankommen als die hochaufgelösten Daten.

Somit ist es nicht nur aus kartografischer Sicht, sondern gleichzeitig auch aus Sicht des Systems sinnvoll, Daten zu generalisieren, bevor diese aus der Datenbank abgerufen werden. Der effektivste Weg, um Geodaten über einen Datenservice bereitzustellen, führt somit über eine MRDB, welche Daten für den gewünschten Zielmaßstab bereithält. Auf der einen Seite wird hierdurch die zu übertragende Datenmenge reduziert und auf der anderen Seite kann die Zeit aufwändiger Generalisierungsprozesse gespart werden. Nicht zuletzt kann durch eine Vorprozessierung, welche auch eine manuelle Nachbearbeitung erlaubt, die Qualität der Geodatenvisualisierung deutlich gesteigert werden.

8 MRDB-Webdienst: Client-Ebene

In der Client-Ebene treten nun die anwendungsspezifischen Funktionen auf, welche einen Nutzer bei der Bearbeitung seiner speziellen Aufgabe unterstützen sollen. Somit besitzen die in dieser Ebene auftretenden Funktionalitäten die höchste Individualität und die Zahl möglicher Facetten ist gleichzeitig am größten. Dabei kann der Anwender seine Informationen direkt über den MR-WFS abfragen, wie es die später beschriebene, desktopgebundene GIS-Anwendung demonstriert oder aber er greift indirekt über einen weiteren, spezialisierten Webdienst auf diese zu. Dieser Fall wird durch eine Applikation für einen mobilen Nutzer, einen *Thin Client*, in dem folgenden Abschnitt dargestellt. Dieser mobile Client greift auf die im vorangegangenen Kapitel beschriebenen Funktionalitäten eines Mapservices zurück, welcher der Applikation auf dem mobilen Geräte auf diese Weise Prozessierungsleistungen abnimmt und gleichzeitig als Schnittstelle zum Abruf der benötigten Informationen dient.

Beide Applikation, der mobile und der desktopgebundene Anwender, decken zusammen den größten Teilbereich potenzieller Nutzer eines MRDB-Service ab, sind auf der anderen Seite jedoch nur spezialisierte Stellvertreter ihres Anwendungsgebietes.

8.1 Eine Applikation für mobile Nutzer

Beide in den Nutzerszenarien dargestellte Anwender greifen im Idealfall mit ihrem mobilen Gerät über einen für ihre Erfordernisse spezialisierten Webdienst auf die Daten einer MRDB zu. Die Applikation des mobilen Gerätes fungiert dabei als grafisches Nutzerinterface, durch welches der Anwender seine Bedürfnisse formulieren kann und mit Hilfe dessen die Ergebnisse visualisiert werden. Die Darstellung und Interaktion mit einer SVG-basierten Kartendarstellung stellt die zentrale Anforderung an die mobile Applikation dar. Sowohl der Feuerwehrmann als auch die Studentin erhalten ihre benötigten Informationen primär über eine Kartendarstellung. Zusätzlich sollen Informationen zu bestimmten Objekten abgefragt und dargestellt werden.

Zur Realisierung einer solchen, auf mobile Geräte spezialisierte Anwendung stehen diverse Umgebungen zur Programmentwicklung bereit. Diese bauen auf gängigen Programmiersprachen, wie *Visual Basic* oder *C++* auf. Der Programmierer muss sich in der Regel jedoch für eine Zielplattform, beispielsweise *Windows Mobile*, *Palm OS*, *EPOC* oder *Symbian OS* (vgl. Unterabschnitt 2.4.1) entscheiden. Daneben gibt es plattformübergreifende Umgebungen, wie die *J2ME¹*, *Personal Java* für leistungsstärkere mobile Geräte (z.B. PDAs) oder das *Embedded Linux*.

Die gegenwärtige Webservice-Architektur (vgl. Abschnitt 2.2) bevorzugt von Natur aus ein browserbasiertes Thin-Client-Modell (Chu et al. 2004). Da die Darstellung von SVG-Objekten zukünftig auch von Browsern mobiler Geräte, wie derzeit dem *Opera mini*, nativ unterstützt wird, bietet es sich an, die gesamte Applikation auf der Basis von SVG aufzubauen. Diese kann dann auf jedem mobilen Gerät, welches einen SVG-fähigen Browser besitzt, verwendet werden. Alternativ existieren diverse mobile SVG-Viewer, welche die vollen SVG-Funktionalitäten unterstützen. Die Browser mobiler Geräte unterstützen dagegen nur die eingeschränkten Profile *SVG Tiny* für low-end Geräte (Mobilfunkgeräte, Smartphones etc.) sowie *SVG Basic* für mobile high-end Geräte (PDAs etc.) (W3C 2003).

Die SVG-Objekte können mit Hilfe der Programmiersprache *ECMA-Script* (einer standardisierten Variante von *JavaScript*) manipuliert werden, so dass durch eine derartige Erweiterung Interaktionen möglich sind. Hierdurch lassen sich komplette, auf SVG beruhende Applikationen erschaffen, welche klickbare Objekte, Buttons, Menüs und andere Interaktionsmöglichkeiten enthalten.

8.1.1 Funktionalitäten

Die Architektur, in welche der mobile Client integriert ist, wird durch die Abbildung 7.1 auf Seite 136 beschrieben. Die Applikation des mobilen Nutzers steht über das Internet mit dem Kartendienst in Verbindung, welcher diese

¹Java2 Micro Edition

mit Serviceinformationen auf XML-Basis sowie Geodaten in Form von SVG-Dateien versorgt. Das Nutzerinterface der Applikation ist in Abbildung 8.1 dargestellt. Das Hauptfenster der Anwendung ist derart aufgebaut, dass

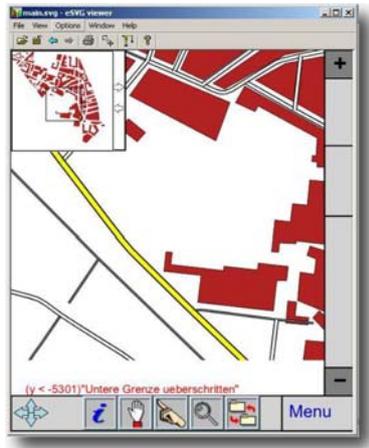


Abb. 8.1: Nutzerinterface der mobilen, SVG-basierten Applikation mit Kartenfenster (Mitte), Nebenkarte (oben links), Zoombalken (rechts) und Menüleiste (unten)

sich am unteren Rand des Bildschirms eine Leiste befindet, um zum einen die unterschiedlichen Zeigerfunktion auszuwählen und zum anderen die Untermenüs aufrufen zu können. Am rechten Bildrand befindet sich eine Zoomleiste zur schrittweisen Vergrößerung oder Verkleinerung der Kartendarstellung. Schließlich ist in der oberen linken Ecke ein kleines Zusatzfenster platziert, welches in dem Kartenmodus eine Übersichtskarte darstellt. Dieses Fenster kann bei Bedarf ein- oder ausgeblendet werden, um wertvollen Bildschirmplatz freizugeben. Der Hauptteil des Bildschirms ist für die Darstellung der Karte reserviert. Alle übrigen Informationen und Interfaces werden nur bei Bedarf eingeblendet.

Die Applikation bietet zunächst die Möglichkeit, einige Grundeinstellungen vorzunehmen, wie die Benennung der URL des Mapservices. Dieser Eintrag wird gespeichert und steht bei zukünftigen Programmaufrufen erneut zur Verfügung. Nach dem Programmstart baut das System eine Verbindung mit dem Kartendienst auf und ruft zunächst die Servicefunktion 'GetMetaData' auf. Der implementierte Mapservice-Prototyp stellt dem Client Information über die verfügbaren Objektarten sowie die vorliegenden Auflösungsstufen in Form eines XML-Dokumentes bereit. Somit kann der Nutzer oder die intelligente Applikation die benötigten Objektarten und deren Auflösungsstufen bei Bedarf auswählen und abfragen. Dem Anwender steht hierzu im Menü eine Auswahlliste zur Verfügung, welche auf Grundlage dieser Metadaten dynamisch erstellt wird (vgl. Abbildung 8.2, links). Da dem System die möglichen Auflösungsstufen inklusive ihrer empfohlenen Maßstabsbereiche nun bekannt sind, kann diese Information zu einem intelligenten Zooming genutzt werden. Entspricht der Maßstab nicht der aktuell dargestellten Auflösungsstufe, so wird ein alternativer LoD nachgeladen. Während der Initialisierungsphase wird der am rechten Bildschirmrand sichtbare Zoombalken den verfügbaren Auflösungsstufen angepasst. In diesem Fall stehen Daten vom Maßstab 1:1k bis 1:100k zur Verfügung. Somit entspricht der obere Rand des Balkens dem Maßstab 1:1k und der untere Rand 1:100k.

Der mobile Nutzer wird nun zunächst über das Menü eine GPS Anbindung aktivieren (vgl. Abbildung 8.2, rechts). Die Applikation liest über den entsprechenden Port im Hintergrund den NMEA²-Datenstrom aus, transformiert die geografischen Koordinaten in ebene Gauß-Krüger-Koordinaten und legt diese alle zehn Sekunden in einer temporären Datei ab. Die Berechnung der Koordinaten erfolgt dabei in mehreren Schritten. Zunächst werden die aus dem GPS stammenden geografischen WGS84-Koordinaten (ϕ, λ, h) in geozentrische (X,Y,Z) Koordinaten umgewandelt. Diese Koordinaten (WGS84:X,Y,Z) werden anschließend durch eine 7-Parameter-Transformation in das Datum des DHDN³ übertragen. In einem nächsten Schritt werden diese Koordinaten wieder in ellipsoidische Koordinaten, bezogen auf das Bessel-Ellipsoid, transformieren. Diese ellipsoidischen Koordinaten (B, L, H) lassen sich schließlich in die Ebene projizieren, so dass sich durch diese Projektion schließlich die gesuchten Gauß-Krüger-Koordinaten (R, H, h) ergeben. Auf diese Weise wird die aktuelle Position des Nutzers ermittelt und auf der Karte angezeigt.

Die Studentin des zweiten Szenarios wird also an dem Hauptbahnhof in Hannover ihren PDA einschalten und die Applikation aufrufen. Nach einer Initialisierungsphase, in der das Gerät Kontakt zu dem Kartendienst aufnimmt und die verfügbaren Auflösungsstufen und Objektarten abfragt, wird die aktuelle Position bestimmt. Anschließend werden zunächst automatisch Siedlungsinformationen geringer Auflösung von dem Mapservice

²National Marine Electronics Association

³Deutsches Hauptdreiecksnetz

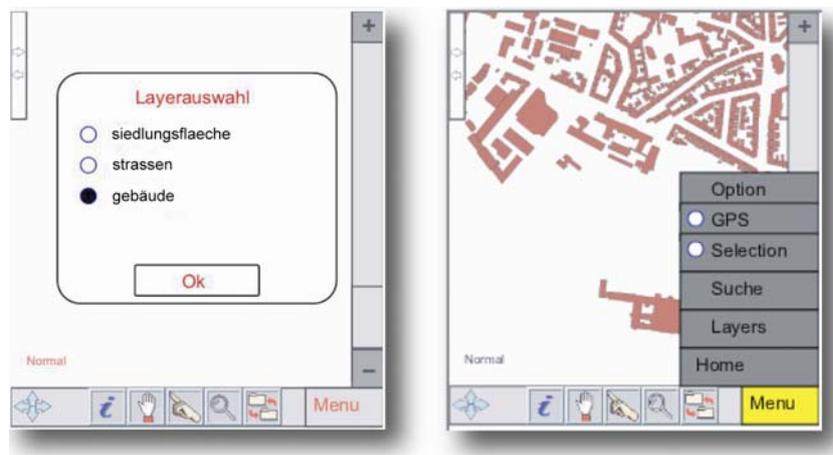


Abb. 8.2: Bestimmung der gewünschten Kartenebenen (links) und Menüauswahl der Applikation (rechts)

abgefragt und in einer kleinen Karte in dem Zusatzfenster angezeigt. Gleichzeitig wird standardmäßig eine Karte höchster Auflösung von dem Mapservice über die Funktion 'Getmap' angefragt und in dem mittleren empfohlenen Maßstab (hier 1:3k) angezeigt. Der Kartenausschnitt wird über den Bounding-Box-Filter so begrenzt, dass dieser bei dem kleinsten empfohlenen Maßstab (hier 1:5k) um eine halbe Kartenbreite über den Bildschirm hinausragt, um den Ausschnitt verschieben zu können, ohne sofort neue Daten nachladen zu müssen. Die Karte wird auf die aktuelle Position des Nutzers zentriert. Zusätzlich ist in der Übersichtskarte der aktuelle Kartenausschnitt durch ein Rechteck markiert, um dem Nutzer die Orientierung und Navigation zu erleichtern (vgl. Abbildung 8.1).

