

Master Thesis

im Rahmen des
Universitätslehrganges „Geographical Information Science & Systems“
(UNIGIS MSc) am Zentrum für GeoInformatik (Z_GIS)
der Paris Lodron-Universität Salzburg

zum Thema

„Einsatz von Open-Source und Freier Software in Geodateninfrastrukturen“

Am Fallbeispiel der Bundesanstalt für
Geowissenschaften und Rohstoffe

vorgelegt von

Dipl.-Geogr. Nina Helle-Mildt
U1298, UNIGIS MSc Jahrgang 2006

Zur Erlangung des Grades
„Master of Science (Geographical Information Science & Systems) – MSc(GIS)“

Gutachter:
Ao. Univ. Prof. Dr. Josef Strobl

Braunschweig, den 29. August 2008

Erklärung

Ich erkläre, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat. Die aus fremden Quellen direkt oder indirekt übernommenen Inhalte sind als solche kenntlich gemacht.

Braunschweig, den 29. August 2008

.....

Nina Helle-Mildt

Kurzfassung

Ein Großteil der vorhandenen Geodaten wird in der öffentlichen Verwaltung geschaffen und gepflegt und ist somit weitestgehend ein öffentliches Gut. Diese räumlichen Daten gilt es durch den Aufbau von interoperablen Geodateninfrastrukturen (GDI) in Wert zu setzen, zu fördern und für Bürger, Wirtschaft sowie Wissenschaft nutzbar zu machen. Dabei entstehen in Deutschland aus den gleichen zentralen und bundeseinheitlichen Vorschriften und Vorgaben (INSPIRE-Richtlinie) bei fast allen öffentlichen Verwaltungen ähnliche Fragestellungen, Anforderungen und Rahmenbedingungen, die eine gemeinsame Lösungsfindung nahelegen. Um eine Interoperabilität zwischen den verschiedenen GDIs, aber auch zwischen den einzelnen Komponenten einer GDI zu gewährleisten, kommt bei der Umsetzung der Einhaltung gemeinsamer Normen und Standards eine zentrale Bedeutung zu. Gleiches gilt bei den erfolgreichen Open-Source und Freie Software-Projekten, bei denen die Einhaltung von „offenen“ Standards die wesentliche Grundlage und die Voraussetzung für deren zum Teil weltweite Verbreitung darstellen.

Ausgangspunkt der vorliegenden Arbeit ist deshalb die Hypothese, dass sich Open-Source- und Freie Software (OS/FS) besonders gut für den Aufbau von Geodateninfrastrukturen in der öffentlichen Verwaltung eignen. Diese Hypothese gilt es am Fallbeispiel der Umsetzung einer GDI des seismischen Zentralobservatoriums (Seismic Data Analysis Center, SDAC) in der Bundesanstalt für Geowissenschaften und Rohstoffe (BGR) in Hannover zu belegen.

Aufbauend auf einer eingehenden Betrachtung der Rahmenbedingungen und Anforderungen an den Aufbau von GDIs sowie einer Analyse der wesentlichen Vorteile und Grenzen des Einsatzes von OS/FS im Allgemeinen und in der öffentlichen Verwaltung im Speziellen wurde ein Systemkonzept (Szenario) für den Aufbau einer GDI auf Basis von OS/FS erstellt. Die Implementierung des Konzepts bildete die Grundlage für die Bewertung der Umsetzbarkeit, des Mehrwerts und der Grenzen des Einsatzes von OS/FS beim Aufbau von GDIs in der öffentlichen Verwaltung.

Im Ergebnis lässt sich feststellen, dass die Hypothese unter den gegebenen Rahmenbedingungen bestätigt werden konnte. Die Implementierung einer lauffähigen GDI-Lösung, über die seismische Geofachdaten (Erdbeben) bereitgestellt werden können, war mit OS/FS gut möglich (<http://www.seismologie.bgr.de/sdac/erdbeben/kartenanwendung/unigis.html>). Ausgewählt wurden dafür die Produkte OpenLayers (WebMapping Client), der UMN Map-Server (WMS), der Apache Webserver und eine PostgreSQL/PostGIS-Datenbank. Ferner ließ sich feststellen, dass die verwendeten OS/FS Produkte gut an den Bedürfnissen von GDI-Lösungen ausgerichtet sind. Anpassungen des Source Codes waren erst gar nicht er-

forderlich. Auch die Konfiguration der einzelnen Komponenten sowie die Einrichtung der Schnittstellen zwischen den verwendeten Programmen waren ohne über das normale Maß hinausgehende Probleme möglich.

Hervorzuheben ist in diesem Zusammenhang aber auch, dass die Umsetzung der Systemlösung unter günstigen organisatorischen, technischen und personellen Rahmenbedingungen und ohne die mitunter sonst zu erwartenden Mehraufwände in Zusammenhang mit der Umstellung auf OS/FS erfolgen konnte. So ist die IT-Infrastruktur und Datenhaltung im SDAC der BGR bereits seit 2007 fast vollständig auf OS/FS umgestellt. Entsprechendes Know-how und Erfahrung beim Einsatz von OS/FS war im IT-Bereich daher bereits vorhanden.

Die Arbeit zeigt, dass die organisatorischen, rechtlichen und politischen Rahmenbedingungen den Einsatz von OS/FS in der öffentlichen Verwaltung begünstigen. Die wesentlichen Aspekte dabei sind, dass Softwarelösungen in der öffentlichen Verwaltung nicht wie in der Privatwirtschaft zur Erzielung eines Wettbewerbsvorteils eingesetzt werden. Somit liegen die kommerziell bedingten Hürden, die einer Verbreitung und Weitergabe von Know-how und Nutzungsrechten häufig entgegenstehen, im öffentlichen Sektor gerade nicht vor. Auch ermöglichen diese Rahmenbedingungen die Kooperation zwischen Verwaltungseinheiten bei der Entwicklung und beim Einsatz von OS/FS.

Abstract

A large part of the existing spatial data is produced in the public administration and is therefore a valuable public good which should be made accessible when developing interoperable spatial data infrastructures (SDI) and be provided to citizens, businesses and science. Due to equivalent or identical regulatory frameworks and guidelines (i.e. INSPIRE-Directive), implementation solutions for SDI are subject to very similar requirements. This yields a common solution in almost all public administration entities nationwide. To ensure interoperability between different SDIs, but also between the various components of a SDI complying with the applicable standards plays a major role. The same applies to the successful open-source and free software projects where the compliance of the so called "open" standards is an essential requirement and also the prerequisite for their global dissemination.

The starting point of this work is thus the hypothesis that open-source and free software (OS/FS) is well suitable for the development of spatial data infrastructures in the public administration. Within this work, the hypothesis is verified by means of a case study of the implementation of relevant SDI components at the Seismic Data Analysis Center (SDAC) of the Federal Institute for Geosciences and Natural Resources (BGR) in Hanover.

Based on a thorough consideration of general conditions and requirements for establishing SDIs as well as an analysis of the key advantages and limitations of OS/FS in general and in public administration in particular, a system concept for a SDI was set up on the basis of OS/FS. The implementation of the concept formed the basis for evaluating the feasibility, the added value and the limits of OS/FS deployment for the establishment of SDIs in public administration.

As a result, it can be said that the hypothesis could be confirmed under the given conditions. The implementation of a serviceable SDI-solution, with which seismic spatial data (earthquakes) is available, was possible using OS/FS (<http://www.seismologie.bgr.de/sdac/erdbeben/kartenanwendung/unigis.html>). As implementation solution OpenLayers (web mapping client), the UMN MapServer (WMS), the Apache web server and a PostgreSQL/PostGIS database were chosen. It was also found that the chosen OS/FS products suited the specific requirements of SDI solutions quite well. Adjusting the source code was not necessary. The configuration of individual components and the setting up the interfaces between the programs that were used was possible without excessive workload.

However, it must also be considered in this assessment that the implementation of the system concept took place under very favorable organizational, technical and human resource conditions. The additional workload which is usually being experienced when introducing new software or when converting to OS/FS products could be avoided as the IT infrastructure and data storage in the SDAC was already entirely converted to OS/FS in 2007. Consequently, knowhow and experience in the use of OS/FS, which constitutes a significant key success factor, was already available at the SDAC.

This thesis shows that the organizational, legal and political framework fosters the use of OS/FS in public administrations. One of the major aspects is that software in the public administration is usually not deployed to achieve a competitive advantage over competitors. Thus, these commercially induced barriers that usually undermine the dissemination and the free transfer of knowhow and the right of use and deployment of software could be eliminated. This also fosters cooperation between administrative units in the development and deployment of OS/FS.

Inhaltsverzeichnis

| | |
|---|-----|
| Abbildungsverzeichnis..... | ii |
| Tabellenverzeichnis | iii |
| Codeverzeichnis..... | iv |
| Liste der verwendeten Abkürzungen und Symbole | iv |
| Einleitung..... | 1 |
| 1 Eine Infrastruktur für Geodaten | 5 |
| 1.1. Definition und Komponenten von Geodateninfrastrukturen | 5 |
| 1.2. Aufbau von Geodateninfrastrukturen – politische und rechtliche Rahmenbedingungen | 8 |
| 1.2.1 Geodateninfrastrukturen in Europa | 8 |
| 1.2.1.1 <i>INSPIRE – Ziel einer europäischen GDI</i> | 9 |
| 1.2.1.2 <i>Umsetzung der INSPIRE-Richtlinie</i> | 10 |
| 1.2.2 Geodateninfrastrukturen in Deutschland | 11 |
| 1.2.2.1 <i>Rechtliche Rahmenbedingungen – ein Überblick</i> | 12 |
| 1.2.2.2 <i>Aufbau einer Geodateninfrastruktur Deutschland (GDI-DE)</i> | 13 |
| 1.3. Interoperabilität durch Nutzung von OpenGIS® | 17 |
| 1.3.1 Normen und Standards für Geodaten und Geodienste | 17 |
| 1.3.2 Das Open Geospatial Consortium (OGC) | 18 |
| 1.3.3 Rahmenarchitektur für Geodienste – das OWS Service Framework (OSF) | 19 |
| 1.3.4 Webbasierte Geodienste | 22 |
| 1.3.4.1 <i>Catalogue Service for the Web</i> | 22 |
| 1.3.4.2 <i>Web Map Service</i> | 24 |
| 1.3.4.3 <i>Web Feature Service</i> | 25 |
| 1.3.5 Bereitstellung und Bezug von Geodaten | 26 |
| 2 Einsatz von Open-Source und Freier Software in Geodateninfrastrukturen..... | 27 |
| 2.1. Grundlagen: Open Source und Freie Software | 27 |
| 2.1.1 Begrifflichkeiten und Konzepte..... | 27 |
| 2.1.1.1 <i>Open Source und Freie Software</i> | 27 |
| 2.1.1.2 <i>Kommerzielle und proprietäre Software</i> | 30 |
| 2.1.1.3 <i>Kategorien von Software – Zusammenfassung</i> | 30 |
| 2.1.2 Lizenzmodelle von OS/FS..... | 31 |
| 2.1.3 Software-Projekte: Entwicklungszyklus vs. Lebenszyklus | 33 |
| 2.1.4 OS/FS Entwicklungs- und Kommunikationswerkzeuge | 35 |
| 2.2. Warum Open Source und Freie Software? | 37 |
| 2.2.1 Chancen und Grenzen des Einsatzes von Open Source und Freier Software..... | 37 |
| 2.2.1.1 <i>Vorteile und Chancen</i> | 38 |
| 2.2.1.2 <i>Nachteile und Risiken</i> | 41 |
| 2.2.2 Einsatz von Open Source und Freier Software im öffentlichen Sektor..... | 44 |
| 2.2.2.1 <i>Rahmenbedingungen für den Einsatz von OS/FS im öffentlichen Sektor</i> | 44 |
| 2.2.2.2 <i>Bereitstellung von Geoinformationen im öffentlichen Sektor</i> | 46 |
| 2.3. GDI Systemarchitektur mit Open Source und Freier Software | 46 |

| | | |
|---------|--|----|
| 2.3.1 | Beispiel einer einfachen Kartenanfrage..... | 47 |
| 2.3.2 | Datenhaltungskomponente | 49 |
| 2.3.3 | Webserver..... | 51 |
| 2.3.4 | MapServer | 52 |
| 2.3.5 | Darstellungskomponente / Client / Web-Mapping-Client..... | 54 |
| 2.3.6 | Metadaten Management | 55 |
| 3 | Praxisbeispiel – Implementierung des Systementwurfs..... | 56 |
| 3.1. | Nutzerorientierte Bereitstellung von Geodaten bei der Bundesanstalt für Geowissenschaften und Rohstoffe..... | 56 |
| 3.1.1 | Rechtliche und politische Rahmenbedingungen – Auswirkungen für die Bundesanstalt für Geowissenschaften und Rohstoffe | 56 |
| 3.1.1.1 | <i>INSPIRE-Richtlinie</i> | 56 |
| 3.1.1.2 | <i>GDI-DE</i> | 58 |
| 3.1.2 | Aufbau einer Geofachdateninfrastruktur bei der BGR..... | 58 |
| 3.2. | Fallbeispiel – Bereitstellung von seismischen Daten am seismologischen Zentralobservatorium der BGR (SDAC)..... | 60 |
| 3.2.1 | Identifizierung von Nutzergruppen | 61 |
| 3.2.2 | Einsatz von OS/FS-Produkten am SDAC | 62 |
| 3.2.3 | Die Datenbasis..... | 62 |
| 3.2.3.1 | <i>Der Datenbestand am SDAC</i> | 62 |
| 3.2.3.2 | <i>Datenauswahl</i> | 63 |
| 3.2.3.3 | <i>Datenhaltung in einer PostGIS Datenbank</i> | 64 |
| 3.2.4 | Serverseitige Implementierung I – der Webserver | 67 |
| 3.2.5 | Serverseitige Implementierung II – der UMN MapServer | 67 |
| 3.2.5.1 | <i>Konfiguration des UMN MapServers</i> | 68 |
| 3.2.5.2 | <i>Test der Mapdatei auf OGC-Konformität</i> | 72 |
| 3.2.6 | Der Client – Schnittstelle zum Nutzer | 76 |
| 3.2.7 | Aufnahme von Metainformationen in den Geodatenkatalog des Geozentrums Hannover..... | 82 |
| 4 | Bewertung und Ausblick..... | 83 |
| 4.1. | Bewertung | 83 |
| 4.1.1 | Bewertung der Entwicklung eines Szenarios zur Umsetzung der INSPIRE-Richtlinie im Rahmen der GfDI-BGR auf Basis von OS/FS (GDI-Systemarchitektur)... | 83 |
| 4.1.2 | Bewertung der eingesetzten OS/FS-Produkte | 84 |
| 4.1.2.1 | <i>OpenLayers</i> | 85 |
| 4.1.3 | Bewertung von Mehrwert, Umsetzbarkeit und Grenzen des Einsatzes von OS/FS beim Aufbau von GDIs | 86 |
| 4.2. | Zusammenfassende Bewertung und Ausblick..... | 91 |

Abbildungsverzeichnis

| | | |
|---------|---|----|
| Abb. 1: | Struktur der Master Thesis | 3 |
| Abb. 2: | Vertikale und horizontale Vernetzung von Geodateninfrastrukturen über verschiedene administrative Ebenen hinweg | 5 |
| Abb. 3: | Komponenten von Geodateninfrastrukturen | 6 |
| Abb. 4: | GDI-Hierarchien in Europa | 11 |

| | |
|---|----|
| Abb. 5: Stufenweiser Aufbau der GDI-DE | 14 |
| Abb. 6: Aufbau der GDI-DE – Gremienstruktur mit abgestuften Kompetenzen und Aufgaben | 16 |
| Abb. 7: UML-Komponentendiagramm der OSF-Rahmenarchitektur (vereinfachte Darstellung) | 21 |
| Abb. 8: Referenzarchitektur für Kataloge | 23 |
| Abb. 9: Das Publish-Find-Bind-Paradigma der OpenGIS® Service Architektur | 26 |
| Abb. 10: Lizenzkategorien | 33 |
| Abb. 11: Entwicklungszyklus vs. Lebenszyklus | 34 |
| Abb. 12: Der Repository-Prozess | 36 |
| Abb. 13: Die „Kopieren-Verändern-Zusammenfügen Lösung“ | 37 |
| Abb. 14: Handlungsfelder öffentlicher Verwaltungen | 44 |
| Abb. 15: Einfache Client-Server Architektur | 48 |
| Abb. 16: Marktanteile der Top Webserver in allen Branchen (August 1995 bis Juni 2008) | 51 |
| Abb. 17: Hauptbestandteile einer UMN MapServer-Anwendung | 53 |
| Abb. 18: Seismogramm eines Erdbebens | 63 |
| Abb. 19: Ausschnitt aus dem vorläufigen monatlichen Datenkatalog vom März 2008 | 63 |
| Abb. 20: Speicherschema für die Geodatenhaltung in PostGIS | 65 |
| Abb. 21: Datenbankstruktur von „seis_ows“ | 66 |
| Abb. 22: UMN MapServer Version und unterstützte Daten- und Ausgabeformate | 67 |
| Abb. 23: Die Mapdatei – wesentliche Objekte und dazugehörige Parameter (Auswahl) | 68 |
| Abb. 24: Der GetCapabilities-Request | 73 |
| Abb. 25: Struktur des GetCapabilities-Dokuments | 73 |
| Abb. 26: Der GetMap-Request | 74 |
| Abb. 27: Antwort auf den GetMap-Request | 74 |
| Abb. 28: Der GetFeatureInfo-Request | 75 |
| Abb. 29: Darstellung des WMS in Quantum GIS | 75 |
| Abb. 30: Komponenten der Clientanwendung OpenLayers | 76 |
| Abb. 31: OpenLayers-Anwendung: Seismische Ereignisse in Deutschland und weltweit | 81 |
| Abb. 32: Eintrag des WMS in den GeoDak | 82 |
| Abb. 33: Entwickelte Systemarchitektur auf Basis von OS/FS-Produkten | 83 |

Tabellenverzeichnis

| | |
|---|----|
| Tab. 1: Obligatorische Spezifikationen der GDI-DE | 22 |
| Tab. 2: Unterstützte Operationen des Catalogue Service for the Web (CSW) | 23 |
| Tab. 3: Unterstützte Operationen des Web Map Services (WMS) | 24 |
| Tab. 4: Unterstützte Operationen des Web Feature Services (WFS) | 25 |
| Tab. 5: Kategorien von Software | 31 |
| Tab. 6: Lizenzen der verwendeten Software | 33 |
| Tab. 7: Vergleich der Stärken von OS/FS und proprietärer Software. | 43 |
| Tab. 8: Beispiele für im WKT-Format dargestellte Geometrie-Objekttypen | 50 |
| Tab. 9: Eingesetzte Software | 62 |
| Tab. 10: Ausschnitt aus der Tabelle „beben“ der Datenbank PostGIS- „seis_ows“ (seismische Parameterdaten) | 66 |
| Tab. 11: Zusammenfassende Bewertung des OS/FS-Einsatzes | 91 |

Codeverzeichnis

| | |
|---|----|
| Code 1: Der Header der Mapdatei | 69 |
| Code 2: Weitere Objekte der Header-Sektion | 69 |
| Code 3: Die Web-Sektion der Mapdatei | 71 |
| Code 4: Beispiel eines Layers | 72 |
| Code 5: Integration der API von OpenLayers und Google Maps | 78 |
| Code 6: Einbindung von Hauptkarte und Kartenebenen über die Klassen ‚OpenLayers.Map‘ und ‚OpenLayers.Layer.Google‘ bzw. ‚OpenLayers.Layer.WMS‘ | 79 |
| Code 7: Einbindung von Bedienelementen | 80 |
| Code 8: Abfrage des GetFeatureInfo-Requests | 80 |

Liste der verwendeten Abkürzungen und Symbole

| | |
|----------------|--|
| AK | Arbeitskreis |
| BGR | Bundesanstalt für Geowissenschaften und Rohstoffe |
| BKG | Bundesamt für Kartographie und Geodäsie |
| BMI | Bundesinnenministerium |
| BMWI | Bundesministerium für Wirtschaft |
| CdS | Chefs der Staats- und Senatskanzleien Deutschland |
| CSS | Cascading Style Sheets |
| CSW | Catalogue Service for the Web |
| ESDI | European Spatial Data Infrastructure |
| GDI | Geodateninfrastruktur |
| GDI-DE | Geodateninfrastruktur Deutschland |
| GeoDAK | Geodatenkatalog der BGR |
| GfDI-BGR | Geofachdateninfrastruktur der BGR |
| GIS | Geoinformationssystem |
| GIW-Kommission | Kommission für Geoinformationswirtschaft |
| GIW GSt. | Geschäftsstelle der Geoinformationswirtschaft |
| GKSt GDI-DE | Geschäfts- und Koordinierungsstelle GDI-DE |
| GML | Geographic Markup Language |
| GPL | General Public License (GNU GPL) |
| GSDI | Global Spatial Data Infrastructure |
| FS | Freie Software |
| FSF | Free Software Foundation |
| FOSS | Freie und Open Source Software |
| HTTP | Hypertext Transfer Protokoll |
| IE | Internet Explorer |
| IFG | Informationsfreiheitsgesetz |
| IR | Implementing Rules (INSPIRE Durchführungsbestimmungen) |
| IMAGI | Interministerieller Ausschuss für Geoinformationen |

| | |
|-------------|---|
| IMAGI GKSt. | Geschäfts- und Koordinierungsstelle des IMAGI |
| INSPIRE | Infrastructure for Spatial Information in Europe |
| ISO | International Organization for Standardization |
| ISO/TC 211 | ISO Technical Committee 211 |
| IT | Informationstechnologie |
| IWG | Informationsweiterverwendungsgesetz |
| KML | Keyhole Markup Language |
| LG GDI-DE | Lenkungsremium GDI-DE |
| LGPL | Lesser General Public License |
| LMO | Legally Mandated Organization |
| NGDB | Nationale Geodatenbasis |
| OGC | Open Geospatial Consortium |
| OSD | Open Source Definition |
| OSI | Open Source Initiative |
| OSS | Open Source Software |
| OS/FS | Open Source und Freie Software |
| OWS | Open Web Service |
| SDAC | Seismologisches Datenzentrum, angesiedelt bei der BGR |
| SDI | Spatial Data Infrastructure (engl. GDI) |
| SGD | Staatlich Geologischer Dienst |
| URL | Uniform Resource Locator |
| WFS | Web Feature Service |
| WMS | Web Map Service |
| XML | Extend Markup Language |
| [...] | Auslassungen im Code |

Einleitung

Anlass und Hintergrund der Aufgabenstellung

Der Aufbau von Geodateninfrastrukturen (GDI) wird derzeit weltweit auf den unterschiedlichsten Verwaltungsebenen konzipiert und umgesetzt. Maßgeblich für den Bereich der Europäischen Union ist die am 15. Mai 2007 in Kraft getretene INSPIRE-Richtlinie (Infrastructure for Spatial Information in Europe) zum Aufbau einer europäischen GDI. Die Richtlinie legt den zeitlichen Rahmen für die Umsetzung fest und nennt die Themengebiete, zu denen geo(referenzierte) Basis- und Fachdaten sowie Metadaten über interoperable Geodienste für Bürger, Wirtschaft und Wissenschaft bereit zu stellen sind.

Der Aufbau der europäischen GDI erfolgt generell auf Basis der nationalen GDIs der Mitgliedstaaten. In Deutschland ist die GDI-DE im Sinne von INSPIRE der nationale Ansprechpartner der EU. Ihr Aufbau wiederum erfolgt gemeinsam zwischen Bund, Ländern und Kommunen und stützt sich auf Länder- und regionale GDIs. Ein Beispiel auf Bundesebene ist das derzeit in der konzeptionellen Phase befindliche Geodatenportal der Bundesanstalt für Geowissenschaften und Rohstoffe (BGR) in Hannover.

Nicht zuletzt bedingt durch die beschriebenen politischen und rechtlichen Rahmenbedingungen ist der Aufbau von GDIs schon seit einigen Jahren ein viel beachtetes Thema in der öffentlichen Verwaltung. Ein Großteil der vorhandenen Geodaten wird hier erarbeitet und ist somit weitgehend ein öffentliches Gut, welches es gilt, in Wert zu setzen, zu fördern und für Bürger, Wirtschaft und Wissenschaft nutzbar zu machen. Dabei erfordert die im Rahmen von GDIs angestrebte organisations- und systemübergreifende Nutzung von räumlichen Informationen eine Interoperabilität, die sich nur durch eine konsequente Nutzung von verbindlichen offenen Normen und Standards erreichen lässt. Bedeutende Standardisierungsorganisationen im Bereich der Geodaten und Geodienste sind das Open Geospatial Consortium (OGC) und die International Standardization Organization (ISO).

Auch in der Fachwelt der Geoinformatik sowie im Bereich der Entwicklung von Open-Source und Freier Software (OS/FS) wird der Aufbau von GDIs aktiv begleitet und mitgestaltet. Der Einsatz von OS/FS erleichtert dabei die Schaffung der geforderten Interoperabilität der Systeme. Grundlegend ist die Verwendung von – im Bereich von OS/FS fest etablierten – sog. offenen Standards und Schnittstellen, um eine möglichst große Kompatibilität zu anderen Software-Produkten zu gewährleisten. Gepaart mit dem modularen Aufbau von OS/FS-Produkten, der einen Austausch bzw. ein Hinzufügen einzelner Komponenten erlaubt, und der Möglichkeit bedarfsgerechter Anpassungen des Source-Codes können unter

Verwendung von OS/FS-Produkten aufgebaute GDIs an sich verändernde Rahmenbedingungen und Anforderungen angepasst werden.

Zielsetzung und Aufgabenstellung

Die vorliegende Arbeit folgt der **Hypothese**, dass sich *Open-Source- und Freie Software besonders gut für den Aufbau von Geodateninfrastrukturen in der öffentlichen Verwaltung eignen*. Vor diesem Hintergrund wird der **Aufbau von Geodateninfrastrukturen mit Open-Source- und Freier Software** beleuchtet.

Als Fallbeispiel wurde der Aufbau der Geodateninfrastruktur der Bundesanstalt für Geowissenschaften und Rohstoffe (GfDI-BGR) in Hannover gewählt. Diese befindet sich derzeit in der konzeptionellen Planungsphase, mit der Besonderheit, dass bereits einzelne GDI-Komponenten bestehen und in das fachbereichsübergreifend zu erarbeitende Konzept integriert werden sollen. Bestehende GDI-Komponenten sind dabei ein Geodatenkatalog (einschließlich eines Katalogdienstes, CSW), verschiedene Web Map Services (WMS) sowie drei Fachanwendungen, die selber WMS-Dienste integrieren. Beim Aufbau einer GfDI-BGR ist also eine adaptionsfähige und dynamische Vorgehensweise gefragt, nur so kann es gelingen, bestehende GDI-Komponenten in ein Gesamtkonzept zu integrieren.

Der Aufbau einer GfDI-BGR ist mittel- bis langfristig angelegt. Gleichzeitig besteht im Fachbereich Seismologie die Notwendigkeit, zeitnah seismische Daten mittels einer weiteren Fachanwendung einer breiten Öffentlichkeit bereitzustellen. Die Herausforderung besteht darin, eine solche Anwendung unter Nutzung von grundlegenden Standards und OS/FS-Produkten so aufzubauen, dass die einzelnen Komponenten zu einem späteren Zeitpunkt nahtlos in eine GfDI-BGR integriert werden können.

In diesem Zusammenhang verfolgt die vorliegende Arbeit folgende Zielsetzungen:

Übergeordnete Zielsetzungen:

- Untersuchung der Eingangs aufgestellten Hypothese (vgl. oben)
- Entwicklung eines Szenarios zur Umsetzung der INSPIRE-Richtlinie im Rahmen der GfDI-BGR auf Basis von OS/FS (GDI-Systemarchitektur)
- Bewertung von Mehrwert, Umsetzbarkeit und Grenzen des Einsatzes von OS/FS beim Aufbau von GDIs im Rahmen eines Praxistests

Operative Zielsetzungen:

- Aufbau einer Fachanwendung ‚Seismologie‘ im Kontext der GfDI-BGR auf Basis von OS/FS
- Implementierung einer Datenhaltungskomponente
- Implementierung eines WMS zur Bereitstellung von Daten aus dem Annex III der INSPIRE-Richtlinie zum Thema ‚Zonen natürlicher Risiken‘

- Aufbau einer Clientanwendung

Struktur der Arbeit

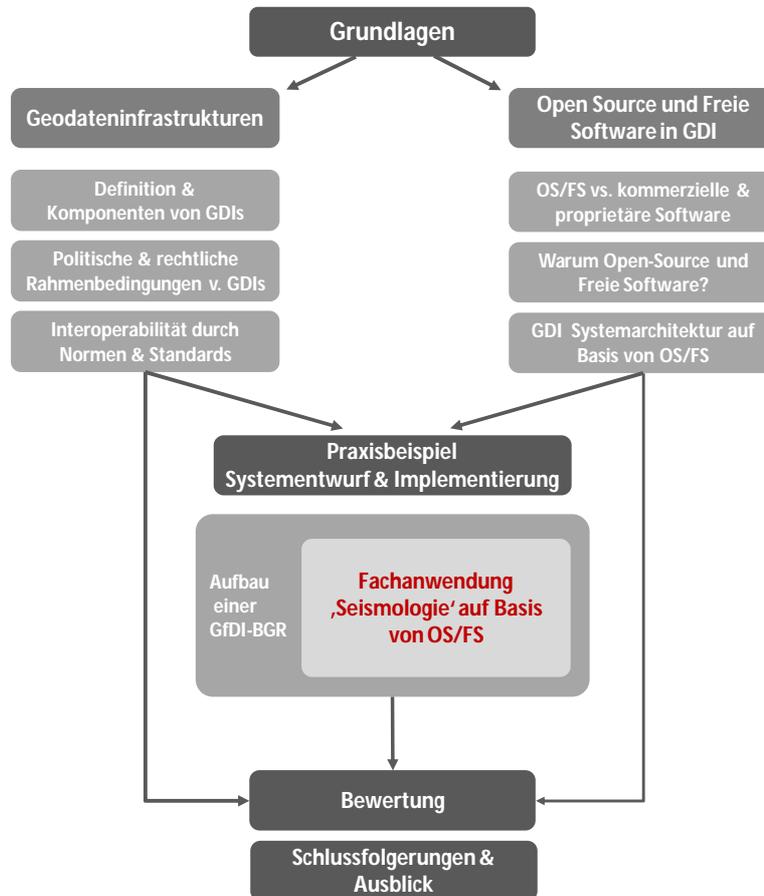


Abb. 1: Struktur der Master Thesis

Die vorliegende Arbeit gliedert sich in vier Teilbereiche, zwei Grundlagenbereiche, einen Praxisteil und eine Bewertung.

Abschnitt 1 befasst sich mit der Bereitstellung von Geoinformationen über Geodateninfrastrukturen (GDI). Nach einer Klärung der Begrifflichkeiten und der Definition der wesentlichen Komponenten einer GDI [Abschnitt 1.1], werden die rechtlichen und politischen Rahmenbedingungen von GDIs in Europa (INSPIRE) und Deutschland (GDI-DE) thematisiert [Abschnitt 1.2]. Abschnitt 1.3 befasst sich dann mit dem in Zusammenhang mit dem Aufbau von GDIs zentralen Aspekt der Interoperabilität durch die Nutzung von OpenGIS®.

Schwerpunkt von Abschnitt 2 bildet der Einsatz von Open-Source und Freier Software (OS/FS) in GDIs. Zunächst werden wesentliche Konzepte von OS/FS im Vergleich zu kommerziell-proprietärer Software herausgearbeitet [Abschnitt 2.1]. Auf dieser Basis folgt dann ein Vergleich der Vorteile und Grenzen von OS/FS sowie eine Analyse der Rahmenbedingungen für die Bereitstellung von Geoinformationen mithilfe von OS/FS im öffentlichen Sektor. Abschnitt 2 schließt mit der Erstellung eines Szenarios für eine GDI-Systemarchitektur auf Basis von OS/FS ab.

In Abschnitt 3 erfolgt dann die praktische Implementierung einer Fachanwendung für seismische Daten im Umfeld der GfDI-BGR unter weitestgehender Verwendung von OS/FS und unter Berücksichtigung wesentlicher Komponenten von GDIs.

Die Ergebnisse werden einer Gesamtbewertung [Abschnitt 4] unter Bezugnahme auf die der Arbeit zu Grunde gelegten Hypothese sowie der eingangs definierten übergeordneten und operativen Zielsetzungen unterzogen. Abschließend erfolgen eine zusammenfassende Diskussion der gewonnen Erkenntnisse sowie ein Ausblick.

Adressaten der Arbeit und weitere Hinweise

Die vorliegende Arbeit adressiert ein breites Publikum, das sich wie folgt zusammensetzt:

- Entscheidungsträger in öffentlichen Verwaltungen, die mit Entscheidungen zum Softwareeinsatz in Hinsicht auf die Umsetzung der INSPIRE-Richtlinie konfrontiert sind
- Fachleute aus der Geoinformatik, die sich für den GDI-Aufbau auf der Basis von OS/FS interessieren, selbst aber noch keine bzw. nur geringe Erfahrungen in diesem Bereich gesammelt haben;
- IT-Fachkräfte, die im Rahmen der Administration der IT-Systemstruktur eine Übersicht zu den Anforderungen des Softwareeinsatzes im Umfeld einer GDI benötigen
- Fachkräfte im Bereich Seismologie, die vor allem im Praxisteil der Arbeit [Abschnitt 3] einen Einblick hinter die Kulissen der neu entstandenen Fachanwendung ‚Seismologie‘ erhalten.

Entsprechend dem adressierten Zielpublikum werden beim Leser ein allgemeines technisches Verständnis, jedoch keine vertieften Kenntnisse in den Bereichen OpenGIS und GDI bzw. OS/FS vorausgesetzt. Wesentliche Sachverhalte werden kurz erklärt. Insbesondere der Einsatz der OS/FS-Produkte wird anhand von Praxisbeispielen erläutert.

Themen, die in der Arbeit nicht behandelt werden

Nach der notwendigen Einordnung der Arbeit in einen Gesamtzusammenhang liegt der Fokus auf der technischen Realisierbarkeit von GDIs (bzw. deren Komponenten) mit OS/FS-Projekten im Bereich der öffentlichen Verwaltung. Um den Umfang einer Master Thesis nicht zu überschreiten, wurden folgende Themen nicht oder nur am Rande behandelt:

- IT-Sicherheit – Implementierung erfolgte im Umfeld einer bestehenden IT-Architektur
- Aufbau eines WCS (Web Catalogue for the Web) – lediglich konzeptionelle Beschreibung, da im Rahmen der im Aufbau befindlichen GfDI-BGR ein solcher bereits realisiert worden ist
- keine Wirtschaftlichkeitsbetrachtung von OS/FS-Produkten

1 Eine Infrastruktur für Geodaten

1.1. Definition und Komponenten von Geodateninfrastrukturen

Der Aufbau von Geodateninfrastrukturen (GDI) wird derzeit weltweit auf den unterschiedlichsten administrativen Ebenen [vgl. Abb. 2], aber auch innerhalb eines Unternehmens bzw. zwischen mehreren Unternehmen konzipiert und umgesetzt. Ziel der Bemühungen ist es, die Verfügbarkeit und effektive Nutzung von Geoinformationen in unterschiedlichen Anwendungszusammenhängen durch eine Vernetzung bestehender Datenbestände zu fördern. Dazu müssen für den Datenaustausch und -zugang hinderliche Barrieren, wie beispielsweise eine durch einen jahrzehntelangen Aufbau monolithischer Geoinformationssysteme (GIS) entstandene Heterogenität an Schnittstellen und Daten, überwunden werden. Weitere Ziele für den Aufbau und Betrieb von GDIs sind die Schaffungen einer größeren Transparenz am Geoinformationsmarkt sowie dessen Stimulation.¹ Es gilt, Mehrwerte für Verwaltung, Wirtschaft, Wissenschaft und Gesellschaft zu schaffen.

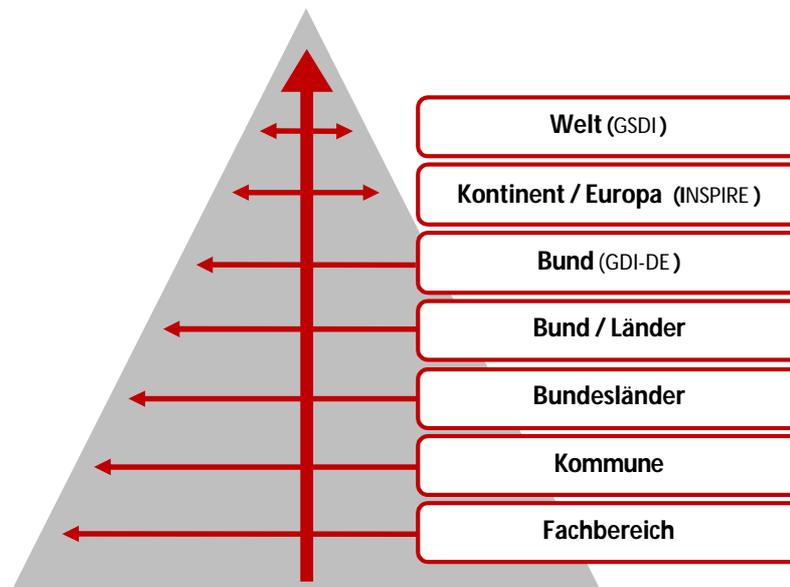


Abb. 2: Vertikale und horizontale Vernetzung von Geodateninfrastrukturen über verschiedene administrative Ebenen hinweg²

Geodateninfrastrukturen – Definition

Unter einer GDI wird eine anwenderfreundliche Bereitstellung von Geoinformationen mithilfe von standardisierten Diensten verstanden, die ebenen- und fachübergreifend auf verteilt vorliegende Geodaten zugreifen.³ Der Datenaustausch erfolgt mithilfe von Standard-

¹ FRONEFELD et. al. 2004: 7.