Die Studentin hat jetzt, je nach der aktuellen Situation, diverse Möglichkeiten. Sie kann mit Hilfe des Zoombalkens am rechten Bildrand den Kartenausschnitt schrittweise vergrößern oder verkleinern. Das System kennt den aktuellen Maßstab sowie den empfohlenen Maßstabsbereich der gegenwärtig verwendeten Geodaten. Wird dieser Bereich verlassen, so fragt das System über den Map-Service automatisch nach einer entsprechenden, alternativen Repräsentation aus der MRDB (`GetMap&scale=x&layers=a,b`). Darüber hinaus kann direkt in der Karte ein Ausschnitt definiert werden, welcher dann entsprechend vergrößert dargestellt wird. Bei Bedarf wird auch hier eine höher aufgelöste Repräsentation nachgeladen. Daneben kann der Kartenausschnitt verschoben werden. Fehlende Daten werden bei Bedarf neu abgefragt.

Christa möchte nun zunächst zur Jugendherberge gelangen. Die Applikation fordert Christa zur Eingabe der Adresse der Jugendherberge auf. Die Möglichkeit der Adresssuche ist in dem Prototypen nicht realisiert. Die Informationen zu den Gebäuden sowie deren Adressen sind jedoch in der MRDB gespeichert, so dass die Abfrage der Position eines einzelnen Gebäudes über den OGC-WFS problemlos umgesetzt werden kann, indem nur das Gebäude mit der spezifischen Hausnummer und Straße abgefragt wird. Somit lässt sich die Position des Gebäudes feststellen und in der Karte darstellen. Das hochauflösende Straßennetzwerk kann zur Berechnung einer Fußgängeroute genutzt werden.

Auf dem Weg zur Jugendherberge wird nun der Navigationsmodus der Anwendung aktiviert. Dieser versucht in einer Karte sowohl den aktuellen Standpunkt als auch den Zielort darzustellen. Hierzu werden die Funktionen `MultiscaleMap` und `Lupe` des Kartendienstes verwendet (siehe Unterabschnitt 7.2.3). Es wird eine multiskalige Karte angefordert, wobei die aktuelle Position dem GPS-Signal entnommen wird (Abbildung 8.3). Hat der Nutzer sich um mehr als zehn Meter bewegt, wird eine neue Karte mit entsprechend aktualisiertem Standpunkt nachgeladen. Die Applikation braucht daher lediglich die Position des Nutzers zu überwachen und bei Bedarf die Funktion `MultiscaleMap` oder `Lupe` aufrufen, in der die aktuelle Position übergeben werden. Desweiteren benötigt der Mapservice zur Ausführung dieser Funktion zwei Maßstäbe. Der kleine Maßstab ergibt sich aus dem aktuellen Abstand zum Ziel. Es soll sowohl der Stand- als auch der Zielort in der Karte darstellbar sein. Bei einem Abstand zum Zielort von 1 km ergibt sich somit ein Kartenausschnitt von 2 km x 2 km, da der aktuelle Standort immer in der Mitte der Karte zu finden ist. Zur Darstellung der höher aufgelösten Informationen im Zentrum der Karte soll die höchste verfügbare Auflösung verwendet werden. Gleichzeitig wird der Kartenausschnitt durch Angabe einer Bounding-Box begrenzt. Auf diese Weise werden die Parameter `scale1`, `scale2`, `layers`, `BBOX` zum Aufruf der Funktion bestimmt. Der Nutzer kann diese automatischen Parameter bei Bedarf manuell verändern.

Neben der Möglichkeit, in der Karte zu navigieren, kann der Nutzer Informationen einzelner Objekte durch Anklicken abfragen. So lassen sich beispielsweise die Straßennamen, welche auf Grund der kleinen Darstellungsfläche nicht im Kartenbild enthalten sind, explizit anzeigen (siehe Abbildung 8.4). Diese Funktionalitäten

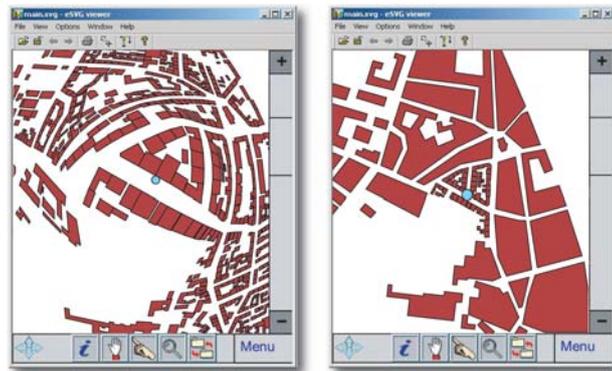


Abb. 8.3: Navigationsmodus mit lupenartiger Vergrößerung (links) und höherer Auflösung (rechts) im definierten Umkreis um die aktuelle Position

werden durch Auswahl des Informationsmodus aktiviert. Hier bietet das SVG-Vektor-Format den Vorteil, jedes Kartenobjekt ansprechen und Attributinformationen abfragen zu können.

In dem Selektionsmodus bietet die Applikation zusätzlich die Möglichkeit, einzelne Objekte auszuwählen. Das System speichert die IDs der ausgewählten Objekte. Über das Menü besteht nun die Möglichkeit, zu den gewählten Objekten alternative Geometrien oder Attribute abzufragen. Christa möchte zum Beispiel zu einem bestimmten Gebäude alle verfügbaren Informationen erhalten. Sie aktiviert den Auswahlmodus und klickt auf das sie interessierende Gebäude. Anschließend kann sie wählen, ob sie geometrische oder beschreibende Informationen zu dem gewählten Objekt erhalten möchte. Sie wählt die letztgenannte Option. Die Applikation hat zuvor die ID des Gebäudes gespeichert und ruft nun die Funktion 'InformationDrilling' zusammen mit dem Parameter 'Objekt=GebäudeLoD1.10' auf. Der Mapservice ruft seinerseits nun Informationen über den MR-WFS ab. Die Prozedur der Informationsabfrage wurde ausführlich in dem Unterabschnitt 7.2.3 dargelegt. In diesem Fall werden die horizontal verknüpften Objekte aus der MRDB abgefragt und deren Attribute in einem XML-Dokument abgelegt. Die mobile Applikation erhält als Abfrageergebnis in diesem Fall ein XML-Dokument mit den Informationen 'ID=SiedlungsflächeLoD1.112', 'GN=Nordstadt', 'EWZ=12056', 'ID=OrtslageLoD1.9', 'GN=Hannover'. Die Anwendung bereitet die Darstellung in einer Liste auf. Auch an dieser Stelle ist das System sicherlich noch zu ergänzen. Zusätzliche Informationen aus anderen Quellen können integriert werden. So bietet der Gutachterausschuss regelmäßig Mietspiegel, bezogen auf eine bestimmte geografische Einheit, zum Beispiel Stadtteile. Die Integration und Nutzung alternativer Informationen erfolgt jedoch nach den hier dargestellten Prinzipien.

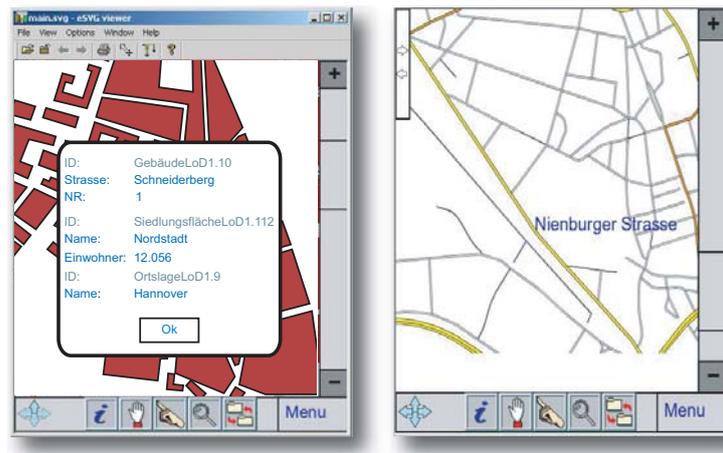


Abb. 8.4: Abfrage zusätzlicher Attribute verknüpfter Objekte durch 'Information Drilling' (links), Abfrage der zu einem Objekt gespeicherten Attribute (rechts)

Wählt der Anwender die Option 'Geometrie' an Stelle von 'Attribute', so kann er aus den verfügbaren Auflösungsstufen die gewünschte Darstellung selektieren. Den Feuerwehrmann interessiert zum Beispiel die detaillierte Darstellung des generalisierten Gebäudes, um den Löscheinsatz planen zu können. Auch er befindet sich in einem Übersichtsmaßstab, zum Beispiel 1:45k, und wählt im Selektionsmodus ein oder mehrere Objekte aus. Entspre-

chend der Auswahl werden alternative Repräsentationen mit Hilfe der Service-Funktion `GetMap` sowie zusätzlich den Parametern `feature=GebäudeLoD2.13`, `scale=1000` und `linkedwith=45000` angefragt. Der Mapservice ruft seinerseits, wie in dem vorangegangenen Kapitel beschrieben wurde, die Funktion `<GetAlternativeFeature>` des MR-WFS auf. Der Ergebnisdatensatz beinhaltet Gebäudegeometrien in der Auflösung für den Maßstab 1:1k, welche mit dem ausgewählten Objekt GebäudeLoD2.13 verknüpft sind. Die resultierenden SVG-Daten werden in dem kleinen Fenster, in dem vorher die Übersichtskarte dargestellt wurde, angezeigt. Zusätzlich enthalten diese SVG-Objekte ein Attribut, welches angibt, mit welchem Objekt sie in dem Maßstab 1:45k verknüpft sind. Auf diese Weise erhält die Applikation Informationen darüber, welche alternativen Repräsentationen nun welchem Kartenobjekt zuzuordnen sind. Durch Klick auf eines der gewählten Objekte in der Hauptkarte wird die alternative Repräsentation im Nebenfenster dargestellt. Somit ist es möglich, nicht nur alle alternativen Repräsentationen gleichzeitig sondern einzeln je nach Bedarf darzustellen (siehe Abbildung 8.5).