² Quelle: frei nach BERNARD et al. 2005: 7.

³ IMAGI 2004, S. 14.

protokollen über ein physisches Datennetz, z. B. das Internet. Aus Sicht der Anwender ist eine GDI im Wesentlichen durch folgende Merkmale gekennzeichnet:⁴

- Einfacher Online-Zugang zu den Daten, z.B. mittels einer in einem Browser aufrufbaren Benutzerschnittstelle (Client),
- Erlangung von Auskünften (eventuell auch Download und Druck von Datenzusammenstellungen) ohne Notwendigkeit komplexer Werkzeuge (z. B. Geoinformationssysteme, GIS) sowie entsprechendes Expertenwissen,
- Möglichkeit der Interaktion mit den Daten,
- Möglichkeit einer Überlagerung von Daten aus unterschiedlichen Quellen sowie
- Zugriff auf zentral gepflegte Daten und damit Gewährleistung einer hohen Datenaktualität.

Aus Systemsicht lässt sich dieser Merkmalskatalog weiter ergänzen:⁵

- Zusammensetzung einer GDI aus verschiedenen, unabhängigen Komponenten [vgl. Abb. 3],
- Verwendung internationaler Standards im Bereich der Daten, Dienste und Netzwerk-Technologie sowie
- Verbleib der Daten beim Datenerzeuger (verteilte Datenhaltung).

Komponenten von Geodateninfrastrukturen – Überblick

Wie bereits oben angesprochen, ist eine GDI durch eine Reihe eigenständiger Komponenten charakterisiert, die in enger Beziehung zueinander stehen. Erst durch ihr Zusammenwirken können Mehrwerte geschaffen werden. Abb. 3 gibt einen Überblick zu den wesentlichen Komponenten einer GDI:

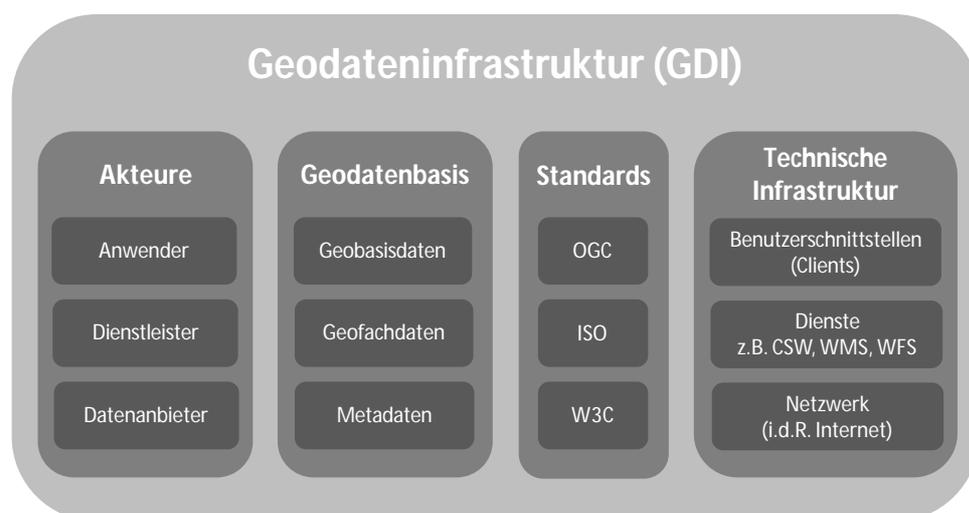


Abb. 3: Komponenten von Geodateninfrastrukturen

⁴ FRONEFELD et. al. 2004: 7.

⁵ FRONEFELD et. al. 2004: 8.

Der Vollständigkeit halber seien an dieser Stelle weitere GDI-Komponenten erwähnt, die nicht in Abb. 3 integriert worden sind. So ließe sich der Bereich der technischen Infrastruktur noch um Sicherheits- und Zugriffskontrollmechanismen sowie Abrechnungskomponenten erweitern. Ebenfalls Bestandteil von GDIs sind Finanzierungs- und Tarifierungsmodelle, auch wenn hier in der Praxis noch Entwicklungsbedarf besteht.⁶

Komponenten von Geodateninfrastrukturen – Zusammenwirken

Erst durch das Zusammenwirken der verschiedenen Komponenten einer GDI kann das übergeordnete Ziel der Schaffung eines einfachen Zugangs für möglichst viele Anwender zu verteilt vorliegenden, heterogenen Daten und Diensten erreicht werden.⁷ Die einzelnen Komponenten einer GDI werden dabei über ein Kommunikationsnetz verknüpft. Je nach Zielgruppe der GDI kann dies das Internet oder auch das Intranet sein.

Die Grundlage einer GDI bilden interoperable Dienste, die über herstellerunabhängige Schnittstellen kommunizieren.⁸ Beispiele für solche Dienste sind der Catalogue Service for the Web (CSW), der Web Map Service (WMS) oder auch der Web Feature Service (WFS), die durch das Open Geospatial Consortium (OGC) [vgl. Abschnitt 1.3.2] entwickelt wurden. Eine kurze Beschreibung der hier genannten Dienste wird in Abschnitt 1.3.4 gegeben.

Über diese interoperablen Dienste erhält der Anwender einen direkten Zugang zu den bei den jeweiligen Datenanbietern vorgehaltenen Geodaten, bestehend aus Geobasis-⁹, Geofach-¹⁰ und Metadaten. Für das Ziel eines transparenten Geoinformationsmarkts sind insbesondere Metadaten von besonderer Bedeutung. Metadaten beschreiben (Geo-)Datensätze („Daten über Daten“). Sie enthalten beispielsweise Angaben zur Qualität der Daten, dem zugrunde liegenden Projektionssystem, dem Originaldatenformat oder zur jeweiligen Kontaktperson bzw. bereitstellenden Organisation. Der Bereitstellung von Metadaten über untereinander kompatible Metadatenkataloge¹¹ kommt (zukünftig) eine bedeutende Rolle innerhalb von GDIs zu. Dabei greifen Metadatenkataloge über standardisierte Katalogdienste [vgl. Abschnitt 1.3.4.1] auf Metadatenbestände zu, die somit recherchierbar werden. Bisher geht der Aufbau von Metadatenkatalogen jedoch recht langsam von statten. Die Leistungen

⁶ FRONEFELD et. al. 2004:14ff.

⁷ FRONEFELD et. al. 2004: 8.

⁸ CRHISTL 2008; http://www.fossgis.de/wiki/index.php/Geoportale_und_Geodateninfrastrukturen (Stand 04.05.08).

⁹ „Geobasisdaten sind grundlegende Geodaten, welche die Landschaft (Topographie), die Grundstücke und Gebäude anwendungsneutral in einem einheitlichen geodätischen Bezugssystem beschreiben“ (IMAGI 2004: 8).

¹⁰ „Geofachdaten sind thematische Daten mit Ortsbezug, der sowohl direkt durch geographische Koordinaten, als auch indirekt, zum Beispiel durch Postleitzahlen oder administrative Einheiten, gegeben sein kann“ (IMAGI 2004: 8).

¹¹ Beispiele für Metadatenkataloge sind z. B. der Geodatenkatalog beim BMI (<http://geoportal.bkg.bund.de>, Stand 24.08.2008) oder der GeoDak am Geozentrum Hannover (<http://geodak.geozentrum-hannover.de>, Stand 24.08.2008).

und die Nutzbarkeit bleiben hinter den gesetzten Erwartungen zurück. Erst durch eine Vernetzung der Kataloge können Mehrwerte erzielt werden.¹²

Auch die Dienste können über Metadatenkataloge recherchiert werden. Die recherchierten Dienste können dann über Benutzerschnittstellen (Clients) durch den Anwender angesprochen werden. Eine direkte Interaktion mit den Geodaten ist somit möglich. Als Start oder Einstiegsseite einer GDI kann ein Geoportal eingerichtet werden. Darunter wird allgemein eine „Website verstanden, die versucht, verschiedene regelmäßig benötigte Dienste (z.B. Routenplanung, Internetbanking, Flugpläne) zu bündeln oder eine Übersicht für den Einstieg in einen Themenkomplex zu schaffen. Aus Sicht der Informatik steht das Bereitstellen von applikationsübergreifenden Leistungen und somit der Integrationsaspekt und nicht die technische Implementierung im Vordergrund.“¹³ „Das ideale Portal eröffnet einen gemeinsamen, personalisierten Zugang zu Daten, Expertisen und Anwendungen“ (Dataquest)¹⁴. Bei der Einstiegsseite eines Geoportals handelt es sich in der Regel um eine HTML-Implementierung, die sich technisch nicht von einer normalen Webseite unterscheidet.¹⁵

Grundsätzlich spielt es keine Rolle, welche Software hinter den Standardschnittstellen steht, solange international anerkannte Normen und Standards im Bereich der Daten, Dienste und Netzwerk-Technologie eingehalten werden [vgl. Abschnitt 1.3]. Erst diese ermöglichen das Zusammenspiel der eigenständigen Komponenten einer GDI und deren Einsatz in verschiedenen GDI-Lösungen. „Der Wert einer GDI ist nicht das fertige Geo-Produkt, sondern die Kombination von Daten und Diensten in Anwendungen.“¹⁶

1.2. Aufbau von Geodateninfrastrukturen – politische und rechtliche Rahmenbedingungen

Die Entwicklung und der Aufbau von GDIs in Deutschland wird in erheblichem Ausmaß von den organisatorischen, politischen und rechtlichen Rahmenbedingungen auf europäischer und Bundesebene beeinflusst.

1.2.1 Geodateninfrastrukturen in Europa

In der Europäischen Union und insbesondere in einigen Mitgliedstaaten zeigen sich vielfältige Bestrebungen zum Aufbau von GDIs. Zunehmend kommt es dabei zu einer Konsolidie-

¹² CHRISTL, A. / EMDE, A. FOSSGIS.

¹³ Beim Begriff des Geoportals handelt es sich noch um einen relativ neuen Begriff, der vielfach unterschiedlich verstanden wird und sich deshalb noch in der Entwicklung befindet. An dieser Stelle wird der an der BGR definierte Arbeitsbegriff „Geoportal“ verwendet.

¹⁴ [http://de.wikipedia.org/wiki/Portal_\(Informatik\)](http://de.wikipedia.org/wiki/Portal_(Informatik)) (Stand: 22.08.2008).

¹⁵ CHRISTL, A. / EMDE, A. FOSSGIS.

¹⁶ STEINMANN, E. S. 1.

ung und Koordination von zuvor isoliert verlaufenden Aktivitäten.¹⁷ Dennoch wird eine europaweite, interoperable Nutzung von Geoinformationen bislang durch eine Vielzahl an Hindernissen, wie beispielsweise Datenlücken, eine fehlende Dokumentation oder inkompatible Geodatenätze und Geodienste, oftmals erschwert bis verhindert.¹⁸ Nach BERNHARD¹⁹ fehlte bisher „ein übergreifendes Rahmenwerk, das für den koordinierten Aufbau einer europäischen GDI (European Spatial Data Infrastructure, ESDI) die notwendigen Standards und Regeln für den interoperablen Austausch von Geoinformationen eindeutig festlegt, sowie geeignete Organisationsstrukturen definiert und benennt.“ Diese Aufgabe übernimmt, wenngleich fokussiert auf umweltpolitische Aspekte, die am 15. April 2007 in Kraft getretene Richtlinie 2007/2/EG des Europäischen Parlaments und des Rates zur Schaffung einer Geodateninfrastruktur in der Europäischen Gemeinschaft (INSPIRE).²⁰ Bis zum 15. Mai 2009 ist die INSPIRE-Richtlinie nationales Recht umzusetzen [vgl. *Abschnitt 1.2.2.1, Geodatenzugangsgesetz (GeoZG)*]. Die sich anschließende schrittweise Implementierung von INSPIRE wird sich vermutlich über den Zeitraum einer Dekade erstrecken.

1.2.1.1 INSPIRE – Ziel einer europäischen GDI

Mit INSPIRE wird das Ziel einer europäischen GDI verfolgt, innerhalb derer Entscheidungsträgern aus Verwaltung und Wirtschaft sowie der Wissenschaft und dem einzelnen Bürger behördliche Geoinformationen aus den Mitgliedstaaten interoperabel und unter einheitlichen Rahmenbedingungen zugänglich gemacht werden sollen. Dabei geht es in erster Linie um bereits existierende, digital vorliegende Geoinformationen, nicht um eine Neuerfassung von Geodaten.

Auch wenn INSPIRE, bedingt durch den Fokus auf umweltpolitische Aspekte sowie dem Ziel der Unterstützung des „policy-making“ auf Ebene der Europäischen Kommission nicht mit einer umfassenden europäischen GDI zu verwechseln ist,²¹ berührt die Richtlinie in Ihrer Regelungswirkung nahezu alle Politikfelder und wird sich auch auf die Wirtschaft auswirken.²² Die 34 Themengruppen, zu denen Geodaten bereitgestellt werden sollen, verdeutlichen, dass die Richtlinie auf einem weitreichenden und querschnittsorientierten Verständnis

¹⁷ BERNHARD 2006: 83.

¹⁸ SEIFERT 2006:3.

¹⁹ BERNHARD 2006: 83.

²⁰ INSPIRE, kurz für „Infrastructure for Spatial Information in Europe“.

²¹ STROBEL 2008: 45.

²² STREUF (2007: 25) verweist auf die explizite Unterstützung von INSPIRE seitens der Geoinformationswirtschaft.

von Umweltpolitik aufbaut.²³ Eine zukünftige Ergänzung um weitere Themenbereiche, wie z. B. Landwirtschaft, Verkehr (Transport) oder Energiepolitik, ist auf jeden Fall denkbar.²⁴

1.2.1.2 Umsetzung der INSPIRE-Richtlinie

Die Umsetzung der INSPIRE-Richtlinie setzt auf Strukturen zur Produktion und Nutzung von Geodaten auf, die in den Mitgliedstaaten bereits etabliert sind oder sich noch im Aufbau befinden.²⁵ Dabei sollen Geodaten, angelehnt an das Subsidiaritätsprinzip, nicht redundant erhoben und ausschließlich dort gepflegt werden, wo dies am effektivsten erfolgen kann.

INSPIRE zielt auf die Entwicklung von technischen und organisatorischen Regeln zur Etablierung einer europäischen GDI ab.²⁶ Dabei sollen die folgenden Kernelemente erarbeitet werden:²⁷

- *Metadaten:*

Bereitstellung harmonisierter Metadaten zur Unterstützung der Recherche und Bewertung von Geodaten und Diensten.

- *Interoperabilität von Geodatenätzen und Diensten:*

Erarbeitung technischer Spezifikationen zur Schaffung von Interoperabilität für die durch INSPIRE adressierten Geodaten und Dienste.

- *Netzwerkdienste:*

Definition interoperabler Dienste für Recherche, Visualisierung, Zugriff und Transformation von Geodaten sowie weitere Dienste, die auf den genannten aufsetzen.

- *Gemeinsame Nutzung von Geodaten:*

Erarbeitung von Regeln zur gemeinsamen Nutzung und Wiederverwendung der in INSPIRE adressierten Geodaten und Geodienste (besonderer Abstimmungsbedarf besteht bei der Lizenzierung nicht frei verfügbarer Geodaten).

- *Koordinierung:*

Erarbeitung von Regeln zur Koordinierung und zum Monitoring der INSPIRE-Umsetzung.

Inhalt und Funktion von Geodaten, Geodiensten und Metadaten sind in der INSPIRE-Richtlinie nicht im Einzelnen festgelegt. Technische, semantische und inhaltliche Details

²³ BMU 2008: 13.

²⁴ SEIFERT 2006: 4.

²⁵ STREUF 2007: 24.

²⁶ SEIFERT 2006: 7.

²⁷ STREUF 2007: 23; BERNARD 2006: 84; ABl. L 108: 6 ff.

werden, basierend auf existierenden Standards und Spezifikationen, innerhalb eines definierten Zeitfensters im Rahmen von sog. Durchführungsbestimmungen entwickelt, die für die EU-Mitgliedstaaten verpflichtend sein werden. Die Erarbeitung dieser auch als „Implementing Rules“ (IR) bezeichneten Bestimmungen erfolgt durch die Europäische Kommission und in enger Abstimmung mit den Mitgliedstaaten, Expertengruppen (sog. ‚Drafting Teams‘) und der Öffentlichkeit auf dem Wege der Komitologie.²⁸

Des Weiteren wird bei den in der INSPIRE-Richtlinie adressierten Geodaten zwischen drei verschiedenen Harmonisierungskategorien (vgl. Richtlinie 2007/2/EG, Annex I – III) unterschieden. Entsprechend der jeweiligen Kategorien werden die Daten hinsichtlich ihres Detaillierungsgrades bzw. ihrer zeitlichen Bereitstellung unterschiedlich behandelt. INSPIRE ebnet somit den Weg zu einer schrittweisen Harmonisierung des europäischen Geodatenbestands.

1.2.2 Geodateninfrastrukturen in Deutschland

Aus deutscher Perspektive bildet INSPIRE das „europäische Dach“ über den im Laufe der vergangenen Jahre aufgebauten sowie derzeit im Aufbau befindlichen nationalen Strukturen,²⁹ allen voran der Geodateninfrastruktur Deutschland (GDI-DE).

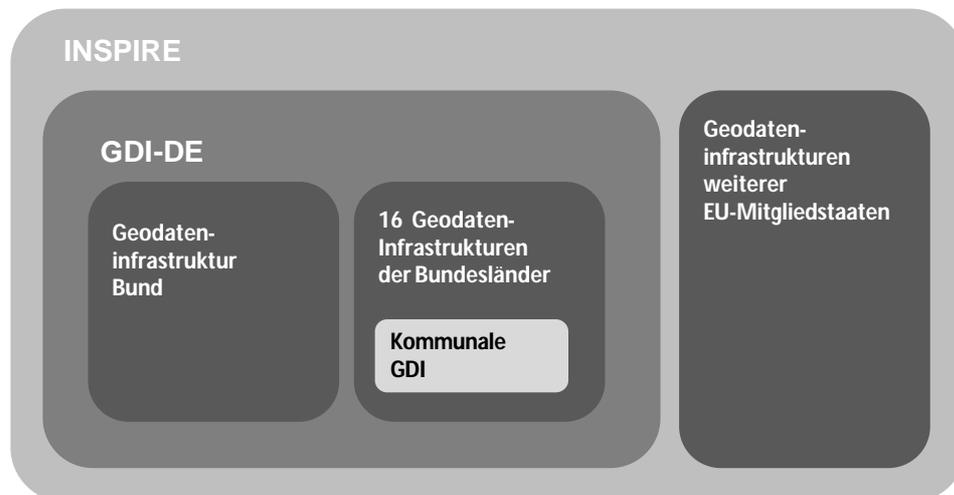


Abb. 4: GDI-Hierarchien in Europa³⁰

Bevor jedoch im Folgenden näher auf den Aufbau der GDI-DE eingegangen wird, soll zunächst ein knapper Überblick über bestehende Regelungen hinsichtlich des Zugangs zu amtlichen Informationen des Bundes im Allgemeinen und zu Geodaten bzw. Geoinformationen im Speziellen gegeben werden.

²⁸ BMU 2008: 14 - Einen guten Überblick zur Vorgehensweise und den Beteiligungsstrukturen bei der Erarbeitung der Durchführungsbestimmungen geben SEIFERT (2006: 8 ff.) und BERNHARD (2006: 85 f.).

²⁹ STREUF 2007: 24.

³⁰ Quelle: GDI-DE Architektur 1.0 2007: 11.

1.2.2.1 *Rechtliche Rahmenbedingungen – ein Überblick*

Das Thema Geoinformation hat in den vergangenen Jahren zunehmend an Bedeutung gewonnen. Auf Seiten der Politik wurde die Notwendigkeit erkannt, rechtliche und technische Regelungen für den Umgang mit Geoinformation zu erarbeiten. Initiiert auf europäischer Ebene existieren auf Bundesebene verschiedene Bestrebungen, die Informationsfreiheit des Einzelnen durch neue Vorschriften zu bestärken. Die im Folgenden kurz vorgestellten Rechtsnormen sowie das hier nur erwähnte Bundesdatenschutzgesetz (BDSG)³¹ bilden die gesetzliche Grundlage für Behörden im Umgang mit öffentlichen Informationen.

Informationsfreiheits- (IFG) und Umweltinformationsgesetz (UIG)

Mit dem Gesetz zur Regelung des Zugangs zu Informationen des Bundes (Informationsfreiheitsgesetz, IFG), das die Verordnung (EG) Nr. 1049/2001 vom 30. Mai 2001 in Deutschland umsetzt, wird fachunspezifisch der freie Zugang der Öffentlichkeit zu amtlichen Informationen des Bundes geregelt. Damit wird eine öffentliche Informations- bzw. Auskunftspflicht geschaffen, was bisher nur möglich war, wenn der bzw. die Auskunftssuchende von der Verwaltungsmaßnahme betroffen war. Der Informationszugang kann jedoch beschränkt oder sogar verweigert werden, wenn eine unzulässige Beeinträchtigung öffentlicher Belange oder des behördlichen Entscheidungsprozesses eintreten sowie wenn der Schutz personenbezogener Daten oder des geistigen Eigentums nicht gewährleistet werden kann.³² Handelt es sich um Informationen aus dem Umweltbereich, kommt das Umweltinformationsgesetz (UIG) des Bundes zur Anwendung, das durch die Richtlinie 2003/4/EG mit Wirkung vom 14. Februar 2005 neuverfasst wurde.

Grundsätzlich erfolgt die Bereitstellung von Geodaten im Rahmen von öffentlichen GDIs im Sinne des IFG und des UIG.³³

Informationsweiterverwendungsgesetz (IWG)

Der durch das IFG und das UIG eröffnete Zugang zu Informationen berechtigt aber nicht automatisch zu deren Weiterverwendung. Hier kommt das Informationsweiterverwendungsgesetz (IWG) zum Tragen, welches am 13. Dezember 2006 in Kraft getreten ist und eine direkte Umsetzung der Richtlinie 2003/98/EG darstellt. Im Hinblick auf die GDI-DE [vgl. Abschnitt 1.2] sind vor allem die hier getroffenen Regelungen zur Gleichbehandlung von Personen, denen amtliche Informationen bereitgestellt wurden, von Bedeutung.³⁴

³¹ Das Bundesdatenschutzgesetz regelt den Umgang mit personenbezogenen Daten.

³² BGR 2008: 7, IFG des Bundes, UIG des Bundes, GDI-DE Architektur 1.0 2007: 26.

³³ GDI-DE Architektur 1.0 2007: 26 f.

³⁴ BGR 2008: 7, IWG, GDI-DE Architektur 1.0 2007: 26.

Geodatenzugangsgesetz (GeoZG)

Mit der INSPIRE-Richtlinie [vgl. Abschnitt 1.2.1.1] wurde die bislang umfassendste Regelung für den Bereich der Geoinformationen geschaffen. Zur Umsetzung in nationales Recht wird in Deutschland derzeit das Geodatenzugangsgesetz (GeoZG) erarbeitet. Es soll den rechtlichen Rahmen für den Zugang zu Geodaten, Geodiensten und Metadaten von den geodatenhaltenden Stellen sowie die Nutzung dieser Daten und Dienste, insbesondere für Maßnahmen herstellen, die Auswirkungen auf die Umwelt haben.³⁵ Um die von INSPIRE geforderte Interoperabilität über die Ebenen hinweg zu gewährleisten, erfolgt die Erarbeitung des GeoZG in enger Abstimmung mit den Ländern und unter Mitwirkung der kommunalen Spitzenverbände.³⁶ Die in den §§ 1, 5 und 10 gegebenen Bezüge zu einer nationalen GDI verdeutlichen, dass das GeoZG als Säule einer solchen zu verstehen ist. Die Aktivitäten zum Aufbau der GDI-DE orientieren sich an den Vorgaben aus der INSPIRE-Richtlinie und ihren Durchführungsbestimmungen.³⁷

1.2.2.2 Aufbau einer Geodateninfrastruktur Deutschland (GDI-DE)

Politische Entscheidungen im Vorfeld

Dem Entschluss zum Aufbau der GDI-DE sind einige politische und administrative Schritte bzw. Entscheidungen vorausgegangen. Bereits am 17. Juni 1998 beschloss das Bundeskabinett die Einrichtung des Interministeriellen Ausschuss für Geoinformationen (IMAGI), um die Koordinierung des Geoinformationswesens in der Bundesverwaltung zu verbessern³⁸ und eine nationale GDI im Zuständigkeitsbereich der Bundesbehörden aufzubauen.³⁹ Mit dem Entschließungsantrag zur „Nutzung von Geoinformationen in der Bundesrepublik Deutschland“ vom 15. Februar 2001 forderte der Deutsche Bundestag die Bundesregierung erstmalig auf, den Aufbau einer nationalen GDI durch gezielte Maßnahmen voranzutreiben.⁴⁰ Gefordert wurde „[...] eine durchgreifende Strategie von Bund und Ländern [...], mit der eine verbesserte Zusammenführung, eine einheitliche Führung sowie eine durchgängige zentrale Abgabe von Geobasisdaten des Bundes und der Länder ermöglicht wird.“⁴¹

Im Oktober 2001 verabschiedete der IMAGI ein Konzept für die Entwicklung und den Aufbau der GDI-DE. Dieses definiert die Nationale Geodatenbasis (NGDB), ein Geoinformationsnetzwerk sowie standardisierte Dienste als Kernbestandteile einer aufzubauenden GDI-DE [vgl. Abb. 3]. Insgesamt sieht das Konzept einen dreistufigen Prozess zum Aufbau einer GDI-DE vor [vgl. Abb. 2]. Dabei erfolgt der „Aufbau der Geodateninfrastruktur in Deutsch-

³⁵ BMU 2008: 2, § 1.

³⁶ BMU 2008: 1.

³⁷ BMU 2008: 15.

³⁸ IMAGI 2004: 12.

³⁹ REINDEL 2005: 1.

⁴⁰ IMAGI 2004: 14.

⁴¹ Deutscher Bundestag 2001: 2, Drucksache 14/5323.

land [...] nicht in einem einzelnen Projekt, sondern kann nur durch eine Vielzahl sich ergänzender Vorhaben umgesetzt werden.“⁴² Der Aufbau einer GDI-DE ist ein langfristiger Prozess, der kontinuierlich an veränderte rechtliche und politische Rahmenbedingungen, sowie an die durch den technischen Fortschritt veränderten Rahmenbedingungen angepasst werden muss.

Aufbau der GDI-DE

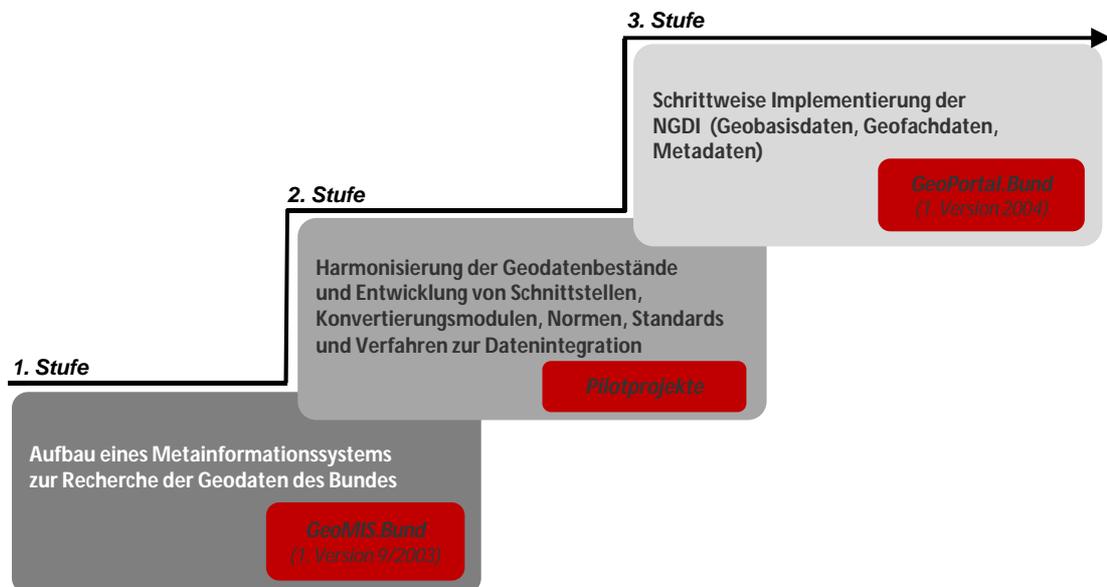


Abb. 5: Stufenweiser Aufbau der GDI-DE⁴³

Politischer Auftrag für den Aufbau der GDI-DE

In verschiedenen Entschlüssen zum Thema Geoinformation⁴⁴ forderte der Deutsche Bundestag eine enge, Ebenen übergreifende Kooperation zwischen Bund, Ländern und Kommunen bei der Bereitstellung von Geodaten. Diesen Forderungen wurde mit dem Beschluss des Chefs des Bundeskanzleramtes und der Chefs der Staats- und Senatskanzleien (CdS) zum gemeinsamen Aufbau der Geodateninfrastruktur Deutschland (GDI-DE) vom 28. November 2003 Rechnung getragen. Die Umsetzung der Beschlussfassung übernimmt die Staatssekretärrunde für eGovernment, da die GDI-DE eine wichtige Säule innerhalb der eGovernment-Initiativen darstellt.⁴⁵ Vorrangiges Ziel beim Aufbau der GDI-DE ist die Einrichtung eines bundesweiten, offenen Geodatennetzwerks, um einen einfachen Zugriff auf Geodaten und Dienste zu ermöglichen und die Bezugs- und Abgabebedingungen für Geodaten zu optimieren.

⁴² GDI-DE Architektur 1.0 2007b: 11.

⁴³ Quelle: IMAGI 2004: 15, verändert.

⁴⁴ Deutscher Bundestag Drucksachen 14/5323 und 15/809.

⁴⁵ IMAGI 2004: 16; GDI-DE Architektur 1.0 2007: 19.b.

Organisationsstruktur der GDI-DE

Für den Aufbau der GDI-DE wurde eine Gremienstruktur mit abgestuften Kompetenzen und Aufgaben ins Leben gerufen [vgl. Abb. 6]. Auf fachpolitischer und konzeptioneller Ebene konstituierte sich am 17. Dezember 2004 das Lenkungs-gremium GDI-DE (LG GDI-DE). Es setzt sich aus Vertretern von Bund⁴⁶, Ländern, Kommunen und den drei kommunalen Spitzenverbänden⁴⁷ zusammen. Es hat die Funktion eines strategischen Entscheidungsgremiums mit folgenden Aufgaben:⁴⁸

- Definition eines abgestimmten Konzepts für den Aufbau einer GDI-DE als Bestandteil einer europäischen GDI,
- Definition von Schlüsseldaten für die NGDB,
- Koordinierung und Mitwirkung bei der Entwicklung, Fortführung und Umsetzung internationaler Normen und Standards,
- Festlegung von Pilotprojekten nach dem Prinzip „Einige für Alle“,
- Sicherstellung des Wissenstransfers und des Austauschs von Verfahrenslösungen über alle Ebenen hinweg sowie
- Finanzierung einer Geschäfts- und Koordinierungsstelle.

Dem LG GDI-DE steht die Kommission für Geoinformationswirtschaft (GIW-Kommission) in beratender Funktion zur Seite. Diese vertritt die Anforderungen der Wirtschaft an Geoinformationen gegenüber der Politik. Der operative Betrieb (fachlich-technischer Ebene) der GIW-Kommission wird von der GIW-Geschäftsstelle (GIW GSt.) geleitet, die bei der Bundesanstalt für Geowissenschaften und Rohstoffe (BGR) in Hannover angesiedelt ist.

Auf fachlich-technischer Ebene bedient sich das LG GDI-DE einer Geschäfts- und Koordinierungsstelle (GKSt. GDI-DE), die beim Bundesamt für Kartographie und Geodäsie in Frankfurt am Main eingerichtet ist. Sie arbeitet in einem organisatorischen Netzwerk und stützt in Zusammenarbeit mit den Daten erhebenden, haltenden und verarbeitenden Stellen (eigentliche Träger der GDI-DE) die Koordinierungstätigkeiten und den Geschäftsbetrieb des LG GDI-DE. Ebenfalls beim BKG angesiedelt und zwecks Vermeidung von Redundanzen und parallelen Strukturen in die GKSt. GDI-DE integriert, ist die Geschäfts- und Koordinierungsstelle des IMAGI (IMAGI GKSt.).

⁴⁶ Der Bund ist durch das BMI, welches den Vorsitz des IMAGI inne hat, und das BMWI, in dessen Zuständigkeitsbereich die GIW-Kommission angesiedelt ist, im LG GDI-DE vertreten.

⁴⁷ Die drei Kommunalen Spitzenverbände sind der Deutsche Städtetag, der Deutsche Landkreistag und der Deutsche Städte- und Gemeindebund.

⁴⁸ GDI-DE Architektur 1.0 2007: 19.

GDI-DE und INSPIRE

Im Sinne von INSPIRE ist die GDI-DE der nationale Ansprechpartner der EU. Über die Durchführungsbestimmungen übt INSPIRE einen großen Einfluss auf die Ausgestaltung der nationalen GDIs aus. Diese sind für die Mitgliedstaaten verpflichtend und müssen bei der Erarbeitung der Architekturkonzepte berücksichtigt werden. In Deutschland ist das LG GDI-DE für die Erarbeitung eines abgestimmten Konzepts für den Aufbau einer GDI-DE (aktuell in der Version 1.0 vom 17. August 2007) als Bestandteil einer europäischen GDI im Sinne von INSPIRE verantwortlich. Als „Legally Mandated Organization“ (LMO)⁴⁹ ist das LG GDI-DE an der Erarbeitung der INSPIRE-Durchführungsbestimmungen beteiligt.

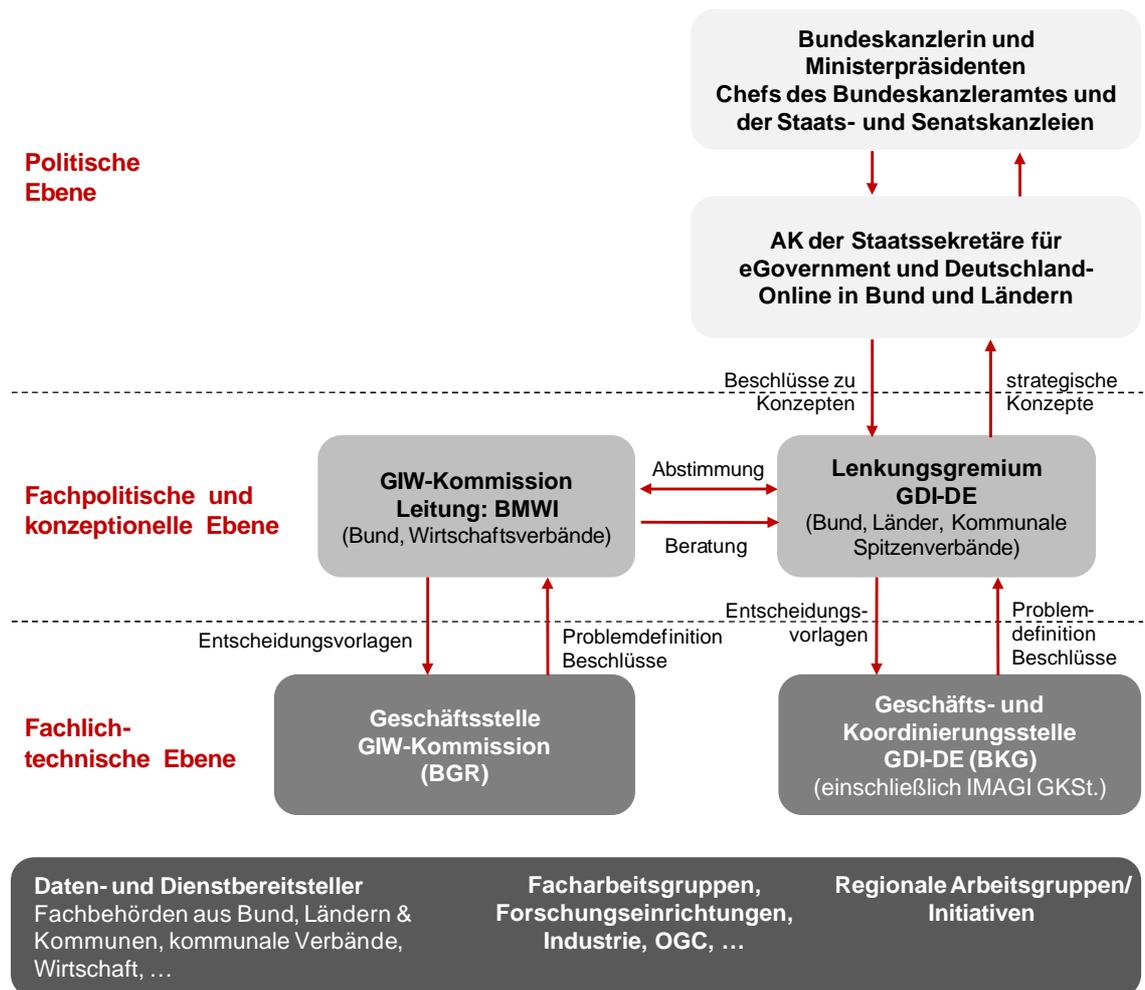


Abb. 6: Aufbau der GDI-DE – Gremienstruktur mit abgestuften Kompetenzen und Aufgaben⁵⁰

⁴⁹ LMO: vgl. erläuternde Fußnote in Abschnitt 3.1.1.1.