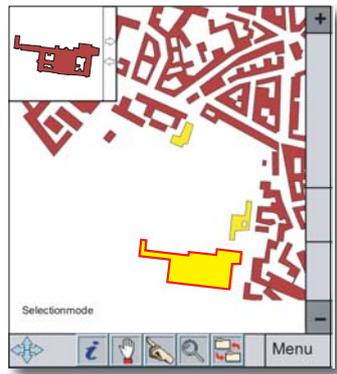


Abb. 8.5: Abfrage und Darstellung alternativer Repräsentationen

Der bis hierher beschriebene Prototyp birgt sicherlich noch umfangreiches Erweiterungspotenzial. Sowohl die Client- als auch die Applikationsebene können beispielsweise zusätzlich eine Form der Personalisierung bereitstellen. Auf diese Weise wählt das System automatisch, je nach Anwender und vor allen Dingen Verwendungszweck eine entsprechende Darstellung der geometrischen sowie semantischen Details. Der eilige Geschäftsmann möchte lediglich die Information über den kürzesten Weg zum Bahnhof erhalten, während der Tourist Zeit hat, um alle verfügbaren Informationen auf dem Weg dorthin aufzunehmen. Auch die Analysemöglichkeiten zur Abfrage von Informationen aus den alternativen Repräsentationen können, wie bereits angedeutet, in der Applikationsebene noch intelligenter umgesetzt werden. Der vorgestellte Prototyp soll hier lediglich eine Idee möglicher Funktionalitäten geben.

8.2 Desktopgebundener GIS-Client

Neben der Verwendung multiskaliger Daten in mobilen Applikationen bietet die MRDB auch für GIS-Anwender nützliche Informationen und Funktionen. Die Funktionalitäten können in diesem Szenario auf die Seite des Clients, also in die GIS-Anwendung integriert werden, da dieser über ausreichende Rechenleistung verfügt, um die Daten zu prozessieren und gleichzeitig die Anforderungen so speziell werden können, dass diese von dem Nutzer selbst vorgegeben werden. Die Anwendung verzichtet daher an dieser Stelle auf den Rückgriff auf einen Mapservice und greift direkt auf den MR-WFS zu.

Als Testplattform dient hier das im Unterabschnitt 5.3.1 vorgestellte Java-basierte GIS *JUMP*, welches es ermöglicht, eigene Menüeinträge und Zeigerfunktionen zu integrieren.

Die neu integrierten Funktionalitäten entsprechen in etwa denen des Mapservices. So besteht die Möglichkeit, über eine *Drillingfunktion* durch Anklicken eines Objektes die verlinkten Repräsentationen der nächst höheren Auflösungsstufe aufzurufen und das gewählte Objekt zu ersetzen. So können beispielsweise für Planungszwecke an interessanten Stellen in der Karte durch einfaches Klicken mehr Details aufgerufen werden (siehe Abbildung 8.6 oben). Die Abbildung zeigt, dass an mehreren Stellen die Darstellung der Siedlungsflächen (rot) nach Auswahl dieser durch generalisierte Gebäudedarstellungen (orange) ersetzt wurden. Durch erneutes Auswählen dieser generalisierten Gebäudedarstellung wird die nächste verfügbare Auflösungsstufe (violett) aufgerufen. Neben der Darstellung zusätzlicher Details ist auch eine Funktion zur Bewegung in die entgegengesetzte Richtung denkbar, also zu weniger Details. Diese Möglichkeit besteht, da die Verlinkungen in der Datenbank bidirektional sind und eine Abfrage in beide Richtungen erlauben.



Abb. 8.6: *Information Drilling: Abfrage und Darstellung verlinkter Objekte höherer Auflösung an spezifischen Stellen (oben). Intelligentes Zooming: Vergrößern mit gleichzeitiger Darstellung eines höheren LoD (unten)*

Des Weiteren ist eine „intelligente Lupe“ integriert, welche der herkömmlichen Zoomingfunktion entspricht. Zusätzlich werden jedoch bei Vergrößerungen oder Verkleinerungen entsprechend hochauflösende oder geringer aufgelöste Daten zur Darstellung verwendet. Somit kann der Anwender durch die Daten navigieren und an Stellen, die ihn besonders interessieren, durch eine Vergrößerung auch gleichzeitig mehr geometrische Details erkennen (siehe Abbildung 8.6, unten).

Schließlich besteht die Möglichkeit, sich die Verknüpfungsstruktur einer Objektart in einer Art Verzeichnisbaum anzeigen zu lassen (siehe Abbildung 8.7). In der höchsten Ebene des Baumes befinden sich die am niedrigsten aufgelösten Instanzen der Objektart. Die einzelnen Objekte werden jeweils durch ihre ID repräsentiert. Durch Anklicken einer ID öffnet sich an dieser Stelle die nächste Ebene des Baumes und zeigt die IDs der verknüpften Objekte. Wurden die entsprechenden Daten in das GIS geladen, so werden diese Daten im Baum farblich markiert. Durch Auswahl eines Objektes im Baum wird das korrespondierende Objekt im Datensatz hervorgehoben. Auf diese Weise lassen sich zum einen die Zuordnungen zwischen korrespondierenden Objekten überprüfen, zum anderen ergeben sich hier neue Möglichkeiten der Datenanalyse. Die Verknüpfung einer Gruppe von Objekten mit einem übergeordnetem Objekt lässt Zusammenhänge zwischen dieser Gruppe von Objekten erkennen. Vor allen Dingen auch horizontale Verknüpfungen zwischen Objekten derselben Auflösungsstufe, welche in einem gewissen Zusammenhang stehen, können somit erkannt und angezeigt werden. Die horizontalen Verknüpfungen können dabei auf vielfältige Weise entstehen und sind nicht auf die geometrische Nähe und Zugehörigkeit zu demselben Objektbereich, wie in dieser Arbeit, beschränkt.

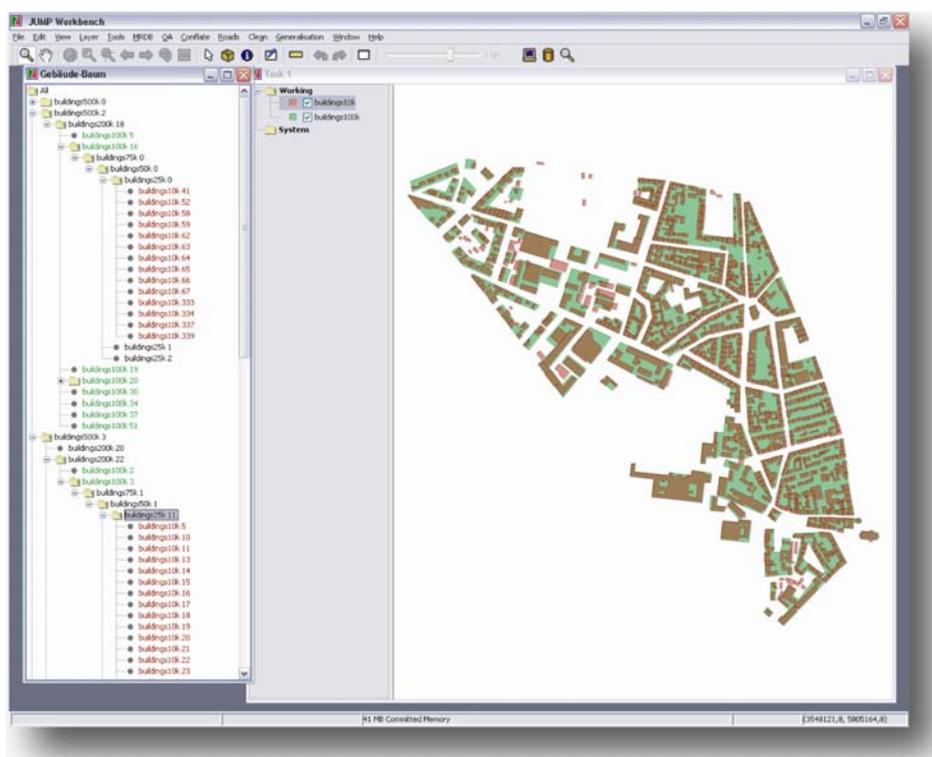


Abb. 8.7: JUMP-GIS MRDB Funktionalität: Darstellung der Verknüpfungen der MRDB in Form eines Baumes (links) und die korrespondierenden Daten (rechts)

9 Ergebnisse und Ausblick

9.1 Ergebnisse der Arbeit

Das Ziel der vorliegenden Arbeit besteht in der Integration einer multiskaligen Datenbank in eine Webservice-Umgebung, um auf diese Weise die Vorteile beider Technologien zu vereinen und Probleme sowie neue Nutzungsmöglichkeiten, die hieraus erwachsen, aufzudecken und zu evaluieren.

In dem Arbeitsfeld multirepräsentativer Datenbanken haben vorangegangene Arbeiten die Vorteile der Integration multipler Geodaten in ein konsistentes DBMS aufgezeigt. Erst durch die explizite Modellierung der Übergänge zwischen den verschiedenen Repräsentationen entsteht ein widerspruchsfreies System. Alle verfügbaren Informationen können auf diese Weise fehlerfrei vorgehalten und verwendet werden.

In dem Kapitel 2 werden die Vorteile aufgeführt, die durch das Internet und den hierauf aufbauenden Webdiensten entstehen. Durch die offene Architektur entsteht unter anderem das *spatially enabled web*, welches es jedem Nutzer ermöglicht, auf die verschiedensten Geodatenquellen zuzugreifen, ohne dass dieser auf spezifische, proprietäre Formate der Anbieter Rücksicht nehmen muss. Auch innerhalb einer derartigen Webumgebung gilt es, zur fehlerfreien Nutzung der diversen verfügbaren Informationsquellen, multirepräsentative Daten bewusst in eine hierfür ausgelegte Umgebung einer MRDB einzubetten, ähnlich den bisher evaluierten Problemen interner Prozesse (siehe Abschnitt 3.1). Eine Anforderungsanalyse eines mobilen Geoinformationsservices dekuviert den Bedarf einer integrierten, multirepräsentativen Datenbasis mit der gleichzeitigen Möglichkeit, von überall auf diesen Informationspool zuzugreifen zu können.

In der vorliegenden Arbeit wird, basierend auf den Erkenntnissen vorangegangener Arbeiten, eine Systemarchitektur erarbeitet, welche den Zugriff auf eine multiskalige Datenbasis über eine offene Schnittstelle ermöglicht. Somit werden die Erkenntnisse zur sinnvollen Verwendung multirepräsentativer Datenbestände in einem integrierten System an Stelle von mehreren nebeneinander existierenden, monorepräsentativen Daten auch innerhalb einer Webumgebung umgesetzt. Es entstehen neue Konzepte zur Verwendung multirepräsentativer Informationen innerhalb eines Webdienstes sowie neuartige Nutzungsmöglichkeiten in der Geodatenvisualisierung und der Informationsanalyse.

Die einzelnen Komponenten der vorgeschlagenen Architektur werden explizit modelliert mit dem Ziel der mobilen Verwendung multiskaliger Daten. Es wird das Datenmodell einer multiskaligen Datenbank vorgeschlagen, welches bestehende und auch neu zu generierende Datensätze unterschiedlicher geometrischer Auflösung integriert. Dabei wird Wert darauf gelegt, die ursprünglichen Informationen möglichst unverändert in das System zu übernehmen, um ein allgemeingültiges, auf andere Datensätze übertragbares System zu erhalten.

Es werden Methoden zur Integration bestehender und auch neu zu generierender Datensätze beschrieben. Die entwickelten, halbautomatischen Generalisierungsverfahren demonstrieren eine Möglichkeit zum Aufbau unterschiedlicher Auflösungsstufen mit gleichzeitiger Integration dieser generalisierten Daten in ein konsistentes, multiskaliges System.

Des Weiteren werden in der vorliegenden Arbeit die Anforderungen an eine Web-Schnittstelle zum Zugriff auf multiskalige Datensätze erarbeitet. Es bleibt festzuhalten, dass es an dieser Stelle nicht genügt, eine auf monorepräsentative Daten ausgelegte Schnittstelle zu verwenden, um die Vorteile einer multirepräsentativen Datenbasis auszuschöpfen. Es gilt, zusätzliche Funktionalitäten bereitzustellen, welche den Zugriff auf eine spezifische Repräsentation zulassen. Daher wird in dieser Arbeit der OGC-WFS um entsprechende Operationen erweitert.

Die Arbeit betont den Nutzwert einer MRDB nicht nur aus Sicht der Datenhaltung, sondern hebt vor allen Dingen auch den erweiterten Informationsgehalt für den Anwender heraus. Es erwachsen neue Möglichkeiten, welche durch die Implementierung eines Kartendienstes exemplarisch umgesetzt werden. Der Nutzer hat über die Verknüpfungen Zugriff auf alle verfügbaren Informationen und Sichtweisen. Multiskalige Datensätze eröffnen neue Formen der Geodatenvisualisierung gerade für mobile Nutzer. Die Arbeit zeigt Wege zur Reduktion der überflüssigen Informationsmenge auf, indem entsprechende Repräsentationen der MRDB aufgerufen werden. Hierdurch wird das überlastete Kartenbild sowie die limitierte Bandbreite der Internetanbindung mobiler Geräte entlastet.

Das gesamte System wird in Form eines Prototypen implementiert. Auf diese Weise wird die Realisierbarkeit demonstriert und es werden die Probleme alternativer Wege und Möglichkeiten im Vergleich zu der MRDB-basierten Lösung dargelegt.

Folgende Ergebnisse sind somit festzuhalten:

- Entwicklung eines Konzeptes zum Aufbau einer MRDB (Kapitel 5)
 - Entwicklung eines multiskaligen Datenbankmodells
 - Entwicklung eines multiskaligen Datenbankschemas
 - Exemplarische Datengenerierung und gleichzeitige -integration durch Generalisierungsverfahren
 - Exemplarische Datenintegration durch Matching
- Entwicklung einer MRDB Service-Schnittstelle (Kapitel 6)
 - Anforderungsanalyse an eine Service-Schnittstelle
 - Erweiterung der MRDB um relevante Metadaten
 - Erweiterung des OGC-WFS um MRDB-spezifische Funktionalitäten und Besonderheiten
- Beschreibung neuer Möglichkeiten zur Nutzung einer MRDB innerhalb einer Webumgebung (Kapitel 7 und Kapitel 8)
 - Neue Formen der Informationsabfrage: Der Nutzer erhält Zugriffsmöglichkeiten auf alle mit einem Phänomen verknüpfte Informationen
 - Neue Formen der Visualisierung: Unterschiedliche Auflösungsstufen können zur Darstellung von Karten unterschiedlicher Maßstäbe, aber auch innerhalb derselben Visualisierung, verwendet werden
 - Möglichkeit der Datenreduktion durch Verwendung unterschiedlicher Repräsentationen der MRDB
- Prototypische Entwicklung der Komponenten
 - Aufdeckung der Probleme bestehender Lösungen
 - Darstellung der Anwendbarkeit und Praktikabilität der vorgestellten Lösung
 - Vergleich mit alternativen Lösungsmöglichkeiten

In der vorliegenden Arbeit wird somit versucht, die Vorteile einer MRDB nicht nur aus Sicht der Datenhaltung zu sehen. Vielmehr sollten die multirepräsentativen Informationen einer möglichst breiten Nutzerschicht bereitgestellt werden. Dies wird erstmalig durch die Nutzung der Webservice-Technologie realisiert.

Die wesentlichen Entwicklungsschritte der vorliegenden Arbeit sind daher in den Kapiteln 5 bis 8 zu finden. Das Datenmodell und -schema werden im Hinblick auf die Nutzung aus einer Webumgebung heraus an diese neuen Anforderungen angepasst. Der Zugriff auf die integrierten, multirepräsentativen Informationen erfolgt über eine überarbeitete Webservice-Schnittstelle, welche im Vergleich zu existierenden Datenservice-Schnittstellen die Abfrage aller zu einem bestimmten Phänomen verfügbaren Informationen zulässt. Der entwickelte Prototyp stellt neue Nutzungsmöglichkeiten vor, welche erst durch die Verfügbarkeit einer multiskaligen Datenbasis innerhalb einer Webservice-Umgebung realisierbar sind. Dabei wird das Nutzungspotenzial durch die vorgestellten Applikationen jedoch keinesfalls ausgeschöpft. Multiple Repräsentationen entstehen durch die Kombination verschiedenster Sichtweisen und jede Kombination eröffnet neue Nutzungsmöglichkeiten; unterschiedliche Erfassungszeitpunkte ermöglichen zum Beispiel die Erforschung historischer Entwicklungen.

Die vorliegende Arbeit zeigt somit, dass es mit entsprechenden Modifikationen bestehender Systeme möglich ist, eine MRDB als Datenbasis innerhalb einer Webumgebung einzusetzen und dass darüber hinaus auf diese Weise neue Möglichkeiten für den Nutzer entstehen. Die Arbeit legt ein Muster eines MRDB-basierten Webdienstes vor, welches auf andere Datensätze und Problemstellungen übertragbar ist und es dem interessierten Leser ermöglicht, ein für seine Zwecke angepasstes System aufzubauen.

9.2 Diskussion und Ausblick

Die Modellierung und Implementierung einer MRDB wird in der vorliegenden Arbeit durch einen multiskaligen Datensatz realisiert. Multiskalige Datensätze bilden unter anderem die Basis kartografischer Dienste, um Visualisierungen unterschiedlicher Maßstäbe zu generieren, und dienen darüber hinaus vielen weiteren Anwendungen mit räumlichem Kontext als Grundlage.

Multiplizität bedeutet jedoch mehr als nur eine Variation der geometrischen Auflösung der Daten. Allgemein resultieren multiple Datensätze aus den verschiedenen Ansprüchen an die Daten oder, anders formuliert, aus unterschiedlichen Sichten auf die beschriebenen Phänomene. Multiskalige Daten repräsentieren eine Form der Multiplizität, welche wegen der individuellen Stellung geografischer Informationen eine besondere Bedeutung besitzt (Balley et al. 2004). Das beschriebene Modell ist jedoch nicht auf multiskalige Daten beschränkt, sondern kann auf andere multirepräsentative Datensätze übertragen werden.

Die bestehenden und durch Generalisierung abgeleiteten, in sich konsistenten Datensätze wurden unverändert in das DBMS übernommen und Verknüpfungen zwischen korrespondierenden Objekten modelliert. Durch die Verknüpfung monorepräsentativer Datensätze, welche unverändert in das System übernommen werden können, durch Beschreibung der Verknüpfungen in separaten Tabellen sowie die Zusammenfassung aller relevanten Informationen in einer Metadatentabelle ist es problemlos möglich, die MRDB zu erweitern oder eine eigene Datenbank nach dem beschriebenen Muster aufzubauen. Es ist lediglich sicherzustellen, dass die zu integrierenden, monorepräsentativen Datensätze in sich und auch die verschiedenen Repräsentationen untereinander konsistent sind. Die Konsistenz zwischen den einzelnen Repräsentationen kann geprüft werden, nachdem korrespondierende Objekte durch entsprechende Matchingverfahren bestimmt wurden. Dabei bleibt die Konsistenz innerhalb eines multiskaligen Systems auch nach Veränderungen einzelner Datensätze erhalten. Änderungen einer Auflösungsstufe ziehen demzufolge auch Anpassungen an den verknüpften Objekten nach sich. Derartige Folgemaßnahmen auf Grund einer Veränderung wurden in dem hier vorgestellten Prototypen nicht explizit modelliert, sondern lediglich durch die Verknüpfungen angedeutet. Eine MRDB, welche über den prototypischen Status hinausgeht, sollte konkrete Konsistenzregeln und darauf beruhende Prozeduren zur Aufrechterhaltung der Konsistenz enthalten.

Die Modellierung multirepräsentativer Informationen ist auf verschiedene Weisen möglich. Die hier demonstrierte Form zur Verknüpfung monorepräsentativer Datensätze durch entsprechende Links stellt nur eine Möglichkeit dar. Alternativ können die Daten in einen multirepräsentativen Objekttypen integriert oder durch eine hierarchische Struktur modelliert werden, in der die Unterklassen jeweils eine Repräsentation der Oberklasse darstellen (vgl. Unterabschnitt 3.4.2). Der Datenanbieter muss entscheiden, welches Datenmodell er favorisiert. In dem gezeigten Fall ist es einfacher, die einzelnen Repräsentationen getrennt voneinander vorzuhalten, als alle Informationen in einem Objekttypen zu vereinen. Diese Form der Datenintegration wäre beispielsweise sinnvoll, wenn die Abfragen sich in den überwiegenden Fällen auf alle Repräsentationen gleichzeitig beziehen.

Auch die Verknüpfungsstrategie ist variabel und von der intendierten Anwendung und auch dem gewählten Datenmodell abhängig, wie in dem Unterabschnitt 5.2.3 gezeigt wird. Probleme treten in der vorgestellten Modellierung auf, wenn entlang der Verknüpfungen höher aufgelöste Objekte abgefragt werden sollen, da hochauflösende Objekte existieren, welche keinen Bezug zu den geringer aufgelösten Repräsentationen besitzen. Sollen Informationen auf diesem Wege abgefragt werden, so sind die vorgeschlagenen, alternativen Verknüpfungsstrategien zu verfolgen.

Die entwickelte MRDB-Schnittstelle (Kapitel 6) zum Zugriff auf die MRDB bleibt nach außen hin unverändert, unabhängig von dem Datenmodell der angeschlossenen MRDB. Die Kommunikation zwischen der MRDB und dem Service dagegen liegt in der Verantwortlichkeit des Daten- und Serviceanbieters. Dieser muss den Service an die Datenbank anpassen. Der hier beschriebene und implementierte MR-WFS greift zur Abfrage der jeweiligen Repräsentation auf die Metadaten-, die Verknüpfungs- und schließlich auf die Objektdatentabelle zurück. In einer hierarchischen Modellierung bietet das übergeordnete Objekt den Einstieg, sind die Informationen in einem multirepräsentativen Objekttypen vereint, so sind die relevanten Informationen aus diesem zu extrahieren. Es ist schwierig, einen Service so allgemein zu gestalten, dass dieser unabhängig von der gewählten Modellierung einsetzbar ist. Der Service muss für einen solchen Fall in bestimmten Teilen redundant entwickelt werden, um verschiedene Modelle bedienen zu können.

Gleichzeitig müssen gewisse Restriktionen für die MRDB aufgestellt werden, um zumindest innerhalb derselben Modellierungsstrategie einen Konsens zu erreichen, so dass nicht für jede MRDB der Datenservice neu angepasst werden muss. Werden multirepräsentative Datensätze zukünftig von den gängigen DBMS unterstützt, ähnlich wie es sich derzeit mit der Unterstützung von Geometriedaten verhält, so lässt sich eine MR-Serviceschnittstelle auf die verschiedenen DBMS anpassen, vergleichbar mit den einzelnen Implementierungen des OGC-WFS. Diese unterstützen zum Beispiel die Datenbankmodelle von *Oracle Spatial*, der *PostgreSQL/PostGIS* Lösung oder die *ESRI-Shape-Files* als Datenbasis.

Die vorgestellten Verfahren zur Bestimmung der Verknüpfungen erfüllen lediglich exemplarischen Charakter. Der Bereich des Datenmatchings umfasst weit mehr Aspekte als die hier vorgestellte Möglichkeit, durch Ähnlichkeit in der Lage eine Zuordnung vorzunehmen. Auch an dieser Stelle besteht Erweiterungspotenzial. Weitere Parameter zur Bestimmung der Ähnlichkeit bestehender Datensätze, welche auch semantische Informationen einbeziehen, sollten berücksichtigt werden. Das Verfahren zur Bestimmung korrespondierender Objekte ist dabei stark

abhängig von den verwendeten Datensätzen. Multitemporale Daten gehen zum Beispiel von einer Veränderung des Objektes aus. Somit sind hier wieder andere Parameter anzusetzen.