⁵⁰ http://www.gdi-de.de/de/organisation/f_organisation.html (Stand: 24.05.2008); GDI-DE Architektur 1.0 2007: 21.

1.3. Interoperabilität durch Nutzung von OpenGIS®

1.3.1 Normen und Standards für Geodaten und Geodienste

Die im Rahmen von GDIs angestrebte organisations- und systemübergreifende Nutzung von räumlichen Informationen erfordert Interoperabilität. Dabei wird unter Interoperabilität die Fähigkeit verstanden, „über verschiedenste Systeme hinweg auf eine Art und Weise Informationen auszutauschen, Programme auszuführen oder Daten zwischen funktionalen Komponenten auszutauschen, ohne dass der Endanwender vertiefte Kenntnisse über die zu Grunde liegenden Mechanismen haben muss.“⁵¹ Dies erfordert die Nutzung abgestimmter und verbindlich festgelegter Normen und Standards, die genau eingehalten werden müssen.

„Normen und Standards sind technische Vorschriften zur Vereinheitlichung von Technikbestandteilen und ihrem Zusammenwirken.“⁵² Entsprechend dem normierenden Gremium werden De-jure- und De-facto-Standards unterschieden. Die sog. *De-jure-Standards* (Normen) werden durch internationale und nationale Normierungsinstitutionen, wie beispielsweise der International Organization for Standardization (ISO), im Rahmen formeller Normungsverfahren festgelegt und veröffentlicht. *De-facto-Standards* werden auch als Industriestandards bezeichnet, da sie von nicht staatlich autorisierten Gremien entwickelt werden. Grundsätzlich muss zwischen De-facto-Standards unterschieden werden, die aus einer Abstimmung von Industriekonsortien, wie z.B. dem Open Geospatial Consortium (OGC) [vgl. *Abschnitt 1.3.2*], hervorgehen und solchen, die sich aufgrund ihrer starken Verbreitung in der Praxis durchsetzen.⁵³

Für eine Standardisierung im Bereich der Geodaten und Geodienste sind das OGC und die ISO mit dem Technical Committee 211, Geographic Information/Geomatics (ISO/TC 211) von zentraler Bedeutung.⁵⁴ Die ISO Normenreihe 19100 behandelt die Standardisierung digitaler geographischer Informationen. Dabei wird zwischen abstrakt-formalen und anwendungsnahen Standards unterschieden. Eine ähnliche Differenzierung findet sich auch beim OGC, welches zwischen abstrakten Spezifikationen (vgl. OpenGIS® Abstract Specifications) und Anwendungsspezifikationen (OpenGIS® Implementation Specifications) unterscheidet. Formale bzw. abstrakte Standards stellen jeweils ein dem aktuellen Stand von Forschung und Technik entsprechendes konzeptionelles Modell zur Modellierung und Verarbeitung raumbezogener Daten zur Verfügung. Auf Basis dieser Modelle können dann

⁵¹ Interoperability is the “capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units” (Source: OGC Abstract Specification Topic 12: Services Derived from ISO 2382-1. In: KLOPFER 2007: 11).

⁵² GREVE / KIEHLE 2006a: 28.

⁵³ GREVE / KIEHLE 2006a: 28.

⁵⁴ Zukünftig werden auch die Festlegungen des European Committee for Standardization (CEN) und Empfehlungen und Vorgaben von INSPIRE von Bedeutung sein.

anwendungsbezogene Spezifikationen derart entwickelt werden, dass sie interoperabel zusammenwirken können.⁵⁵ Dabei reicht es nicht aus, einzelne Schnittstellen zu normieren. Für den Aufbau interoperabler GDIs (oder auch GIS) müssen möglichst viele Kommunikationshindernisse bei der Verbreitung raumbezogener Informationen überwunden werden.

Um ähnliche Aktivitäten der beiden Standardisierungsinstitutionen zu bündeln und keine widersprüchlichen Standards zu erzeugen, wurde eine verstärkte Kooperation (Class A Liaison) vereinbart.⁵⁶ Diese geht so weit, dass einzelne Standards sowohl ISO-Normen als auch OpenGIS® Standards sind. Ein Beispiel ist die OpenGIS Abstract Specification Topic 12: The OpenGIS Service Architecture und die ISO 19119 (Dienste).⁵⁷ Den Hauptunterschied zwischen den von ISO und OGC erarbeiteten Normen und Standards formuliert KIEHLE wie folgt: „Das OGC erarbeitet und veröffentlicht frei verfügbare Standards in einem konsensorientierten Prozess. Die durch ISO/TC211 veröffentlichten Normen haben einen (oftmals langwierigen) Normierungsprozess durchlaufen, welcher der Öffentlichkeit nicht zugänglich ist.“⁵⁸

1.3.2 Das Open Geospatial Consortium (OGC)

Das 1994⁵⁹ aus der Open GRASS Foundation (OGF) hervorgegangene OGC ist ein globales Konsortium mit derzeit über 360 Mitgliedern aus Industrie, klein- und mittelständischen Unternehmen, öffentlicher Verwaltung und Behörden sowie Universitäten. Es verfolgt die Vision, „eine Informationswelt [zu er]schaffen, in der jeder Geoinformationen und Geodienste über Netzwerk-, Applikations- und Plattformgrenzen hinweg nutzen kann.“⁶⁰ Kern der Arbeit des OGC bzw. seiner Mitglieder ist die Entwicklung, Überprüfung, Veröffentlichung und Verbreitung frei verfügbarer OpenGIS® Spezifikationen in drei parallel geführten strukturierten und abgestimmten Programmen:⁶¹

- Im *Specification Development Program* werden in konsensbasierten Prozessen OGC Spezifikationen erarbeitet und verabschiedet (seit 1994).
- Im *Interoperability Program (IP)* werden im Rahmen von Pilotprojekten und Testbeds Prototypen und Spezifikationsentwürfe erarbeitet (seit 1999).
- Das *Outreach and Community Adoption Program (OCAP)* bietet den OGC-Mitgliedern eine Unterstützung bei der Marktentwicklung und dient der Ausbildung und Vernetzung (seit 2002).

⁵⁵ GREVE / KIEHLE 2006b: 29.

⁵⁶ GREVE / KIEHLE 2006a: 28.

⁵⁷ GREVE / KIEHLE 2006b: 29.

⁵⁸ KIEHLE 2006: 52.

⁵⁹ Das OGC wurde 1994 als The Open GIS Consortium gegründet und nennt sich seit 2004 Open Geospatial Consortium (OGC).

⁶⁰ KLOPFER 2007: 3.

⁶¹ KLOPFER 2007: 16; TRAKAS / KLOPFER 2007: 9 f.

Die Implementierungsspezifikationen des OGC definieren Schnittstellen zu Geodiensten, die bestimmte Funktionalitäten systemunabhängig bereitstellen. Dies ermöglicht einen interoperablen Zugriff auf heterogene Datenbestände, ohne dass eine vorherige Konvertierung erforderlich wird. Ein wesentliches Merkmal der OGC Spezifikationen ist deren Offenheit, d.h. sie⁶²

- sind frei und öffentlich verfügbar,
- sind nicht diskriminierend,
- erfordern keine Lizenzgebühren,
- sind anbieterneutral,
- sind datenneutral und
- sind in einem formalen, mitgliederbasierten Konsensprozess verabschiedet worden.

Parallel zu den drei oben kurz vorgestellten Programmen veröffentlicht das OGC ein Reference Model (ORM) [vgl. Abschnitt 1.3.3].

1.3.3 Rahmenarchitektur für Geodienste – das OWS Service Framework (OSF)

Das im Folgenden näher vorgestellte *OWS Service Framework (OSF)* ist Bestandteil des *Reference Model for Open Distributed Processing (RM-ODP)*. Beim RM-ODP handelt es sich um eine ISO-Norm (ISO 10746), die ein Metamodell für die Standardisierung von offenen, verteilten Systemen zum Gegenstand hat und grundlegende Konzepte für die ISO-Normenserie 19100 liefert. Es bildet ebenfalls die Grundlage für das *OGC Reference Model (ORM)*, welches wiederum das Rahmenwerk für alle Aktivitäten des OGC darstellt, das Zusammenwirken der einzelnen Spezifikationen verdeutlicht und kontinuierlich fortgeschrieben wird. Alle OpenGIS® Spezifikationen – sowohl OpenGIS® Abstract Specifications (AS)⁶³ als auch OpenGIS® Implementation Specifications (IS) [vgl. Tab. 1] – werden auf das ORM abgestimmt.

OWS Service Framework (OSF)

Die vom OGC entwickelten OpenGIS® Webservices (OWS) besitzen eine hohe Bedeutung für den Aufbau von GDIs. In einem engen Zusammenhang zu diesen steht das OWS Service Framework (OSF).⁶⁴ Es stellt eine Rahmenarchitektur dar, die Dienste, Schnittstellen und Austauschprotokolle ausweist, die von jeder OpenGIS® konformen Anwendung genutzt werden können. Dabei erfolgt die Kommunikation zwischen den Diensten und Anwendungen über gemeinsame standardisierte Schnittstellen. So entstehen sehr agile Systeme, in

⁶² KLOPFER 2007: 6.

⁶³ Die AS bilden die Grundlage für die Entwicklung der IS. Einen aktuellen Überblick zur den AS findet sich unter <http://www.opengeospatial.org/standards/as> (Stand: 4. 06. 2008).

⁶⁴ KIEHLE 2006: 52.

denen Dienste und Anwendungen verändert, ergänzt, ersetzt oder auch entfernt werden können, ohne andere Komponenten des jeweiligen Systems zu beeinflussen. Das OSF unterscheidet fünf abstrakte Kategorien von Diensten:⁶⁵

- **Application Services:** Sogenannte Anwendungsdienste verbinden den Nutzer (Mensch oder Maschine) je nach Art der Anwendung mit Processing, Data, Registry oder Portrayal Services. Sie sind entweder in eine Benutzeroberfläche eingebunden oder laufen auf in einem Netzwerk eingebundenen Servern.
- **Registry Services:** Registrierungsdienste stellen einen einheitlichen Zugriffspunkt auf Ressourcen bereit. Dabei werden unterschiedliche Typen von Registrierungsdiensten, z.B. Datenkataloge oder Servicekataloge, unterschieden.
- **Data Services:** Der eigentliche Zugriff auf die in Repositories oder Datenbanken vorgehaltenen Ressourcen erfolgt über Datendienste. Sie ermöglichen geometrische und sachdatenbezogene Filteroperationen. Entsprechend den Ausgangsdaten wird zwischen Feature Access Services (FAS) und Coverage Access Services (CAS) unterschieden. Erstere ermöglichen einen Zugriff auf als Simple Features organisierte Vektordaten, letztere machen mehrdimensionale Rasterdaten zugänglich.
- **Portrayal Services:** Die Visualisierung und Präsentation der raumbezogenen Daten erfolgt über Darstellungsdienste. Diese können eng oder lose mit Daten- oder Verarbeitungsdiensten gekoppelt werden.
- **Processing Services:** Mithilfe von Verarbeitungsdiensten können die Daten verarbeitet, d.h. transformiert, kombiniert oder erzeugt werden. Processing Services können lose oder eng mit anderen Diensten wie z.B. Data oder andere Processing Services gekoppelt werden.

Das UML-Komponentendiagramm in Abb. 7 zeigt eine vereinfachte Darstellung der OSF-Rahmenarchitektur. Deutlich wird eine dreistufige Hierarchie mit einer abstrakten Oberklasse, abstrakten Kategorien von Diensten sowie konkreten Diensten. Auf oberster Ebene steht die abstrakte Oberklasse OpenGIS Webservice (OWS) [vgl. Abschnitt 1.3.4]. Für alle OWS wird die GetCapabilities Schnittstelle definiert, deren Aufruf Metadaten über den jeweiligen Service in Form eines standardisierten XML-Dokuments (XML – Extensible Markup Language) zurückliefert. Diese Eigenschaft der Selbstbeschreibung ermöglicht eine dynamische Verkettung verschiedener Dienste (Service Chaining) nach dem Publish-Find-Bind-Paradigma [vgl. Abschnitt 1.3.4].⁶⁶ Die mittlere Darstellungsebene visualisiert die oben beschriebenen fünf abstrakten Kategorien. Für diese werden im Implementierungsmodell konkrete Dienste entwickelt, dargestellt auf der untersten Hierarchieebene.

⁶⁵ OGC 2003: 34 ff.; GREVE / KIEHLE 2006a: 31; KIEHLE 2006: 52.

⁶⁶ KIEHLE 2006: S. 53.

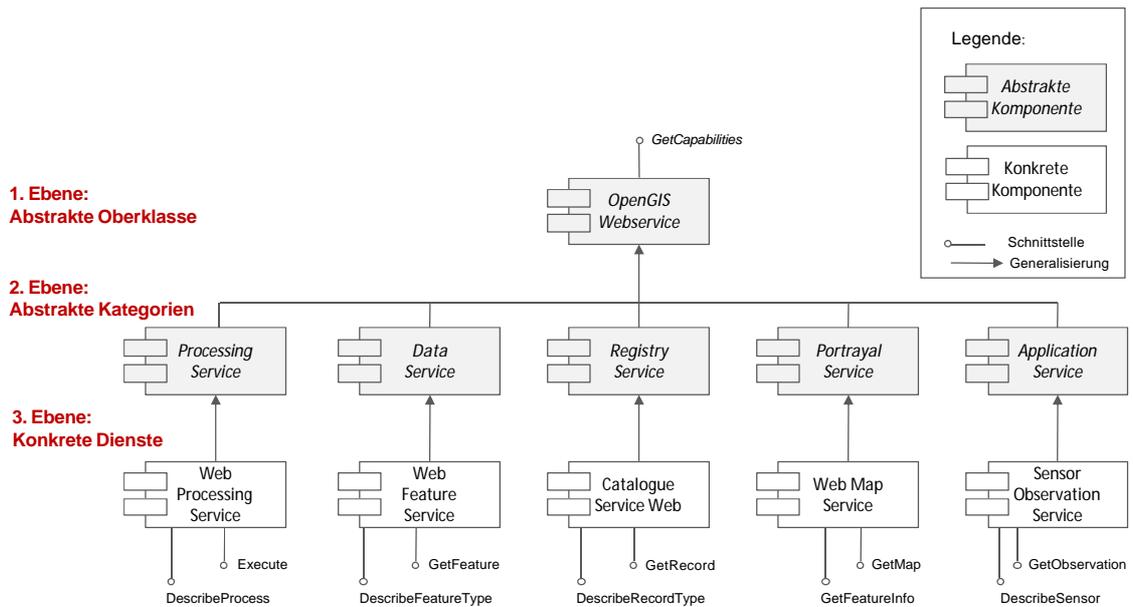


Abb. 7: UML-Komponentendiagramm der OSF-Rahmenarchitektur (vereinfachte Darstellung) ⁶⁷

In Abb. 7 wird dabei jeweils nur ein Beispiel für einen OWS der entsprechenden Kategorie gegeben. Für die beispielhaft benannten konkreten Dienste sind die jeweils dazugehörigen (verpflichtenden) Schnittstellen dargestellt – in ihrer Standardisierung liegt die Lösung der Interoperabilitätsprobleme. Tab. 1 gibt einen Überblick über die obligatorischen Spezifikationen der GDI-DE. Diese sind immer dann anzuwenden, wenn der korrespondierende Dienstyp bzw. das korrespondierende Datenformat bereitgestellt werden soll.

⁶⁷ KIEHLE 2006: 54.

| <i>Diensttyp / Datenformat</i> | <i>GDI-DE obligatorisch</i> |
|--|---|
| Katalogdienst | OpenGIS® Catalogue Service Specification 2.0 ISO Metadata Application Profile, Version 1.0 |
| Vektordatendienst | OpenGIS® WFS 1.0 – Web Feature Service Implementation Specification OpenGIS® WFS 1.1 – Web Feature Service Implementation Specification |
| Rasterdatendienst | OpenGIS® WCS 1.0 – Web Coverage Service Implementation Specification |
| Visualisierungsdienst | WMS-DE Version 1.0 |
| Gazetteer-Dienst | OpenGIS® Gazetteer Service – Application Profile of the Web Feature Service Implementation Specification 0.9.3 |
| Koordinatenreferenzsysteme der Dienste | ETRS89 mit der Abbildung UTM 32 (EPSG-Code: 25832) Geografische Koordinaten im WGS84 (EPSG-Code: 4326) |
| Metadaten und Metadatenformate | ISO 19115:2003 – Metadaten über Geodaten ISO 19119:2003/Cor. 1:2006 – Metadaten über Dienste ISO/TS 19139 (RC2) – Metadaten – XML Schema |
| Vektordatenformate | OpenGIS® Geography Markup Language (GML) Encoding Specification 2.1.2 OpenGIS® Geography Markup Language (GML) Encoding Specification 3.1.1 (GML 3.2, so bald Spezifikation verfügbar) |
| Rasterdatenformate | GeoTIFF (Geo Tagged Image File Format) HDF-EOS (Hierarchical Data Format - Earth Observing System) DTED (Digital Terrain Elevation Data) NITF (National Imagery Transmission) |

Tab. 1: Obligatorische Spezifikationen der GDI-DE⁶⁸

1.3.4 Webbasierte Geodienste

„Als Webservice wird ein Dienst bezeichnet, der über das Internet [HTTP-Transferprotokoll] verfügbar ist, ein standardisiertes XML-Kommunikationssystem verwendet und unabhängig von Betriebssystem und Programmiersprachen ist.“⁶⁹ Im Folgenden werden die für den Aufbau einer einfachen GDI in Abschnitt 3 der vorliegenden Arbeit besonders relevanten webbasierten Geodienste kurz vorgestellt und erläutert.