Alternativ ist es möglich, die Bestimmung der Verknüpfungen schon innerhalb des Prozesses zur Ableitung alternativer Repräsentationen vorzunehmen. Diese Vorgehensweise wurde beispielhaft anhand von Generalisierungsprozessen demonstriert. Es hat sich gezeigt, dass auf diese Weise eine problemlose und vor allen Dingen fehlerfreie Bestimmung der Zuordnung möglich ist, denn die Verlinkungen beinhalten ebendiese Aussage, dass an dieser Stelle ein Übergang zu einer veränderten Auflösung durch Generalisierung stattfindet. Im Idealfall wird nicht nur die Verknüpfung selbst, sondern zusätzlich der Übergang von einer Repräsentation zur nächsten explizit modelliert; also beispielsweise die Verschmelzung mehrerer Gebäude zu einer neuen Geometrie unter Verwendung bestimmter Algorithmen und Parameter.

Die im Rahmen dieser Arbeit implementierten Generalisierungsverfahren dienen vornehmlich dem Aufbau einer von groß- bis kleinmaßstäbigen Daten reichenden Datenbasis. Diese genügen nicht den Ansprüchen eines ausgereiften, automatisierten Verfahrens zur Generalisierung. Zum einen bedingen die vorgestellten Verfahren manuelle Nachbearbeitungen, zum anderen wurde hier größtenteils ohne Rücksicht auf benachbarte Objekte generalisiert. Produktionsverfahren zur Ableitung unterschiedlich aufgelöster Geodaten sind viel komplexer und ein vollautomatischer Arbeitsfluss zur Generierung aller gewünschten Maßstäbe aus einem hochauflösenden Datensatz ist auf dieser Basis nicht ohne Weiteres möglich.

Gleichzeitig sollen die entwickelten Verfahren einen Vergleich ermöglichen zwischen der zumindest theoretischen Möglichkeit, aus einem hochauflösenden Datensatz auf Anfrage automatisch den gewünschten Maßstab und die entsprechende Auflösung abzuleiten und auf der anderen Seite der Abfrage von Daten aus der MRDB. Die Ergebnisse unterstreichen, dass selbst bei Verfügbarkeit automatischer Generalisierungsverfahren eine Speicherung vorbereiteter, geometrisch und semantisch reduzierter Daten in einer MRDB sinnvoll ist. Neben der Möglichkeit zur Generierung hochqualitativer Visualisierungen für den Nutzer überzeugt vor allen Dingen die Tatsache, dass sich durch Verwendung einer MRDB gleichzeitig die Prozessierungs- und Übertragungszeit, selbst bei wenig zeitintensiven Generalisierungsverfahren, reduziert.

Der Zugriff auf verteilte Geodaten über das Internet erfolgt idealerweise über eine definierte Schnittstelle und transportiert die Daten über das standardisierte HTTP, so dass jeder Nutzer mit seinem Webbrowser auf die Informationen zugreifen kann, ohne sich Gedanken über Formate und Austauschprotokolle machen zu müssen. Dieser Weg wurde vom OGC aufgezeigt und hat zu offenen, integrierten Systemen geführt. Versuche, die multiskaligen Daten über bestehende Schnittstellen abzufragen, haben jedoch aufgedeckt, dass diese keine ausreichenden Funktionalitäten für multirepräsentative Daten bereitstellen. Die beschriebenen Erweiterungen der OGC-WFS Schnittstelle zeigen eine Richtung auf, in welche Veränderungen vorgenommen werden müssen, um auch derartige Datensätze uneingeschränkt nutzen zu können. Dabei bleibt die Schnittstellendefinition unabhängig von der Form der Multiplizität. Diese gestattet nicht nur die Abfrage von Daten unterschiedlicher Auflösungen, sondern erlaubt ebenso eine Analyse von Informationen unterschiedlicher Epochen oder verschiedener Sichtweisen. Die Implementierung der MR-WFS-Schnittstelle ist nicht als allumfassende Lösung zu sehen. Sie zeigt vielmehr die Schwachstellen bestehender Schnittstellen auf und schlägt eine Richtung zur Verbesserung der Situation vor, indem neue Operationen und Parameter zur Verwendung multirepräsentativer Daten eingeführt werden.

Um eine vollständige Anpassung an die Voraussetzungen multirepräsentativer Daten vorzunehmen, müssen alle in dem jeweiligen Service enthaltenen Funktionen zur Abfrage oder Bearbeitung monorepräsentativer Geodaten angepasst werden. Beispielsweise enthält der OGC-WMS Funktionen zur Abfrage kartografischer Visualisierungen von Geodaten. Auch hier besteht Anpassungsbedarf und auf der anderen Seite neues Potenzial zur standardisierten Abfrage beispielsweise von multiskaligen Karten oder Überlagerungen unterschiedlicher Sichtweisen zu Analyse Zwecken. Das grundsätzliche Problem besteht darin, ähnlich der Datenmodellierung, einen Übergang zu formulieren von einer monorepräsentativen zu einer multirepräsentativen Datenbasis.

Das Potenzial, welches dem Nutzer durch die Bereitstellung multirepräsentativer Systeme infolge offener Schnittstellen erwächst, wird in der vorliegenden Arbeit angedeutet. Neue Formen der Visualisierung von Geodaten entstehen. Der Nutzer kann mit den Informationen versorgt werden, welche er aktuell benötigt, indem die entsprechende Repräsentation gewählt wird. Die Bestimmung relevanter, kontextabhängiger Informationen bleibt zukünftigen Systemen vorbehalten. Derartige Systeme bedingen jedoch die hier beschriebene Möglichkeit, auf eine multirepräsentative Datenbasis zurückgreifen zu können. Gleichzeitig besteht nun die Gelegenheit, alternative Sichtweisen oder Auflösungsstufen parallel zu verwenden, zu vergleichen und zu analysieren um hieraus weitere, implizite Informationen abzuleiten.

Daneben sind die hinlänglich bekannten Vorteile einer MRDB für den Datenhalter nicht zu vernachlässigen. Die Wartung des Systems wird vereinfacht und dessen Effektivität erhöht. Dieser Nutzen bleibt auch innerhalb der aufgezeigten Verwendung erhalten und kann mit den neuen, vorgestellten Möglichkeiten ergänzt werden.

Literaturverzeichnis

- 3GPP (2002), Push Service, Technical Specification 3G TS 22.174 0.3.1, 3rd Generation Partnership Project (3GPP).
- Abel, D. J., Ooi, B. C., Tan, K.-L. & Tan, S. H. (1998), 'Towards integrated Geographical Information Processing', *International Journal of Geographical Information Science* **12**(4), 353–371.
- Anderl, R. (2006), 'Skript zur Lehrveranstaltung Produktdatentechnologie B', <http://www.dik.tu-darmstadt.de/dmdocuments/2006kript.pdf>. DiK, Technische Universität Darmstadt.
- Anders, K.-H. & Bobrich, J. (2004), MRDB approach for automatic incremental update, *in*: 'ICA Workshop on Generalisation and Multiple Representation', Leicester.
- Andersen, K. V. B., Cheng, M. & Klitgaard-Nielsen, R. (2003), Online Aalborg Guide - Development of a Location-Based Service, Technischer Report, Aalborg Universität.
- Anderson, P. (2005), Mobile and PDA Technologies: Looking Around the Corner, White paper, JISC Technology and Standards Watch.
- Annoni, A., Bernard, L., Fullerton, K., Kanellopoulos, I., Millot, M., Peedell, S., Smits, P., De Groof, H., Vanderhaegen, F. & Rase, D. (2004), Towards a European Spatial Data Infrastructure: The INSPIRE Initiative, *in*: 'Proceedings of the 7th Conference on Global Spatial Data Infrastructure (GSDI 7), ISRO, DST', Bangalore (IN).
- Atkinson, R. (2001), Gazetteer Service Specification, OGC-IP draft candidate implementation specification, discussion paper, OGC.
- Badard, T. (1999), On the automatic retrieval of updates in geographic databases based on geographic data matching tools, *in*: 'Proceedings of the 19th International Cartographic Conference', ICA/ACI, Ottawa, CA.
- Badard, T. & Lemarié, C. (2000), Propagating updates between geographic databases with different scales, *in*: D. Martin & P. Atkinson, Hrsg., 'Innovations in GIS VII: GeoComputation', Taylor & Francis, Kapitel 10, Seiten 83–94.
- Balley, S., Parent, C. & Spaccapietra, S. (2004), 'Modeling geographic data with multiple representations', *International Journal of Geographical Information Science* **18**(4), 1–26.
- Barnes, S. J. (2003), 'Location-Based Services: The State of the Art', *e-Service Journal*, Indiana University Press, Seiten 59–70.
- Bauer, M. (2003), *Vermessung und Ortung mit Satelliten. GPS und andere satellitengestützte Navigationssysteme*, Wichmann Verlag, Heidelberg.
- Beaulieu, M. (2002), *Wireless Internet Applications and Architecture*, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.
- Becker, B., Six, H.-W. & Widmayer, P. (1991), Spatial priority search: An access technique for scaleless maps, *in*: 'Proceedings of the ACM SIGMOD', Band 20, Seiten 128–137.
- Bédard, Y. & Bernier, E. (2002), 'Supporting Multiple Representations with Spatial View Management and the Concept of VUEL'. Joint Workshop on Multi-Scale Representations of Spatial Data, ISPRS WG IV/3, ICA Commission on Map Generalisation, 7-8 July, Ottawa.
- Beifuss, K., Cranovsky, R., Hauke, E., Boltzmann, L. & Ollenschläger, G. (2001), Entwicklung einer Methodik für die Ausarbeitung von Leitlinien für optimale medizinische Praxis, Technischer Report, Empfehlung Rec (2001) 13 des Europarates und erläuterndes Memorandum.
- Bernier, E., Bédard, Y. & Hubert, F. (2005), 'UMapIt: An on-demand web mapping tool based on a multiple representation database'. Workshop paper, 8th ICA Workshop on Generalisation and Multiple Representation, A Coruña, July 7-8th.
- Bertin, J. (1983), *Graphische Semiologie*, Gruyter-Verlag, Berlin, New York.
- Bill, R. (2006), *Grundlagen der Geo-Informationssysteme, Bd.1, Hardware, Software und Daten*, Wichmann-Verlag.
- Blasby, D., Davis, M., Kim, D. & Ramsey, P. (2003), 'GIS conflation using open source tools', white paper.
- Bollman, J. & Koch, W. (2002), *Lexikon der Kartographie und Geomatik*, Spektrum Akademischer Verlag Heidelberg, Berlin.
- Booth, D., Haas, H., McCabe, F., Newcomer, I., Champion, M., Ferris, C. & Orchard, D. (2004), Web Services Architecture, W3C Working Draft, Technischer Report, W3C.
- Bouch, A., Kuchinsky, A. & Bhatti, N. (2000), Quality is in the eye of the beholder: Meeting users' requirements for Internet quality of service, Technischer Report 297-304, HP Laboratories Palo Alto.
- Brown, J., Rao, A. & Baran, J. (1995), Are you conflated? Integrating TIGER and other data sets through automated network conflation, *in*: 'GIS-T Symposium Proceedings, American Association of State Highway and Transportation Officials', Seiten 220–229.
- Bruegger, B. & Kuhn, W. (1991), Multiple Topological Representations, Technischer Report 91-17, NCGIA, Department of Surveying Engineering, University of Maine.
- Bruegger, B. P. & Frank, A. U. (1989), Hierarchies over topological data structures, *in*: 'Proceedings of ASPRS/ACSM – Annual Convention', Band 4, Baltimore, Seiten 137–145.
- Buehler, K. (2003), 'Data Models and Interoperability', OGC white paper.

- Burghardt, D., Neun, M. & Weibel, R. (2005), 'Generalization services on the web—classification and an initial prototype implementation', *Cartography and Geographic Information Science* **32**(4), 257–268.
- Burnet, R. (2000), MurMur Project, Workpackage 3: Services, Deliverable 7: Service analysis, Technischer Report, Cemagref.
- Buttenfield, B. P. & Delotto, J. S. (1989), Multiple representations: Scientific report for the specialist meeting, Technischer Report 89-3, National Center for Geographic Information and Analysis (NCGIA).
- Canalys (2005), Changing times in the smart mobile device market, Technischer Report, Canalys. Company press release, 29th September 2005.
- Caspary, W. (1993), 'Qualitätsaspekte bei Geoinformationssystemen', *Zeitschrift für Vermessungswesen* **8/9**, 444–450.
- Cecconi, A. (2003), Integration of cartographic generalization and multi-scale databases for enhanced web mapping, Doktorarbeit, Universität Zürich.
- Chu, H.-H., You, C.-W. & Teng, C.-M. (2004), Challenges: Wireless Web Services, *in*: 'Proceedings of the 10th International Conference on Parallel and Distributed Systems (ICPADS 2004)', Newport Beach, CA, USA, Seiten 657–664.
- Codd, E. F. (1982), 'Relational database: A practical foundation for productivity', *Communications of the ACM* **25**(2), 109–117.
- Conrad, S. (1997), *Föderierte Datenbanksysteme : Konzepte der Datenintegration.*, Springer-Verlag, Berlin.
- Cowen, D. J., Jenson, J. R., Bresnahan, P. J., Ehler, G. B., Graves, D., Huang, X., Wiesner, C. & Halkard E. Mackey, J. (1995), 'The design and implementation of an integrated geographic information system for environmental applications', *Photogrammetric Engineering and Remote Sensing*. **61**(11), 1393–1404.
- Cox, S., Daisey, P., Lake, R., Portele, C. & Whiteside, A. (2004), OpenGIS Geography Markup Language (GML) Implementation Specification 3.1.0, Technischer Report OGC 03-105r1, Open GIS Consortium, Inc.
- Cromley, R. G. (1991), 'Hierarchical methods of line simplification', *Cartography and Geographic Information Systems* **18**(2), 125–131.
- Crompvoets, J., Bregt, A., Rajabifard, A. & Williamson, I. (2004), 'Assessing the worldwide developments of national spatial data clearinghouses', *International Journal of Geographical Information Science* **18**(7), 665–689.
- Davis, C. & Alves, L. L. (2005), Local spatial data infrastructures based on a service-oriented architecture, *in*: 'GEOINFO – Brazilian Symposium on GeoInformatics', Seiten 71–86.
- Devogele, T. (2002), A new merging process for data integration based on the discrete frèchet distance, *in*: 'Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications', Ottawa.
- Devogele, T., Parent, C. & Spaccapietra, S. (1998), 'On spatial database intergration', *International Journal of Geographical Information Systems* **12**(3), 335–352.
- Devogele, T., Trevisan, J. & Raynal, L. (1996), Building a multi-scale database with scale-transition relationships, *in*: 'Proceedings of the 7th International Symposium on Spatial Data Handling (SDH), Advances in GIS research II', Seiten 6.19–6.33.
- Dimpfel, F. & Heep, T. (2004), Deliverable 4.1: Report on infrastructure, applications and services, Technischer Report ELBA-03-4.1, European Location Based Advertising (ELBA).
- Dobson, M. & McAvoy, J. M. (1989), Position statement, *in*: B. P. Buttenfield & J. S. Delotto, Hrsg., 'Multiple representations: Scientific report for the specialist meeting', National Center for Geographic Information and Analysis (NCGIA), Seiten 42–43.
- Douglas, D. & Peucker, T. (1973), 'Algorithms for the reduction of the number of points required to represent a digitized line or its caricature', *The Canadian Cartographer* **10**(2), 112–122.
- Doyle, A. & Reed, C. (2001), Introduction to OGC Web Services, Technischer Report, OGC.
- Doytsher, Y. (2000), 'A rubber sheeting algorithm for non-rectangular maps', *Computers & GeoSciences* **26**(9), 1001–1010.
- Dunkars, M. (2004a), Automated generalisation in a multiple representation database, *in*: S. A. Brandt, Hrsg., 'Proceedings of the 12th International Conference on Geoinformatics Geospatial Information Research: Bridging the Pacific and Atlantic', University of Gävle, Sweden.
- Dunkars, M. (2004b), Multiple representation databases for topographic information, Doktorarbeit, Royal Institute of Technology (KTH), Stockholm, Sweden.
- Edwardes, A., Burghardt, D. & Weibel, R. (2003), Webpark – location based services for species search in recreational areas, *in*: 'Proceedings of the 21st International Cartographic Conference', Durban, South Africa.
- Egenhofer, M., Clementini, E. & Felice, P. D. (1994), Evaluating inconsistencies among multiple representations, *in*: 'Proceedings of the 6th International Symposium on Spatial Data Handling', Edinburgh, Scotland, UK, Seiten 901–920.
- Elias, B., Hampe, M. & Sester, M. (2005), *Map-based mobile services - Theories, Methods and Implementations*, Springer Verlag, Kapitel Adaptive visualisation of landmarks using an MRDB, Seiten 73–86.
- Elmagarmid, A., Rusinkiewicz, M. & Sheth, A. (1999), *Management of Heterogeneous and Autonomous Database Systems*, Morgan Kaufmann, San Francisco, CA.
- Fairbairn, D. & Taylor, G. (1995), 'Developing a variable scale map projection for urban areas', *Computers & GeoSciences* **21**(9), 1053–1064.
- Farley, P. & Capp, M. (2005), 'Mobile Web Services', *BT Technology Journal* **23**(3), 202–213.

- Feringa, W. (2001), File formats and plugins, *in*: M. Kraak & A. Brown, Hrsg., 'Webcartography: Developments and prospects', Taylor & Francis, Seiten 177–193.
- Ferraiolo, J. & Jackson, D. (2003), Scalable Vector Graphics (SVG) 1.1 Specification, Technischer Report, W3C.
- Filin, S. & Doytsher, Y. (2000), 'The detection of corresponding objects in a linear-based map conflation', *Surveying and Land Information Systems* **60** (2), 117–128.
- Fitzke, J. & Greve, K. (2002), 'Umweltkarten im Internet: Das Potenzial von WebMap-Services', *Kartographische Nachrichten* **52**, 247–255.
- Fornefeld, M. & Oefinger, P. (2005), Verrechnungsmodelle für Geo-Webdienste, Technischer Report, MICUS Management Consulting GmbH.
- Gabay, Y. & Doytsher, Y. (1994), Adjustment of line maps, *in*: 'Proceedings of GIS/LIS 1994', Phoenix, USA, Seiten 191–199.
- Goodchild, M. (1991), Issue of quality and uncertainty, *in*: J. C. Müller, Hrsg., 'Advances in Cartography', Pergamon, Seiten 113–139.
- Goodwins, R. (2005), 'Otellini outlines new chip architecture', <http://news.zdnet.com/>. zdnet news.
- Gösseln, G. v. & Sester, M. (2003), Semantic and geometric integration of geoscientific data sets with ATKIS - applied to geo-objects from geology and soil science, *in*: 'ISPRS Commission IV Joint Workshop: Challenges in Geospatial Analysis, Integration and Visualization II', Stuttgart, Seiten 111–116.
- Gotthard, W., Lockemann, P. C. & Neufeld, A. (1992), 'System-guided view integration for object-oriented databases', *IEEE Transactions on Knowledge and Data Engineering* **4**(1), 1–22.
- Guttman, A. (1984), R-trees: A dynamic indexing structure for spatial searching, *in*: 'Proc. ACM SIGMOD International Conference on Management of Data', Seiten 47–57.
- Hage, C., Jensen, C., Pedersen, T., Speicys, L. & Timko, I. (2003), Integrated data management for mobile services in the real world, *in*: 'Proceesings of VLDB', Seiten 1019–1030.
- Hake, G., Grünreich, D. & Meng, L. (2002), *Kartographie*, Walter de Gruyter, Berlin, New York.
- Hampe, M., Harrie, L. & Sester, M. (2004), Multiple representation databases to support visualization on mobile devices, *in*: 'Proceedings of the XXth ISPRS Congress, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, XXXV(B4:IV)', Istanbul, Turkey.
- Hampe, M. & Intas, S. (2006), Extension of the ogc web feature service standard for multiple representation data, *in*: W. Kainz & A. Pucher, Hrsg., 'Proceedings of the ISPRS Technical Commission II Symposium 2006', Band XXXVI aus Reihe *ISPRS Archives*, Vienna, Austria, Seiten 49–54.
- Hampe, M., Hose, K., Hatger, C. & Katterfeld, C. (2005), A web based and time dependent visualization of an event calendar, *in*: 'Proceedings of 22nd International Cartographic Conference', La Coruña, Spain.
- Harbeck, R. (2000), '15 Jahre ATKIS, und die Entwicklung geht weiter', *Vermessung Brandenburg* **1**, 3–14.
- Harrie, L. & Hellstrom, A.-K. (1999), A prototype system for propagating updates between cartographic data sets, *in*: 'Proceedings of ICC', Ottawa, Canada, Seiten 1397–1401.
- Harrie, L., Sarjakoski, L. T. & Lehto, L. (2002), A variable-scale map for small-display cartography, *in*: 'Proceedings of the Joint International Symposium on Geospatial Theory, Processing and Applications', Ottawa, Canada.
- Haunert, J.-H. & Sester, M. (2005), Propagating updates between linked datasets of different scales, *in*: 'Proceedings of the 22nd International Cartographic Conference', La Coruña, Spain.
- Hausdorff, F. (1919), 'Dimension äusseres in Mass', *Mathematische Annalen* **79**, 157–179.
- Heuer, A. & Saake, G. (2000), *Datenbanken: Konzepte und Sprachen*, MITP-Verlag, Bonn.
- Jones, C. B., Abdelmoty, A., Lonergan, M. E., van der Poorten, P. & Zhou, S. (2000), Multi-scale spatial database design for online generalisation, *in*: 'Proceedings of the 9th International Symposium on Spatial Data Handling', Beijing, Seiten 7b.34 – 7b.44.
- Jones, C., Kidner, D. B., Luo, Q., Bundy, G. L. & Ware, J. M. (1996), 'Database design for a multi-scale spatial information system', *International Journal Geographical Information Systems* **10**(8), 901–920.
- Kampshoff, S. (2005), Integration heterogener raumbezogener Objekte aus fragmentierten Geodatenbeständen, Doktorarbeit, Geodätisches Institut der Rheinisch-Westfälischen Technischen Hochschule Aachen.
- Kang, H.-K., Moon, J.-W. & Li, K.-J. (2004), Data update across multi-scale databases, *in*: S. A. Brandt, Hrsg., 'Proceedings of the 12th International Conference on Geoinformatics Geospatial Information Research: Bridging the Pacific and Atlantic', Seiten 749–756.
- Kavouras, M. & Kokla, M. (2002), Developing multi-scale, multi-context databases through the semantic integration of heterogeneous datasets, *in*: 'Proceedings of the 8th EC-GI&GIS Workshop, ESDI - A Work in Progress', Dublin, Ireland.
- Kidner, D. B. & Jones, C. B. (1994), A deductive object-oriented GIS for handling multiple representations, *in*: T. C. Waugh & R. G. Healey, Hrsg., 'Proceedings of the 6th international Symposium on Spatial Data Handling', Edinburgh, Scotland, UK, Seiten 882–900.
- Kilpeläinen, T. (1995), 'Updating multiple representation geodata bases by incremental generalization', *Geo-Information-Systeme* **8**(4), 13–18.
- Kilpeläinen, T. (1997), Multiple representation and generalization of geo-databases for topographic maps, Doktorarbeit, Finish Geodetic Institute, Masala, Finland.