1.3.4.1 *Catalogue Service for the Web*

Ein Catalogue Service for the Web (CSW) verwaltet und publiziert Metadaten zu den von einem Datenanbieter bereitgestellten Geodaten, Geodiensten und Anwendungen. Die Verwaltung der Metadaten erfolgt konform zu den ISO-Standards 19139 (Metadaten Implementation Specification) und 19115/19119 (Metadaten). Mithilfe eine Client-Anwendung können die Einträge (Metadatenelemente) in verteilt vorliegenden Datenkatalogen durchsucht werden. Sind mehrere Kataloge miteinander verbunden, kann eine Datensuche auch

⁶⁸ Quelle: GDI-DE Architektur 1.0 2007: 38.

⁶⁹ CERAMI, E. 2002, In CHRISTL 2008: 231.

kaskadierend erfolgen. Damit bildet der CSW ein zentrales Element innerhalb einer interoperablen, verteilten Architektur und ermöglicht einen Überblick über die vorhandene Geodaten und Geodienste. Abb. 8 zeigt die Referenzarchitektur für Datenkataloge.

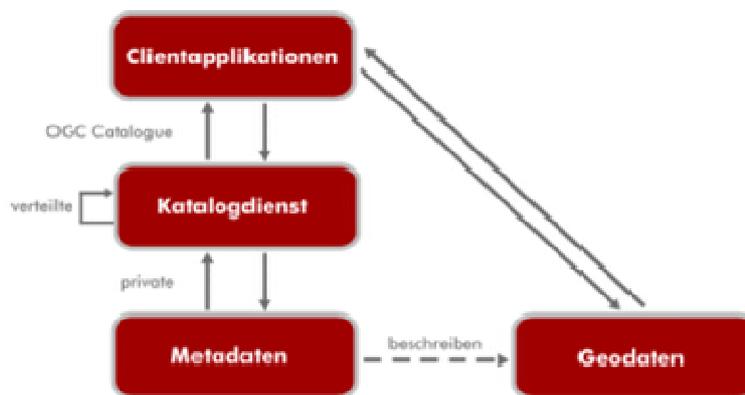


Abb. 8: Referenzarchitektur für Kataloge⁷⁰

Für einen CSW sind verschiedene Schnittstellenbeschreibungen definiert. Die unterstützten Requests sind in Tab. 2 aufgeführt.

| <i>Catalogue Service for the Web</i> | <i>Request</i> | <i>Beschreibung</i> |
|--------------------------------------|------------------------|---|
| <i>Service operations</i> | <i>GetCapabilities</i> | Liefert eine Beschreibung des Leistungsumfangs des Dienstes in Form eines XML-Dokuments zurück. |
| <i>Discovery operations</i> | <i>DescribeRecord</i> | Liefert eine Beschreibung des Metadatenmodells des Katalogs zurück. |
| | <i>GetDomain</i> | Liefert eine Beschreibung der Domain zurück. |
| | <i>GetRecords</i> | Sucht Metadatensätze und zeigt diese an. |
| | <i>GetRecordByID</i> | Sucht Metadatensätze unter Verwendung der ID des Datensatzes und zeigt diese an. |
| <i>Management operations</i> | <i>Transaction</i> | Schnittstelle zur Erzeugung, Modifizierung und Entfernung von Metadatensätzen. |
| | <i>Harvest</i> | Aufnahme und Änderung von Datensätzen durch Referenzierung anderer Datenquellen. |

Tab. 2: Unterstützte Operationen des Catalogue Service for the Web (CSW)

⁷⁰ Quelle: GDI-DE Architektur 1.0 2007: 48

1.3.4.2 Web Map Service

Ein Web Map Service dient der dynamischen Visualisierung von online verfügbaren Vektor- und Rasterdaten. Diese werden durch den WMS in Form einfacher Raster-Bildformate (z.B. .png, .gif oder .jpeg) oder Vektorgraphiken (z.B. SVG) zurück gegeben. Für die Darstellung des jeweiligen Kartenbildes genügt ein einfacher Internetbrowser. Dabei kann ein WMS direkt über eine URL, an die weitere Request-Parameter angehängt sind oder auch über einen in eine HTML-Seite eingebundenen Map Viewer angesprochen werden. Darüber hinaus ist es auch möglich, einen WMS über eine entsprechende WMS-Schnittstelle in eine GIS-Software einzubinden. Diese Möglichkeit wird im Rahmen dieser Arbeit z.B. zur Überprüfung der OGC-Konformität genutzt.

| <i>Web Map Service</i> | <i>Request</i> | <i>Beschreibung</i> |
|--|-----------------------------|--|
| Basic (Ein WMS muss mindestens die Requests GetCapabilities und GetMap beantworten.) | <i>GetCapabilities</i> | Liefert eine Beschreibung des Leistungsumfangs des Dienstes (Metadaten zu verfügbaren Layern, Projektionssystemen, Koordinatenausschnitt etc.) in Form eines XML-Dokuments zurück. |
| | <i>GetMap</i> | Liefert eine durch den Nutzer zusammengestellte dynamisch erzeugte Karte zurück (Layer, Kartenausschnitt, Ausgabeformat etc.). |
| Queryable | <i>GetFeatureInfo</i> | Liefert Informationen (Sachdaten) zu einzelnen Objekten (Features) zurück. |
| SLD | <i>DescribeLayer</i> | Liefert Informationen zu den Daten(quellen), auf denen ein oder mehrere Layer basieren. |
| | <i>GetLegendGraphic</i> | Liefert eine Legendgrafik für den entsprechenden Layer zurück. |
| | <i>GetStyles, PutStyles</i> | Durch die Verwendung des Request-Parameters STYLE können die Layer eines WMS in unterschiedlicher Darstellung angefordert werden. |

Tab. 3: Unterstützte Operationen des Web Map Services (WMS)⁷¹

Für einen WMS sind verschiedene Requests definiert. Wie in Abb. 7 dargestellt, unterstützt auch der WMS die GetCapabilities-Schnittstelle, die allen OWS gemeinsam ist. Je nach unterstützten Operationen wird zwischen einem Static bzw. Basic WMS, einem Queryable WMS sowie einem Styled Layer Descriptor WMS (SLD-WMS) unterschieden. Letzterer ermöglicht eine benutzerdefinierte Darstellung der über einen WMS angeforderten Daten, setzt aber eine Bindung des WMS an einen WFS [vgl. Abschnitt 1.3.4.3] voraus. Tab. 3 stellt die verschiedenen von WMS unterstützten Requests überblicksartig zusammen und

⁷¹ IMAGI 2006: 20; OGC 2007; CHRISTL 2008: 238 ff.;

liefert jeweils eine Kurzbeschreibung. Auf den Aufbau verschiedener Requests unter Angabe definierter Request-Parameter wird im Praxisteil der vorliegenden Arbeit näher eingegangen.

1.3.4.3 Web Feature Service

Der Web Feature Service (WFS) ist ebenfalls eine OGC Implementation Specification (derzeit 1.1), welche einen Zugriff auf online verfügbare Vektordaten ermöglicht. Je nach Ausprägung wird zwischen einem Basic WFS und einem Transaktional WFS (WFS-T) unterschieden. Ersterer gewährt einen rein lesenden Datenzugriff während letzterer auch einen schreibenden Zugriff erlaubt, d.h. wenn der Server dies zulässt, können Vektorgeometrien erzeugt, aktualisiert und gelöscht werden. Wie auch der WMS wird ein WFS über eine URL angesprochen, an die weitere Request-Parameter angehängt werden. Anstatt von Kartenbildern werden durch einen WFS die Geodaten selbst in Form einer XML-Datei, genauer gesagt im GML-Format, ausgeliefert. Die Weiterverarbeitung der Ergebnis-Daten erfolgt durch den Clienten. Erst dann werden die Daten für den Nutzer interpretierbar. Der WFS bietet gegenüber einem WMS mehr Flexibilität, da die Daten weiterverarbeitet und ausgewertet werden können.⁷²

| <i>Web Feature Service</i> | <i>Request</i> | <i>Beschreibung</i> |
|----------------------------|---------------------------|---|
| Basic | <i>GetCapabilities</i> | Liefert eine Beschreibung des Leistungsumfangs des Dienstes in Form eines XML-Dokuments zurück. |
| | <i>DescribeFeaturType</i> | Liefert eine Beschreibung der Struktur der einzelnen Feature Types zurück. |
| | <i>GetFeature</i> | Liefert die eigentlichen Feature Instanzen (Vektordaten) im GML-Format zurück. |
| Transactional | <i>Transaction</i> | Ermöglicht das Erstellen, Aktualisieren und Löschen von einzelnen Features. |
| | <i>LockFeature</i> | Belegt einzelne Feature Instanzen mit einer Veränderungssperre (Vermeidung von Dateninkonsistenzen durch parallele Zugriffe). |
| | <i>GetFeatureWithLock</i> | |

Tab. 4: Unterstützte Operationen des Web Feature Services (WFS)⁷³

⁷² CHRISTL 2008: 238 f.

⁷³ IMAGI 2006: 21; CHRISTL 2008: 253.

1.3.5 Bereitstellung und Bezug von Geodaten

Die OpenGIS® Service Architektur folgt dem in *Publish-Find-Bind-Paradigma*, welches in Abb. 9 dargestellt ist. Als zentrale Akteure (Rollen) lassen sich Anbieter, Nutzer und Broker von Diensten identifizieren.

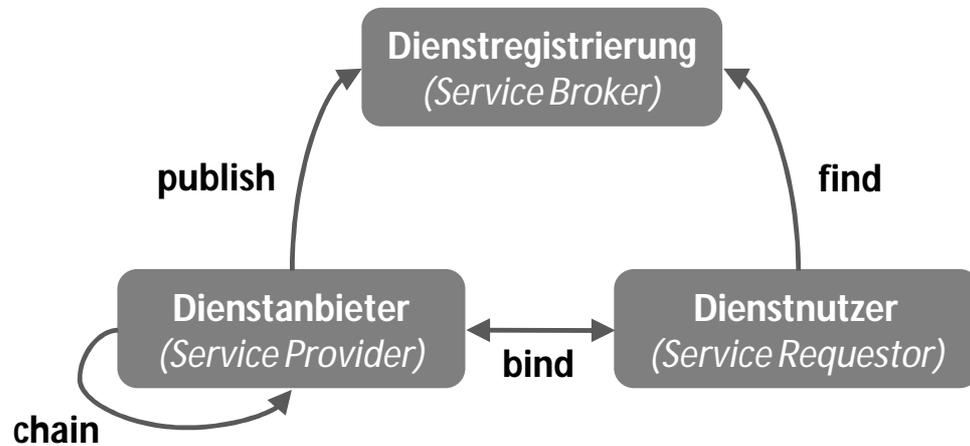


Abb. 9: Das Publish-Find-Bind-Paradigma der OpenGIS® Service Architektur⁷⁴

Die Metadaten eines Dienstes werden durch den jeweiligen Anbieter mithilfe eines sogenannten Brokers registriert und veröffentlicht (*publish*). Hierfür hat das OGC die Catalogue Web Specification (CWS) definiert [vgl. *Abschnitt 1.3.4.1*]. Jetzt können die Nutzer den Broker nutzen und eine Metadatensuche durchführen, um die benötigten Dienste zu finden (*find*). Wurden die gewünschten Dienste gefunden, verbinden sich die Nutzer mit den Anbietern (*bind*), die dann den gewünschten Dienst zu Verfügung stellen.⁷⁵ Die hier beschriebene Form der Metadatenregistrierung und -veröffentlichung ist eine der grundlegenden Ideen serviceorientierter Architekturen.⁷⁶ Auf diese Art und Weise können bedarfsorientiert mehrere Dienste lose mit einander verknüpft werden, ein Prozess, der auch als *Service Chaining* bezeichnet wird.

⁷⁴ nach OGC 2003: 31 ff., angepasst.

⁷⁵ OGC 2003: 30 ff.

⁷⁶ KIEHLE 2006: 53.

2 Einsatz von Open-Source und Freier Software in Geodateninfrastrukturen

2.1. Grundlagen: Open Source und Freie Software

Zentraler Gegenstand dieses Unterabschnitts ist ‚Software‘, genauer gesagt Open Source und Freie Software. Bei Software handelt es sich um eine Wissens-Ressource, die „beliebig oft verlustfrei dupliziert werden [kann]. Jede Kopie einer Software ist immer identisch. Das ‚Rohmaterial‘, der Quelltext, ist unbegrenzt haltbar. [Insofern nutzt] Software [...] sich nicht ab, rostet nicht, zerbricht nicht und hört nie auf zu funktionieren.“⁷⁷ Wird sie weitergegeben, entstehen dem Anbieter keine Verluste, denn die Kopiervorlage geht nicht verloren. Bedingt durch diese Eigenschaften tendiert der Stückpreis von Software gegen Null, es sei denn, es kommt zu einer künstlichen Verknappung der Ressource Software durch die Anwendung proprietärer Lizenzen.⁷⁸

2.1.1 Begrifflichkeiten und Konzepte

Der Eigenschaft von Software als Wissens-Ressource werden insbesondere die beiden Kategorien ‚Open Source‘ und ‚Freie Software‘ gerecht. Sie werden im Folgenden kurz vorgestellt und mit proprietärer und kommerzieller Software verglichen. Abschließend erfolgt eine Einordnung der vorgestellten Begriffe in bestehende Kategorien freier und unfreier Software.⁷⁹

2.1.1.1 Open Source und Freie Software

Freie Software (FS)

Das Konzept der *Freien Software (FS)* geht auf Richard Stallmann zurück, der 1985 die Free Software Foundation (FSF) gründete. Zentrale Aufgabe der Stiftung ist die Förderung und Entwicklung von Freier Software, wobei ein Schwerpunkt auf den rechtlichen und strukturellen Belangen der Freie-Software-Gemeinschaft liegt.⁸⁰ Software wird als frei bezeichnet, wenn sie dem Nutzer die Freiheit gibt, diese „zu (...)nutzen, zu kopieren, sie zu vertreiben, zu studieren, zu verändern und zu verbessern“⁸¹. Damit eine Software gemäß der

⁷⁷ CHRISTL 2008: 2.

⁷⁸ CHRISTL 2008: 2 f.

⁷⁹ Ein Überblick über sämtliche Softwarekategorien kann im Rahmen der vorliegenden Arbeit nicht gegeben werden. Dazu sei auf GWOSDZ (2006).

⁸⁰ http://de.wikipedia.org/wiki/Free_Software_Foundation (Stand 24.04.2008).

⁸¹ <http://www.gnu.org/philosophy/free-sw.de.html> (Stand 24.04.2008).

Definition der FSF „frei“ ist, muss ihre Lizenz die folgenden vier Freiheiten gewährleisten:⁸²

- Freiheit (0): Nutzung der Software zu jedem Zweck,
- Freiheit (1): Studium der Funktionsweise der Software und deren Anpassung an die eigenen Bedürfnisse,
- Freiheit (2): Verbreitung der Software durch Kopien,
- Freiheit (3): Verbesserung der Software und zur Weitergabe der Änderungen zum Nutzen der Allgemeinheit (3).

Aus dieser Definition folgt, dass ein offener und ohne jegliche Restriktionen zugänglicher Quellcode (Source Code) eine grundlegende Bedingung Freier Software ist. Nur so können die erste und dritte Freiheit auch genutzt werden.⁸³

Der Begriff der Freien Software kann insbesondere im englischsprachigen Raum leicht missverstanden werden. „Frei“ (bzw. „free“) ist nicht per se mit „kostenlos“ oder „unkommerziell“ gleichzusetzen. Um einen populären Vergleich zu wählen, kann auch von Freiheit im Sinne von „freier Rede“ jedoch nicht im Sinne von „Freibier“ gesprochen werden.⁸⁴ Auch wenn Freie Software dem Nutzer vielfach tatsächlich kostenlos bereitgestellt wird, muss ein freies Programm auch „für den kommerziellen Gebrauch, die kommerzielle Entwicklung und für die kommerzielle Verteilung zur Verfügung stehen“⁸⁵ können.

Open Source Software (OSS)

Vor diesem Hintergrund wurde 1989 auf dem Gründungstreffen der Non-Profit-Organisation *Open Source Initiative (OSI)* der Begriff *Open Source (OSS)* (Christine Peterson, Foresight Institute)⁸⁶ geprägt. Damit sollte die Mehrdeutigkeit des Terminus Freie Software ausgeräumt werden. Der neue Begriff Open Source rückte anfänglich nur den Quellcode in den Mittelpunkt und legt explizit fest, dass dieser frei zugänglich sein muss.⁸⁷ „Mit dem Begriff soll(..) ein Symbol, ein Dach, ein gemeinsamer Nenner für alles dargestellt werden, was Freie Software charakterisiert.“⁸⁸ In der Argumentation werden eher technische Aspekte wie Nützlichkeit, Zuverlässigkeit und Effizienz der Software hervorgehoben – im Gegensatz zu den Verfechtern der Freien Software um R. Stallmann, die soziale und politische Gesichtspunkte von Software betonen. Ziel war es, auch auf Leitungsebene

⁸² u. a. CCGIS / terrestris 2004: 11 f.; MÖLLER 2005: 62; GRASSMUCK 2004: 233.

⁸³ Eine umfassende Definition Freier Software liefert:
<http://www.gnu.org/philosophy/free-sw.de.html> (Stand: 28.04.2008).

⁸⁴ u. a. GRASSMUCK 2004, S. 230; CCGIS / terrestris 2004, S. 11.

⁸⁵ <http://www.gnu.org/philosophy/free-sw.de.html> (Stand: 28.04.2008).

⁸⁶ Als Urheber des neuen Begriffs Open Source wird auch Eric S. Raymond, einer der Mitbegründer der OSI, genannt (http://de.wikipedia.org/wiki/Freie_Software#Abgrenzung_zu_Open_Source (Stand: 28.04.2008)).

⁸⁷ CCGIS / terrestris 2004, S. 11.