- Kilpeläinen, T. (1998), Maintenance of topographic data by multiple representations, *in*: 'Proceedings of the annual conference and exposition of GIS/LIS', Forth Worth, TX, Seiten 342–351.
- Kim, W. (1995), *Modern database systems: The object model, interoperability, and beyond*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA.
- Lackner, W. (2004), Verteilte Informationsstrukturen zur modernen Wissensauffindung im Internet, Diplomarbeit, Technische Universität Graz, Institut für Informationssysteme und Computer Medien (IICM).
- Larson, J. A., Navathe, S. B. & Elmasri, R. (1989), 'A Theory of Attributed Equivalence in Databases with Application to Schema Integration', *IEEE Transactions on Software Engineering* **15**(4), 449–463.
- Lehto, L. (2003), Architecture specification. GiMoDig-project, IST-2000-30090, Deliverable D4.4.1: Final system architecture, Public EC report, Finnish Geodetic Institute.
- Lehto, L. & Kilpeläinen, T. (2001), Generalizing XML-encoded spatial data on the web, *in*: 'Proceedings of the 20th ICA/ACI Conference', Seiten 2390–2396.
- Lichtner, W. (1979), 'Kartennetztransformationen bei der Herstellung thematischer Karten', *Nachrichten aus dem Karten- und Vermessungswesen* **1**(79), 109–119.
- Lieberman, J. (2003), OpenGIS Web Services Architecture, Discussion Paper OGC 03-025, OGC.
- Maguire, D. J. & Longley, P. A. (2005), 'The emergence of geoportals and their role in spatial data infrastructures', *Computers, Environmet and Urban Systems* **29**(1), 3–14.
- Mantel, D. (2002), Konzeption eines Förderierungsdienstes für geographische Datenbanken, Diplomarbeit, Universität Hannover.
- Mantel, D. & Lipeck, U. (2004), 'Datenbankgestütztes Matching von Kartenobjekten', *Mitteilungen des Bundesamtes für Kartographie und Geodäsie* **31**, 145–153.
- Mark, D. (1989), 'Conceptual Basis for Geographic Line Generalization.', *AutoCarto, Baltimore, MD* **9**, 68–77.
- McMaster, R. B. (1991), Conceptual frameworks for geographical knowledge., *in*: B. Buttenfield & R. McMaster, Hrsg., 'Map generalization: Making rules for knowledge representation', Wiley, New York, NY, USA, Seiten 21–39.
- Middleton, C. (2002), 'Who needs a killer app? Two perspectives on content in residential broadband networks', *Journal of Research and Practice in Information Technology* **34**(2), 67–81.
- Müller-Veerse, F. (1999), Mobile Commerce Report, Technischer Report, Durlacher Research Ltd.
- Mohan, C. (2002), Dynamic E-business: Trends in Web Services, *in*: 'Technologies for E-Services :Third International Workshop, TES 2002, Proceedings', Springer, Berlin / Heidelberg, Seiten 1–5.
- Müller, J., Lagrange, J., Weibel, R. & Salge, F. (1995), Generalization: State of the art and issues, *in*: J. Müller, J. Lagrange & R. Weibel, Hrsg., 'GIS and Generalization', Taylor & Francis, London, UK, Seiten 3–17.
- National Reserach Council (1993), Towards a Coordinated Spatial Data Infrastructure for the Nation, Technischer Report, Mapping Sciences Committee. National Research Council. National Academy Press, Washington D.C.
- Nebert, D. & Whiteside, A. (2005), OGC Catalogue Services Specification, OGC Implementation Specification OGC 04-021r3, OGC.
- Nivala, A.-M. & Sarjakoski, L. T. (2003), Need for context-aware topographic maps in mobile devices, *in*: K. Virrantaus & H. Tveite, Hrsg., 'ScanGIS 2003, Proceedings of the 9th Scandinavian Research Conference on Geographical Information Science', Espoo, Finland, Seiten 15–29.
- Nivala, A.-M. & Sarjakoski, L. T. (2004), Preventing interruptions in mobile map reading process by personalisation, *in*: 'Proceedings of the 3rd workshop on HCI in mobile guides, in adjunction to: Mobile HCI, 6th international conference on human computer interaction with mobile devices and services', Glasgow, Scotland.
- OGC (1999), Simple Features Specification for SQL, OGC Implementation Specification Revision 1.1, Open Geospatial Consortium, Inc.
- OGC (2003a), OpenGIS Reference Model, Technischer Report OGC 03-040, Open Geospatial Consortium, Inc.
- OGC (2003b), Web Coverage Service (WCS), Version 1.0.0, Technischer Report OGC 03-065r6, Open Geospatial Consortium, Inc.
- OGC (2004), The spatial web, Technischer Report, Open Geospatial Consortium, Inc.
- OGC (2005a), Filter Encoding (Filter), OGC Implementation Specification OGC 04-095, Open Geospatial Consortium, Inc.
- OGC (2005b), OpenGIS Location Services (OpenLS): Core Services, Implementation Specification OGC 05-016, Open Geospatial Consortium, Inc.
- OGC (2005c), Web Feature Service Implementation Secification, OGC Implementation Specification 04-94, Open Geospatial Consortium, Inc.
- OGC (2006), Web Map Service (WMS), OGC Implementation Specification OGC 06-042, Open Geospatial Consortium, Inc.
- Paiva, J. (1998), Topological equivalence and similarity in multi-representation geographic databases, Doktorarbeit, University of Maine.
- Parent, C. (2000), MurMur Project, Workpackage 2: Data Modeling Approach, Deliverable 4: State of the Art review, Technischer Report, L'Université de Lausanne.
- Parent, C. & Spaccapietra, S. (1998), 'Database integration: An overview of issues and approaches', *Communications of the ACM* **41** (5), 166–178.

- Parent, C. & Spaccapietra, S. (2000), Database integration: The key to data interoperability, *in*: M. P. Papazoglou, S. Spaccapietra & Z. Tari, Hrsg., 'Advances in Object-Oriented Data Modeling', The MIT Press.
- Parent, C., Spaccapietra, S. & Zimanyi, E. (2000), MurMur: Database management of multiple representations, *in*: 'Proceedings of the AAAI-2000 Workshop on Spatial and Temporal Granularity', Austin, TX.
- Peng, Z.-R. & Tsou, M.-H. (2003), *Internet GIS: Distributed geographic information services for the internet and wireless networks*, John Wiley & Sons, Inc.
- Pfoser, D., Tryfona, N. & Verykios, V. (2003), Services-based data management in a global computing environment, *in*: 'Proceedings of Fourth International Conference on Web Information Systems Engineering Workshops, 2003', Seiten 45–53.
- Rajabifard, A. & Williamson, I. (2001), Spatial data infrastructures: Concept, SDI hierarchy and future directions, *in*: 'Proceedings of GEOMATICS'80 Conference', Tehran, Iran.
- Reichenbacher, T. (2003), Mobile cartography - Adaptive visualisation of geographic information on mobile devices, Doktorarbeit, Institut für Photogrammetrie und Kartographie, Technische Universität München.
- Reindl, W. (2005), 'Geodateninfrastruktur in Deutschland', *Mitteilungen des DVW-Bayern e. V.* **4**, 1–8.
- Rigaux, P. & Scholl, M. (1994), Multiple Representation Modelling and Querying, *in*: 'Proceedings of IGIS', Seiten 59–69.
- Rosson, M. B. & Carroll, J. M. (2002), *The human-computer interaction handbook: Fundamentals, evolving technologies and emerging applications*, Lawrence Erlbaum Associates, Inc., Mahwah, NJ, USA, Kapitel Scenario-based design, Seiten 1032–1050.
- Ruas, A. (1995), Multiple paradigms for automating map generalization: Geometry, topology, hierarchical partitioning and local triangulation, *in*: 'Proceedings of the ACSM/ASPRS, AutoCarto 12', Charlotte, NC, Seiten 69–78.
- Ruzak Mazur, E. & Castner, H. W. (1990), 'Horton's ordering scheme and the generalisation of river networks', *The Cartographic Journal* **27**, 104–112.
- Saalfeld, A. (1988), 'Conflation: Automated map compilation', *International Journal of Geographic Information systems* **2(3)**, 217–228.
- Samal, A., Seth, S. C. & Cueto, K. (2004), 'A feature-based approach to conflation of geospatial sources', *International Journal of Geographical Information Science* **18(5)**, 459–489.
- Sarjakoski, T., Sarjakoski, L. T., Lehto, L., Sester, M., Illert, A., Nissen, F., Rystedt, R. & Ruotsalainen, R. (2002), Geospatial info-mobility services - A challenge for interoperability and national mapping agencies, *in*: 'Proceedings of the 5th AGILE Conference on Geographic Information Science', Palma (Mallorca), Spain, Seiten 585–589.
- Schlender, B. (2000), 'Sony plays to win', *Fortune* **141(9)**, 142ff.
- Schut, P. & Whiteside, A. (2005), OpenGIS Web Processing Service, Discussion Paper OGC 05-007r4, Open Geospatial Consortium, Inc.
- Sester, M. (2000), 'Maßstabsabhängige Darstellung in digitalen räumlichen Datenbeständen', Habilitation, Deutsche Geodätische Kommission, Reihe C, Nr. 544.
- Sester, M. (2001), Optimization approaches for generalisation, *in*: D. B. Kidner & G. G. Higgs, Hrsg., 'GIS Research in the UK, Proceedings of the GIS Research UK, the 9th Annual Conference GISRUK', University of Glamorgan.
- Sester, M. (2002), Application dependent generalisation - The case of pedestrian navigation, *in*: 'Proceedings of the Joint International Symposium on GeoSpatial Theory, Processing and Applications', Ottawa, Canada.
- Sester, M., Anders, K.-H. & Walter, V. (1998), 'Linking objects of different spatial data sets by integration and aggregation', *GeoInformatica* **2, 4**, 335–358.
- Sester, M. & Brenner, C. (2004), Continuous generalization for visualization on small mobile devices, *in*: P. Fisher, Hrsg., 'Developments in Spatial Data Handling - 11th International Symposium on Spatial Data Handling', Springer Verlag, Seiten 469–480.
- Sester, M., Sarjakoski, L., Harrie, L., Hampe, M., Koivula, T., Sarjakoski, T., Lehto, L., Elias, B., Nivala, A.-M. & Stigmar, H. (2004), Real-time generalisation and multiple representation in the GiMoDig mobile service, Public EC report Deliverables D7.1.1, D7.2.1 and D7.3.1, GiMoDig Project.
- Shadbolt, N., Berners-Lee, T. & Hall, W. (2006), 'The semantic web revisited', *IEEE Intelligent Systems* **21(3)**, 96–101.
- Shekhar, S. & Chawla, S. (2003), *Spatial Databases. A Tour*, Prentice Hall, New Jersey.
- Skogan, D. (2005), Multi-resolution geographic data and consistency, Doktorarbeit, University of Oslo, Faculty of Mathematics and Natural Sciences.
- Smits, P., Düren, U., Østensen, O., Murre, L., Gould, M., Sandgren, U., Marinelli, M., Murray, K., Pross, E., Wirthmann, A., Salgé, F. & Konecny, M. (2002), INSPIRE Architecture and standards position paper, Technischer Report, INSPIRE Architecture and Standards Working Group, Brussels.
- Sonnet, J. (2004), OWS 2 Common Architecture: WSDL SOAP UDDI, OGC Discussion Paper OGC 04-060r1, Open Geospatial Consortium Inc.
- Spaccapietra, S. (2000), MurMur Project, Workpackage 2: Data Modeling Approach, Deliverable 5: Data Model specification, Technischer Report, Ecole Polytechnique Fédérale de Lausanne.
- Spaccapietra, S., Parent, C. & Dupont, Y. (1992), 'Model independent assertions for integration of heterogeneous schemas', *VLDB Journal: Very large databases* **1(1)**, 81–126.

- Spaccapietra, S., Parent, C. & Vangenot, C. (2000), GIS databases: From multiscale to multiresolution, *in*: 'Proceedings 4th International Symposium, SARA-2000', Texas, USA.
- Spaccapietra, S., Vangenot, C., Parent, C. & Zimanyi, E. (1999), MurMur: A research agenda on multiple representations, *in*: 'Proceedings of the International Symposium on Database Applications in Non-Traditional Environments', Kyoto, Japan.
- Spéry, L. (1998), A framework for update process in GIS, *in*: 'Proceedings of the 3rd Int. Conference on Geo-Computation', University of Bristol, UK.
- Spiekermann, S. (2004), General aspects of location based services, *in*: J. Schiller & A. Voisard, Hrsg., 'Location-Based Services', Morgan Kaufmann, Kapitel 1, Seiten 9–26.
- Spieß, E., Baumgartner, U., Arn, S. & Vez, C. (2002), Topografische Karten – Kartengrafik und Generalisierung, *in*: 'Kartografische Publikationsreihe', Band 16, Schweizerische Gesellschaft für Kartographie.
- Stadler, A. (2004), Verknüpfung korrespondierender Kartenelemente im Hinblick auf automatisierte Fortführung, Diplomarbeit, Technische Universität Wien, Fakultät für Mathematik und Geoinformation.
- Stanek, H. (1994), 'Datenqualität - Modellierung im GIS', *Österreichische Zeitschrift für Vermessung und Geoinformation* **1/2**, 14–20.
- Stein, B. (2006), 'Datenbankentwurf und Datenbankmodelle', Bauhaus Universität Weimar. Vorlesungsskript.
- Steiniger, S., Neun, M. & Edwardes, A. (2006), 'Foundations of Location Based Services'. Lecture Notes on LBS.
- Stell, J. & Worboys, M. (1998), Stratified map spaces: A formal basis for multi-resolution spatial databases, *in*: P. T. & N. & Chrisman, Hrsg., 'Proceedings of the 8th International Symposium on Spatial Data handling', Vancouver, Canada, Seiten 180–189.
- Sutcliffe, A. (2003), Scenario-Based Requirements Engineering, *in*: 'RE '03: Proceedings of the 11th IEEE International Conference on Requirements Engineering', IEEE Computer Society, Washington, DC, USA, Seite 320.
- Tait, M. G. (2005), 'Implementing geoportals: Applications of distributed GIS', *Computers, Environment and Urban Systems* **29**(1), 33–47.
- Teege, G. (2001), 'Ein interoperables GeoPortal zur Nutzung von Geodaten im Internet', *Zeitschrift für Vermessungswesen (ZfV)* **4**, 224–230.
- Timpf, S. (1998), Hierarchical structures in map series, Doktorarbeit, Technische Universität Wien.
- Timpf, S. & Devogele, T. (1997), New tools for multiple representations, *in*: 'Proceedings 18th ICA/ACI International Cartographic Conference, 3, Stockholm', Seiten 1381–1386.
- Timpf, S. & Frank, A. (1995), A multi-scale dag for cartographic objects, *in*: 'Proceedings of the Auto-Carto 12. ACM/ASPRS'.
- Tomaselli, L. (1994), Topological transfer: Evolving linear GIS accuracy, *in*: 'URISA 1994 conference proceedings', Seiten 245–259.
- Tryfona, N. & Egenhofer, M. (1997), 'Consistency among parts and aggregates: A computational model', *Transactions in GIS* **1**(3), 189–206.
- Uitermark, H., van Oosterom, P., Mars, N. & Molenaar, M. (1998), Propagating updates: Finding corresponding objects in a multi-source environment, *in*: T. Poiker & N. Chrisman, Hrsg., 'Proceedings of 8th Int. Symposium on Spatial Data Handling (SDH)', Vancouver, BC, Canada, Seiten 580 – 591.
- Umbaugh, S. E. (1998), *Computer Vision and Image Processing: A Practical Approach Using CVIptools*, Prentice Hall.
- van Beurden, A. U. C. J. & Douven, W. J. A. M. (1999), 'Aggregation issues of spatial information in environmental research', *International Journal of Geographical Information Science* **13**(5), 513–527.
- van den Bos, J., van Naelten, M. & Teunissen, W. (1984), 'Interactive pictorial information system for demographic and environmental planning applications', *Comput. Graph. Forum* **3**(1), 91–102.
- van Oosterom, P. (1989), A reactive data structure for geographic information systems, *in*: E. Anderson, Hrsg., 'Proceedings of the Auto-Carto 9', ASPRS & ACSM, Baltimore, MA, Seiten 665–674.
- van Oosterom, P. (1991), 'The reactive tree: A storage structure for a seamless, scaleless geographic database', *Auto Carto* **10**, 393–407.
- van Oosterom, P. (1995), The GAP-tree, an approach to on-the-fly map generalization of an area partitioning, *in*: J.-C. Müller, J.-P. Lagrange & R. Weibel, Hrsg., 'GIS and Generalization: Methodology and Practice', Taylor & Francis, London, Seiten 120–132.
- van Oosterom, P. & Schenkelaars, V. (1995), 'The development of an interactive multi-scale GIS', *International Journal of Geographical Information Systems* **9**(5), 489–507.
- van Wijngaarden, F., van Putten, J., van Oosterom, P. & Uitermark, H. (1997), Map integration - Update propagation in a multi-source environment, *in*: 'Proceedings 5th Int. Workshop on advances in Geographic Informations Systems', ACM, Las Vegas, Nevada, USA, Seiten 71–76.
- Vangenot, C., Parent, C. & Spaccapietra, S. (2002), Modeling and manipulating multiple representation of spatial data, *in*: 'Proceedings of the 10th International Symposium on Spatial Data Handling', Ottawa, Canada.
- Vauglin, F. (2000), MurMur Project, Workpackage 1: Background Analysis, Deliverable 1: Analysis of the Cartographic and Risk-Management Case Studies, Technischer Report, Institut Géographique National.
- Vauglin, F. (2001), MurMur Project, Workpackage 3: Services, Deliverable 9: Other Services, Technischer Report, IGN.

- Virrantaus, K., Markkula, J., Garmash, A., Terziyan, V., Veijalainen, J., Katanosov, A. & Tirri, H. (2001), Developing GIS-supported location-based services, *in*: 'Proceedings of the Web Information Systems Engineering, 2001', Band 2, Seiten 66–75.
- W3C, S. W. G. (2003), Mobile SVG Profiles: SVG Tiny and SVG Basic, Technischer Report, W3C.
- Walter, V. & Fritsch, D. (1999), 'Matching spatial data sets: A statistical approach', *International Journal of Geographic Information Systems* **13** (5), 445–473.
- Weibel, R. & Dutton, G. (1999), Generalising spatial data and dealing with multiple representations, *in*: P. Longley, M. Goodchild, D. Maguire & D. Rhind, Hrsg., 'Geographic Information Systems - Principles and Technical Issues', Band 1, John Wiley & Sons, Inc., Kapitel 10, Seiten 125–155.
- Whiteside, A. (2005a), OpenGIS web services architecture description, Best Practices Paper OGC 05-042r2, Open Geospatial Consortium, Inc.
- Whiteside, A. (2005b), OpenGIS web services common specification, Implementation Specification OGC 05-008, Open Geospatial Consortium, Inc.
- Yuan, S. & Tao, C. V. (1999), Development of conflation components, *in*: 'Proceedings of Geoinformatics 1999, International conference on Geoinformatics and Socioinformatics', Ann Arbor, Michigan, USA, Seiten 579–591.
- Zahn, C. T. & Roskies, R. (1972), 'Fourier descriptors for plane closed curves', *IEEE Transactions on Computers* **C-21** (3), 269–281.
- Zhou, S. & Jones, C. B. (2001), Design and implementation of multi-scale databases, *in*: 'Proceedings of the International Symposium on Spatial and Temporal Databases (SSTD)', Seiten 365–386.
- Zimányi, E. (2000), MurMur Project, Workpackage 1: Background Analysis, Deliverable 3: Multi-representation technology assessment, Technischer Report, Université Libre de Bruxelles.

Danksagung

Diese Arbeit ist im Rahmen meiner Tätigkeit beim Institut für Kartographie und Geoinformatik der Leibniz Universität Hannover entstanden. Meine Arbeit wurde finanziert durch die EU im Rahmen des EU-Projektes GiMoDig sowie dem Institut für Kartographie und Geoinformatik der Universität Hannover.

Mein Dank gilt zunächst Prof. Dr.-Ing. Monika Sester für ihre stetige Betreuung und Unterstützung während meiner gesamten Forschungstätigkeit. Bei Fragen und Problemen stand ihre Tür immer offen. Sie half mir durch konstruktive Kritik und provokative Fragen, mich nicht in dem Dickicht von Forschungsproblemen zu verlieren, und führte mich stets auf den richtigen Weg zurück. Sie ließ mir daneben allerdings auch immer die notwendigen Freiheiten in meiner Forschungsarbeit.

Für die Übernahme des Korreferates möchte ich Prof. Dr.-Ing. Liqiu Meng und Prof. Dr.-Ing. Christian Heipke danken.

Ich danke außerdem meinen Kollegen vom igk für ihre Hilfe und die angenehme Arbeitsatmosphäre, die sie mir bereitet haben. Hier fanden sich immer Mitarbeiter, mit denen man über seine Ideen diskutieren konnte oder die mich anspornten.

Den größten Teil meiner Arbeit umfasste die Mitarbeit in dem EU-Projekt GiMoDig. Aus diesem Projekt und vor allen Dingen aus dem Wissen und der Motivation der Projektmitarbeiter konnte ich den größten Teil der wissenschaftlichen Erkenntnisse ziehen. Drei Monate meiner Forschungszeit durfte ich an der Universität in Lund (Schweden) sowie an dem Finnish Geodetic Institute (FGI) in Helsinki verbringen. In dieser Zeit wurde die endgültige Idee für diese Arbeit geboren. Ich möchte Dr. Lars Harrie von der Universität in Lund danken für die zahlreichen Gespräche, gemeinsamen Überlegungen und Diskussionen, durch die sich die untersuchte Problematik herauskristallisierte.

Während meiner Zeit in Finnland durfte ich dann die herzliche Gastfreundschaft von Dr. Tiina Sarjakoski und Prof. Dr. Tapani Sarjakoski genießen. Mit Dr. Tiina Sarjakoski konnte ich während meines Aufenthaltes von Ihrem reichhaltigen Wissen auf dem Gebiet der MRDB profitieren. Gleichzeitig möchte ich Lassi Lehto danken, durch dessen technisches Know-How sowie Hilfestellungen auch über die Projektarbeit hinaus die praktische Realisierung gelang.

Daneben bin ich während meiner Forschungstätigkeit weiteren Menschen begegnet, die mir neben den wissenschaftlichen Erkenntnissen auch die während der Arbeit wichtige Motivation und Inspiration gegeben haben. Mein besonderer Dank gilt hier Annu-Maaria Nivala sowie Dr. Tumasch Reichenbacher.

Schließlich möchte ich mich bei meinen Eltern bedanken, die mir den Weg hierher erst ermöglicht haben. Sie haben mich auf diesem Weg ständig unterstützt und mir stets Vertrauen geschenkt.

Wissenschaftlicher Werdegang

Mark Hampe

Geburtstag, -ort	22.06.1973, Brilon
1983–1992	Städt. Gymnasium Marsberg
Juni 1992	Schulabschluss: Abitur
1993–1999	Studium Vermessungswesen, Universität Hannover
Juli 1999	Sudienabschluss: Dipl.-Ing.
1999–2001	Referendariat, Fachrichtung Vermessungs- und Liegenschaftswesen, Bezirksregierung Hannover
Dezember 2001	Referendariat Abschluss: Assessor
2002–2007	Wiss. Mitarbeiter am Institut für Kartographie und Geoinformatik, Leibniz Universität Hannover
seit März 2007	Mitarbeiter im Vermessungsbüro Dr. Menke, ÖbVI, Garbsen