⁸⁸ BEHNCKE 2006, S. 26.

und auf Aktionärsversammlungen größerer Unternehmen ein Interesse an Produkten unter dem Warenzeichen „*Open Source*“ zu schaffen.⁸⁹

Tatsächlich hat der Begriff Open Source auf Seiten vieler Unternehmen Aufmerksamkeit erregt. Gleichzeitig hat sich jedoch auch gezeigt, dass ein offengelegter Quellcode noch kein Garant für die vier oben genannten Freiheiten Freier Software ist. So wurde der Begriff Open Source von einigen Unternehmen zu Marketingzwecken genutzt. Nach Ansicht GRASSMUCKS⁹⁰ ist der „Versuch der OSI, das Warenzeichen „Open Source“ als Gütesiegel für Software durchzusetzen, deren Lizenz der Open-Source-Definition (OSI) genügt, (...) gescheitert. Heute führen viele Produkte das Label, die keine Modifikationsfreiheit gewähren – und genau sie ist der Sinn der Quelloffenheit.“

Die OSI setzt sich für die Organisation und Verbreitung der *Open-Source-Definition (OSD)* als Standard für frei verfügbare Software ein. Diese geht auf die Debian Free Software Guidelines von Bruce Perens zurück. Über den freien Zugang zum Quellcode hinaus garantieren Softwareprodukte, deren Lizenz unter die *Open-Source-Definition (OSD)* der OSI fallen, im Wesentlichen folgende Aspekte (zusammengefasst):⁹¹

- Die Lizenz darf Niemanden beim Verkauf oder der Weitergabe der Software einschränken.
- Der Quellcode eines Programms muss in einer für den Menschen lesbaren Form zur Verfügung gestellt werden.
- Die Lizenz muss Programmänderungen und -weiterentwicklungen sowie deren Verbreitung unter den Lizenzbedingungen der Originalsoftware zulassen.
- Die Lizenz darf keine Personen oder Gruppen diskriminieren oder Anwendungsbereiche der Software einschränken.
- Für Open-Source-Software sind Nutzungsbeschränkungen nicht zulässig.

In der Literatur werden die Begriffe *Open Source und Freie Software (OS/FS)* häufig synonym verwendet. Dieser Handhabung folgt auch die vorliegende Arbeit, denn solange Open Source Produkte unter die OSD fallen, sind sie Produkten, die als Freie Software unter der GPL Lizenz lizenziert worden sind, recht ähnlich. MÖLLER formuliert es so: „... die meisten Software-Lizenzen [fallen] unter beide Definitionen oder keine von beiden.“⁹²

⁸⁹ GRASSMUCK 2004, S. 230.

⁹⁰ GRASSMUCK 2004, S. 231.

⁹¹ <http://opensource.org/osi3.0/node/4>; <http://www.opensource.org/docs/definition.php>;
<http://debiananwenderhandbuch.de/freiesoftware.html#osid>; (Stand: 28.04.2008).

⁹² MÖLLER 2005: 61.

2.1.1.2 *Kommerzielle und proprietäre Software*

In der Diskussion um OS/FS werden verschiedene Kategorien von Software angesprochen. Als Gegenpart zu OS/FS werden häufig kommerzielle oder proprietäre Software bezeichnet. Dabei ist zu beachten, dass die beiden Begriffe nicht synonym verwendet werden können.

Proprietäre Software, auch als herstellereigene Software bezeichnet, stellt das Gegenteil von Freier oder unter die OSD fallende Open-Source Software dar. Ihre Vervielfältigung, Weiterverbreitung und Modifizierung ist untersagt oder erfordert eine ausdrückliche Erlaubnis. Letztere ist allerdings oftmals so einschränkend, dass sie eine effektive Veränderung oder Verbreitung der Software nicht zulässt. Die Nutzung proprietärer Software erfordert den Erwerb einer Lizenz mit den damit einhergehenden Nutzungsbedingungen. Das Urheberrecht an der Software verbleibt beim Hersteller.⁹³

Kommerzielle Software wird mit dem Ziel entwickelt, einen Profit zu erwirtschaften. Fast jede kommerzielle Software ist daher auch proprietär. Dennoch sind diese beiden Software-Kategorien nicht in jedem Fall gleichzusetzen, denn es gibt sowohl kommerzielle Software, die gleichzeitig auch OS/FS ist, als auch nicht-kommerzielle proprietäre Software.⁹⁴

2.1.1.3 *Kategorien von Software – Zusammenfassung*

Nachfolgende Tabelle fasst die grundlegenden Unterschiede von OS/FS sowie kommerzieller und proprietärer Software zusammen. Deutlich werden die jeweiligen Anforderungen an die Benutzung sowie die bestehenden Möglichkeiten der Modifikation. In der tabellarischen Darstellung sind weitere Softwarekategorien aufgeführt, die im Zusammenhang mit der Diskussion von Software-Kategorien immer wieder auftauchen. In dieser Arbeit bilden sie aber lediglich den Rahmen für die Einordnung der vier diskutierten Software-Kategorien, d.h. sie werden erwähnt, aber nicht detaillierter vorgestellt.

⁹³ <http://www.gnu.org/philosophy/categories.de> (Stand: 28.04.2008).

⁹⁴ <http://www.gnu.org/philosophy/categories.de> (Stand: 28.04.2008).

| <i>Software-Kategorien</i> | <i>kostenlos¹</i> | <i>Nutzung²</i> | <i>Weitergabe³</i> | <i>Source-Code offen⁴</i> | <i>Modifizierbar⁵</i> | <i>Kommerzialisierbar⁶</i> |
|--|------------------------------|----------------------------|-------------------------------|--------------------------------------|----------------------------------|---------------------------------------|
| <i>Public Domain</i> | ja | uneingeschränkt | ja | ja | ja | ja |
| <i>Freie Software (FS)⁷</i> | ja ⁸ | uneingeschränkt | ja | ja | ja | nein |
| <i>Open Source (OS)</i> | ja ⁸ | uneingeschränkt | ja | ja | ja | nein |
| <i>Trialware</i> | ja | Zeit ⁹ | nein | nein | nein | nein |
| <i>Shareware</i> | ja ¹⁰ | Funktion ⁹ | nein | nein | nein | nein |
| <i>Freeware</i> | ja | uneingeschränkt | ja | nein | nein | nein |
| <i>Abandonware</i> | ja | uneingeschränkt | ja | nein | nein | nein |
| <i>Kommerzielle Software</i> | nein | Lizenz | nein | nein | nein | nein |
| <i>Proprietäre Software</i> | nein | Lizenz | nein | nein | nein | nein |

¹ Mit kostenlos ist hier nur der Erwerb (Lizenz) der Software gemeint, nicht die Distribution.

² Betrachtung von eventuellen Nutzungsbeschränkungen, wie z.B. Funktionsumfang oder Nutzungsdauer.

³ Ist die Weitergabe des Softwarepaket erlaubt?

⁴ Ist der Source-Code einsehbar?

⁵ Darf der Source-Code, falls offen, auch modifiziert werden?

⁶ Darf das Produkt unter eigener Lizenz verkauft werden (nach der Modifizierung)?

⁷ FS im Sinne der Free Software Foundation [vgl. Abschnitt 2.1.1.1].

⁸ Einschränkung: OS/FS ist meistens kostenlos, aber nicht zwingend.

⁹ Die Nutzung kann sowohl zeitlich als auch von der Funktion her limitiert sein.

¹⁰ Bei längerer Nutzung erwartet der Hersteller, dass der Benutzer das Produkt freiwillig lizenziert.

Tab. 5: Kategorien von Software⁹⁵

2.1.2 Lizenzmodelle von OS/FS

Lizenzen haben insbesondere für OS/FS eine besondere Bedeutung. Allgemein beinhalten sie „die vertraglichen Rechte und Pflichten, unter denen die Urheberrechtsinhaber ihre Werke veröffentlichen.“⁹⁶ Dabei nehmen OS/FS-Lizenzmodelle Urheberrechte an einer Software gezielt in Anspruch, „um den offenen, kooperativen Prozess der Erstellung und Weiterentwicklung abzusichern.“⁹⁷ Aufgrund der Vielzahl an verschiedenen OS/FS-Lizenzen werden im Folgenden die von REITER⁹⁸ skizzierten Lizenz-Kategorien vorgestellt, um einen Überblick zu geben.

⁹⁵ Quelle: GWODZ 2006: 15.

⁹⁶ GRASMUCK 2004: 275.

⁹⁷ GRASMUCK 2004: 275.

⁹⁸ REITER 2004: 86 f.

Stark schützend – z.B. GNU GPL

Die GNU General Public License (GNU GPL) ist die Lizenz, die den stärksten Schutz für OS/FS-Projekte bietet. Um zu gewährleisten, dass Freie Software auch zukünftig frei bleibt, unterliegt sie den auf Seite 27 f. genannten vier Freiheiten. Die Schutzwirkung der GNU GPL wird dadurch erzielt, dass eine veränderte Software nur mit offenem Source Code und unter gleicher Lizenz weitergegeben werden darf.

Schwach schützend – z.B. GNU LGPL

Einen schwächeren Schutz bietet die GNU Lesser General Public License (GNU LGPL). In ihrer Grundintention entspricht sie der GNU GPL und schreibt ebenfalls die dort definierten Freiheiten fest. Sie ermöglicht jedoch die Verwendung von unter der GNU LGPL lizensierter Software in proprietären Softwareprojekten. Bei einer Weitergabe muss lediglich der GNU LGPL Source Code frei bleiben.⁹⁹

Sowohl die GNU GPL als auch die GNU LGPL unterliegen dem "Copyleft". D.h. das amerikanische Copyright wird geschickt dazu verwendet, die mit einer Software verbundenen Freiheitsrechte zu schützen.

"To copyleft a program, we first state that it is copyrighted; then we add distribution terms, which are a legal instrument that gives everyone the rights to use, modify, and redistribute the program's code or any program derived from it but only if the distribution terms are unchanged. Thus, the code and the freedoms become legally inseparable."¹⁰⁰

Nicht schützend – Software im Stil modifizierter BSD-Lizenzen (z.B. X11)

Auch Software im Stil modifizierter BSD-Lizenzen (Berkeley Software Distribution) weist Ähnlichkeiten mit der GNU GPL auf. Ihre Lizenzen sind jedoch liberaler. Sie erlauben die Software zu kopieren, zu verändern und zu verbreiten, enthalten aber kein schützendes Copyleft. Lediglich der Copyright-Vermerk darf nicht entfernt werden.¹⁰¹

Unausgeglichen oder inkompatibel zur GPL

In diese Kategorie stuft REITER¹⁰² alle Freie Software Lizenzen ein, die nicht zur GNU GPL kompatibel sind.

⁹⁹ GRASSMUCK 2004: 289 ff.

¹⁰⁰ <http://www.gnu.org/copyleft/> (Stand: 03.07.2008).

¹⁰¹ GRASSMUCK 2004: 279 ff.

¹⁰² REITER 2004: 86.

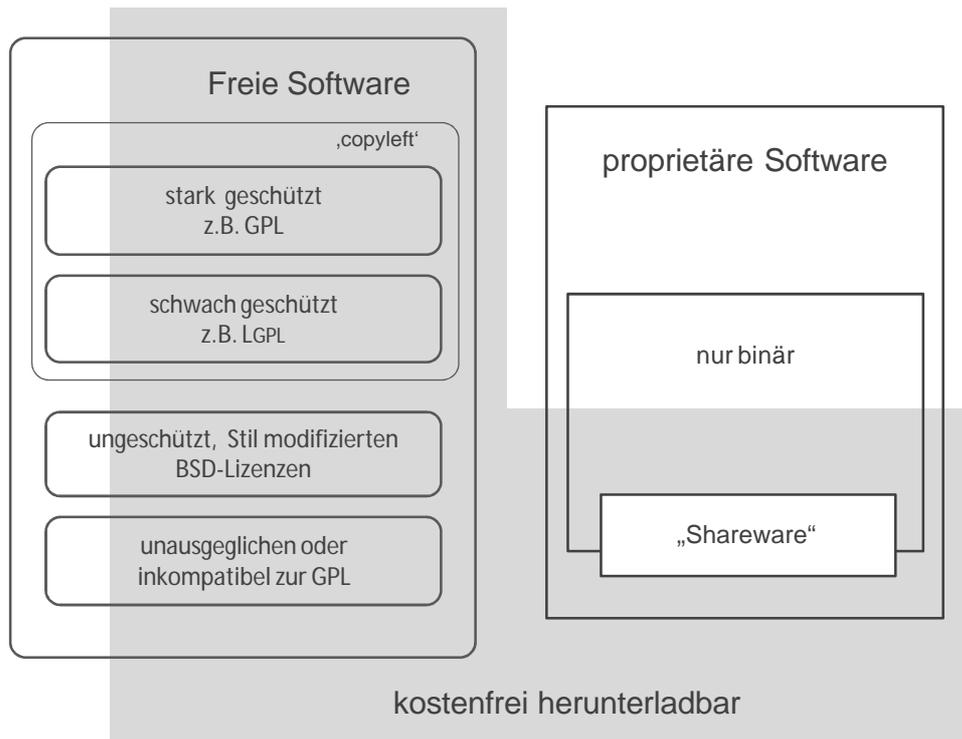


Abb. 10: Lizenzkategorien¹⁰³

Eine Richtlinie zur Bewertung von Lizenzen ist die Open Source Definition (OSD) [vgl. S. 28 f.]. Zu den von der OSD zertifizierten Lizenzen gehören u.a. die oben genannten Lizenzen GNU GPL und GNU LGPL sowie die BSD-Lizenz.

Tab. 6 zeigt die Lizenzen der im praktischen Teil dieser Arbeit eingesetzten Software:

| <i>Software</i> | – <i>Lizenz</i> |
|-----------------|--------------------------------|
| PostGIS | – GNU GPL |
| Apache | – Apache-Lizenz ¹⁰⁴ |
| UMN MapServer | – eigene Lizenz ¹⁰⁵ |
| OpenLayers | – BSD Lizenz |

Tab. 6: Lizenzen der verwendeten Software

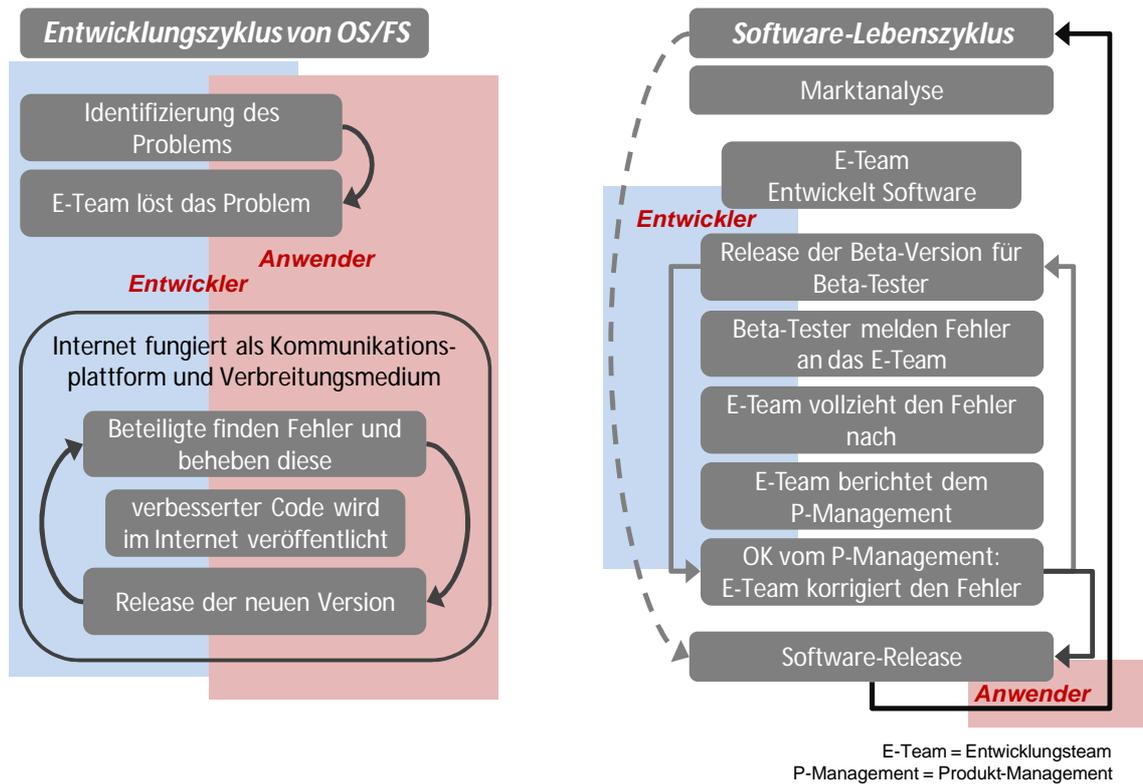
2.1.3 Software-Projekte: Entwicklungszyklus vs. Lebenszyklus

Lizenzen beschreiben insbesondere die Bedingungen der Redistribution einer Software. Die besondere Dynamik vieler OS/FS-Projekte hängt jedoch mit ihrer Organisationsform zusammen. Abb. 11 stellt den Entwicklungszyklus von OS/FS-Projekten des Lebenszyklusmodells proprietärer Softwareprojekte gegenüber.

¹⁰³ REITER 2004: 86 (verändert).

¹⁰⁴ Die Apache-Lizenz ist eine von der BSD abgeleitete Lizenz (Grassmuck 2004: 281). Sie wurde mittlerweile von der FSF anerkannt und ist mit der GNU GPL v3 kompatibel (www.gnu.org/licenses/license-list.html, Stand: 03.06.2008).

¹⁰⁵ Der UMN MapServer wird als OS-Projekt entwickelt und ist als Freie Software mit eigener Lizenz lizenziert.

Abb. 11: Entwicklungszyklus vs. Lebenszyklus¹⁰⁶

Entwicklungszyklus von OS/FS-Projekten

Am Anfang eines jeden OS/FS-Projektes stehen ein Problem und das Interesse eines Entwicklers an dessen Lösung. In der Annahme, dass auch andere Entwickler betroffen und an der Problemlösung interessiert sind, werden OS/FS-Projekte frühzeitig im Internet veröffentlicht. Meist bildet sich bereits kurz nach der Erstveröffentlichung eine Gruppe von Entwicklern, die bereit ist, als „Maintainer“ (Projektleiter) eine koordinierende Mitverantwortung für das Projekt bzw. für bestimmte funktionale Einheiten (Module) zu übernehmen. Bei jedem größeren OS/FS-Projekt rekrutiert sich das sogenannte Entwicklungsteam, auch als Core-Team bezeichnet, aus 10 bis 30 Entwicklern, die beispielsweise bereits lange am Projekt mitarbeiten, besonders viel zu diesem beigetragen haben oder derzeit zu den aktivsten Entwicklern gehören. Somit ist das Core-Team in der Regel kein statisches, einmal eingerichtetes Team, sondern ist kontinuierlichen Veränderungen unterworfen. Aufbauend auf den vier Grundeigenschaften von OS/FS [vgl. Abschnitt 2.1.1.1] sowie durch die Nutzung des Internets als weltweite Kommunikationsplattform und als Verbreitungsmedium ist die Zusammenarbeit in OS/FS-Projekten auf alle ausgeweitet, die sich beteiligen wollen. So besteht jedes OS/FS-Projekt neben dem Core-Team aus einer Vielzahl von Entwicklern und Anwendern. Das Ausmaß der Beteiligung ist sehr unterschiedlich und reicht von einfachen Fehlerberichten (Bug Reports), Vorschlägen für neue Funktionen (Feature Requests) und Ideen für die Weiterentwicklung des Projekts bis hin zu Patches oder umfangreicheren

¹⁰⁶ Quelle: CHRISTL 2008: 5 f. (verändert).

Funktionserweiterungen.¹⁰⁷ Wie in Abb. 11 deutlich wird, haben Entwickler und Anwender eine große gemeinsame Schnittmenge. Der aus Entwicklern und Anwendern bestehenden Community wird insbesondere in jüngeren OS/FS-Projekten ein hoher Stellenwert beige-messen.

Lebenszyklus proprietärer Software

Im Gegensatz zum *Entwicklungszyklus* von OS/FS, wo die Behebung von Bugs und der Release neuer Versionen einen kontinuierlichen Prozess darstellen, ist proprietäre Software einem *Lebenszyklus* unterworfen, welcher stark durch Marketing-Entscheidungen und frühzeitig bekannt gegebene Release-Termine geprägt ist. Ähnlich OS/FS-Projekten haben heute aber auch in der Industrie kreativ selbstständige Entwicklungsteams ehemals stark hierarchische Strukturen unter einem Chefentwickler abgelöst. Nach einer ersten internen Testphase wird in der Regel eine Beta-Version der neuen bzw. weiterentwickelten Software auf den Markt gebracht. Käufer sind vor allem Anwendungsentwickler, die darauf angewiesen sind, dass ihre eigenen Programme zur jeweils neusten Software-Version kompatibel sind. Da bei proprietärer Software der Source-Code nicht veröffentlicht wird, haben Beta-Tester jedoch lediglich die Möglichkeit, gefundene Bugs zu melden (Bug-Reports).¹⁰⁸ Das Beheben von Fehlern obliegt allein dem Entwicklungsteam. Wie in Abb. 11 deutlich wird, bestehen bei proprietärer Software keine Überschneidungen zwischen Entwicklern und Anwendern.

2.1.4 OS/FS Entwicklungs- und Kommunikationswerkzeuge

Der Einsatz von OS/FS-Werkzeugen erfolgt im Wesentlichen in den Bereichen Information, Kommunikation, Dokumentation und Kollaboration und richtet sich nach den jeweiligen Anforderungen eines Projekts. Den einzelnen Werkzeugen gemeinsam ist eine öffentlich einsehbare und transparente Dokumentationsschicht. Eine Schreibberechtigung ist jedoch nur klar definierten Gruppen vorbehalten, die in der Regel eine einfache online Registrierung des Nutzers voraussetzen. Die Mitarbeit in systemkritischen Bereichen erfordert eine förmliche Nominierung und Wahl in die betreffende Arbeitsgruppe bzw. das Komitee.¹⁰⁹

Jedes OS/FS-Projekt präsentiert sich und seine Ressourcen im Internet über eine Website. Durch den verbreiteten Einsatz von Wikis (auch WikiWeb oder WikiWiki genannt) können die Webseiten von den Benutzern nicht nur gelesen sondern, nach vorausgegangener Registrierung auch online geändert werden. Sie dienen der Koordination der Entwickler und ermöglichen ein kollaboratives Wissensmanagement z.B. durch die gemeinschaftliche Erstel-

¹⁰⁷ GRASSMUCK 2004: 235 ff.

¹⁰⁸ GRASSMUCK 2004: 245.

¹⁰⁹ CHRISTL 2008b.

lung von Dokumentationen oder die Verwaltung von Bugs.¹¹⁰ Neben grundsätzlichen Informationen über das Projekt erhalten Entwickler und Anwender über die Projekt-Website auch Zugang zu weiteren Entwicklungs- und Kommunikationswerkzeugen eines OS/FS-Projekts. Die zentralen Kommunikationsmittel im Rahmen von OS/FS-Projekten sind Mailinglisten, Foren und auch Newsgroups. Einige Projekte setzen auch den Internet Relay Chat (IRC) ein, um eine direkte Kommunikation zu ermöglichen.

Eine wesentliche Anforderung in OS/FS-Projekten ist eine kooperative Verwaltung des Source-Codes, der den Kern eines jeden OS/FS-Projekts bildet. Weit verbreitet ist der Einsatz von Concurrent Versions Systems (CVS). In jüngerer Zeit wird auch verstärkt das Open Source Tool Subversion (SVN) verwendet, dessen erste stabile Version im Februar 2004 erschien. SVN wurde als moderne Ablösung von CVS entwickelt und baut auf den bewährten Methoden von CVS auf. Bei beiden Programmen handelt es sich um Versionskontrollsysteme, die es einem Entwickler erlauben, zunächst unabhängig an einer Arbeitskopie des Source-Code zu arbeiten und Änderungen wieder in ein zentrales Repository einzufügen. Das Repository stellt ein Projektarchiv dar, in dem alle zu einem Projekt gehörigen Dateien an einer für alle Entwickler zugänglichen Stelle abgelegt sowie die jeweiligen Versionen überwacht und verwaltet werden [vgl. Abb. 12]. So ist es möglich, auf ältere Versionen des Source-Codes zurückzugreifen oder den Weg der Veränderungen nachzuvollziehen.

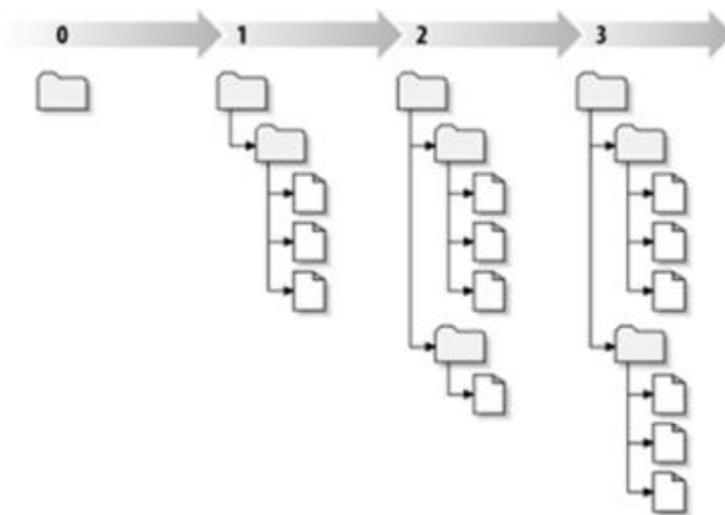


Abb. 12: Der Repository-Prozess¹¹¹

Bevor ein Entwickler Änderungen am Source-Code vornehmen kann, muss er sich registrieren und ist damit für alle anderen ansprechbar. Haben mehrere Entwickler gleichzeitig an

¹¹⁰ <http://de.wikipedia.org/wiki/Wiki> (Stand: 23.06.2008).

¹¹¹ Quelle: COLLIS-SUSSMAN et al. 2004: Kapitel 2. In das Repository werden alle Änderungen am Source-Code zurückgeschrieben. Es wächst mit der Zeit an (0, 1, 2, 3, n). Ein Zurückgreifen auf ältere Versionen ist jederzeit möglich.

einem Problem gearbeitet, werden die jeweiligen Änderungen zunächst lokal verschmolzen. Nur wenn sich das Ergebnis positiv testen lässt, wird der geänderte Source-Code in das Repository zurückgeschrieben. Andernfalls müssen sich die beteiligten Entwickler auf eine gemeinsame Lösung verständigen. Der Vorteil der Verwendung dieser sowohl CSV als auch SVN immanenten Methode, die auch als „Kopieren-Verändern-Zusammenfügen Lösung“ (Copy-Modify-Merge Solution) umschrieben wird – liegt darin, dass die Dateien des Source-Codes, die gerade von einem Entwickler verändert werden, nicht für andere gesperrt sind.¹¹² Dies ist insbesondere bei großen Softwareprojekten von Bedeutung, an denen mehrere Entwickler gleichzeitig arbeiten. Abb. 13 verdeutlicht die beschriebene „Kopieren-Verändern-Zusammenfügen Lösung“.

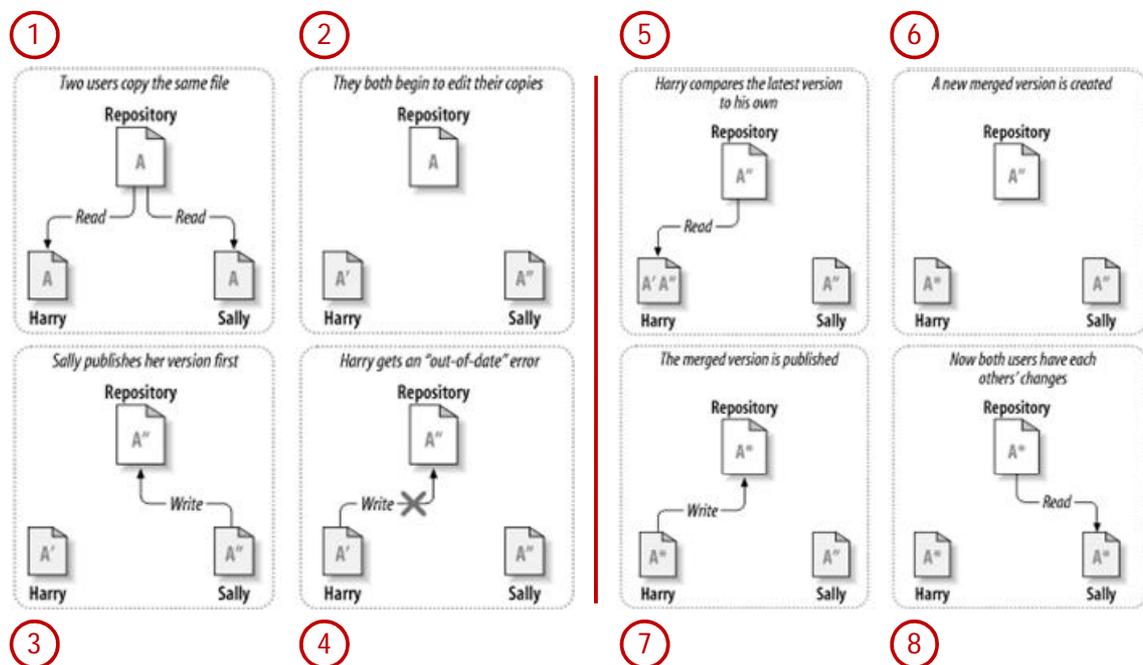


Abb. 13: Die „Kopieren-Verändern-Zusammenfügen Lösung“¹¹³

2.2. Warum Open Source und Freie Software?

In diesem Abschnitt werden Vorteile und Chancen sowie Risiken und Grenzen von OS/FS-Produkten mit proprietärer Software verglichen. Ferner werden die Rahmenbedingungen des Einsatzes von OS/FS in der öffentlichen Verwaltung thematisiert, insbesondere vor dem Aspekt der Bereitstellung von Geoinformationen.

2.2.1 Chancen und Grenzen des Einsatzes von Open Source und Freier Software

Der alleinige Verweis auf die ökonomischen Aspekte als Begründung für den Einsatz für OS/FS-Produkten in GDIs ist sicherlich zu kurz gegriffen. Auch wenn für OS/FS keine Li-

¹¹² GRASSMUCK 2004: 243 f.; COLLIS-SUSSMAN et al. 2004: Kapitel 2.

¹¹³ Quelle: COLLIS-SUSSMAN et al. 2004: Kapitel 2.

zenzgebühren anfallen, so entstehen auch hier, ähnlich wie beim Einsatz proprietärer Software, z.B. Kosten für die Schulung der Mitarbeiter oder für die Entwicklung (Personalkosten) für die Anpassung der Software an die spezifischen Erfordernisse. Im Folgenden werden wesentliche Vor- und Nachteile sowie die Chancen und Risiken in Hinsicht auf den Einsatz von OS/FS im Vergleich zu proprietärer Software thematisiert.

2.2.1.1 Vorteile und Chancen

Sicherheit, Stabilität und Qualität der Software

Wie bereits in Abschnitt 2.1.1 beschrieben, unterscheidet sich OS/FS-Software von proprietärer Software in erster Linie dadurch, dass der Quellcode für jedermann frei verfügbar, einsehbar und analysierbar ist und natürlich auch verändert werden darf. Diese Tatsache zieht eine Reihe von Optionen nach sich:

OS/FS-Produkte werden vergleichsweise hohen Sicherheitsanforderungen gerecht. Größere Sicherheitslücken bestehen in der Regel nur bei proprietären Softwareprodukten, die in der Vergangenheit auch immer wieder Ziel von Virenangriffen waren. Dieses mag zunächst unlogisch erscheinen, zumal aufgrund der Geheimhaltung, der Quellcode eigentlich keinem Externen bekannt sein sollte bzw. niemand außer der Hersteller selbst Veränderungen vornehmen darf. Das oftmals zu beobachtende Phänomen mangelnder Sicherheit ist u.a. darauf zurückzuführen, dass die Entwicklung von proprietärer Software in der Regel aus kommerziellem Interesse erfolgt. Zum einen wird häufig der Reduzierung der Entwicklungskosten größere Bedeutung beigemessen als einer ausführlichen Qualitätskontrolle des Produkts. Zum anderen kommt eine ausreichende Qualitätssicherung, bedingt durch frühzeitig bekannte gegebene Release-Termine, vielfach zu kurz. Durch den Release von sog. Beta-Softwareversionen wird die Fehlersuche und damit Teile der Entwicklung dann an sog. Betatester (Kunden) weitergegeben. Diese können gefundene Fehler nur melden, aber nicht eigenständig beheben. Ein weiterer Grund für Sicherheitslücken bei proprietärer Software ist in ihrer Evolution zu sehen. Der Quellcode der Software wird über Jahre hinweg an die jeweiligen Markterfordernisse angepasst. Aus Kostengründen werden dabei häufig bestehende ältere Technologien weiter verwendet, obwohl eine komplette Neuentwicklung aus technischer Sicht zielführend gewesen wäre.¹¹⁴

Eine der wesentlichen Stärken von OS/FS-Projekten im Vergleich zu proprietärer Software stellt der offene Umgang mit Fehlern (Bugs) in der Software dar. „Die Stabilität [von OS/FS] Software, d.h. ihr Grad an Fehlerfreiheit, verdankt sie nicht der Genialität ihrer Entwickler, sondern der Tatsache, dass jeder Anwender Fehler an den Pranger stellen kann und die kollektive Intelligenz von Hunderten von Entwicklern meist sehr schnell eine Lö-

¹¹⁴ HILLEMAYER et al 2006: 28 f; ADAMS et al 2004: 134

sung dafür findet.“¹¹⁵ Ein weiterer Vorteil von OS/FS-Projekten ist die Möglichkeit, neue Ideen und Funktionalitäten bis ins Detail zu diskutieren und auszutesten, bevor eine neue Version der Software herausgebracht wird.¹¹⁶ Dies gewährleistet die meist vorhandene höhere Stabilität von OS/FS im Vergleich zu proprietären Lösungen.

Interoperabilität bzw. Kompatibilität

OS/FS ist häufig auch zu einer Vielzahl von anderen Produkten kompatibel. Die Entwicklung von OS/FS-Produkten ist fast immer von dem Gedanken getragen, eine größtmögliche Verbreitung der Software zu erreichen. Erreicht wird dies durch die Verwendung von offenen Standards, die sich in der Open Source Szene etabliert haben¹¹⁷. Ergänzend sei darauf hingewiesen, dass große Anbieter von proprietärer Software aus kommerziellen oder Monopolgründen mitunter bewusst Inkompatibilitäten ihrer Programme zu anderer Software, insbesondere zu OS/FS fördern, um ihre Marktdominanz weiter auszubauen.¹¹⁸

Bedarfsgerechte Anpassung durch Modularität (Skalierbarkeit)

Ein herausragendes Charakteristikum von OS/FS ist deren modularer Aufbau. Die Abgrenzung von einzelnen Softwarefunktionalitäten in mehreren Teilpaketen hat den Vorteil, dass Softwarelösungen flexibel erweiterbar, aber auch reduzierbar sind. Bei Stabilitätsproblemen der Software führt dies in der Regel nur zu einem Versagen einer Komponente oder weniger Module, nicht aber zum Totalabsturz des Systems. Des Weiteren kann die Fehlersuche auf wenige Module beschränkt werden und gestaltet sich somit effizienter. Als prominentes Beispiel für diese starke Modularität von OS/FS können stellvertretend für viele andere Beispiele die verschiedenen Distributionen des Betriebssystems Linux genannt werden.

Der modulare Aufbau, gepaart mit der Gewährleistung der vier Freiheiten von OS/FS, ermöglicht auch eine bedarfsgerechte Anpassungen von Softwarefunktionalitäten und damit eine Individualisierung der Software. Kleine Modifikationen lassen sich meist zeitnah und mit wenig Aufwand umsetzen.¹¹⁹

Eine individuelle Anpassung der Software ist jedoch auch mit Kosten (z.B. Personalaufwand) und möglicherweise mit Folgewirkungen für den Support verbunden. Wie auf diese Art entstehende Kosten und Folgewirkungen abgefangen bzw. gemindert werden können, wird im Abschnitt zur Investitionssicherheit thematisiert

¹¹⁵ GRASSMUCK 2004: 243.

¹¹⁶ GRASSMUCK 2004: 246.

¹¹⁷ HILLEMAYER et al 2006: 29.

¹¹⁸ HILLEMAYER et al 2006: 29.

¹¹⁹ HILLEMAYER et al 2006: 27.

Unabhängigkeit von Softwareanbietern

Die Offenlegung des Quellcodes sowie seine freie Verfüg- und Verwendbarkeit kann dazu genutzt werden, marktbeherrschender Monopolwirtschaft entgegenzuwirken und den Wettbewerb zwischen den OS/FS-Anbietern bzw. OS/FS-Dienstleistern zu fördern. Dieser Umstand lässt sich ferner dazu nutzen, dass Anbieter bzw. Dienstleister zumindest theoretisch uneingeschränkt gewechselt werden können. Das gleiche gilt auch bei der Anpassung von bestehenden Software-Applikationen an geänderte Rahmenbedingungen. Der Nutzer ist nicht mehr nur an den Hersteller der Software, der in der Regel außerhalb jeglichen Wettbewerbs tätig werden kann, gebunden, sondern kann auch auf wirtschaftliche Konkurrenzangebote zurückgreifen.¹²⁰

Wettbewerbsförderung

Da die meisten großen proprietären Softwareprodukte von Herstellern aus den USA (Microsoft, IBM, Oracle, Apple) kommen und dort auch der wesentliche Anteil der Entwicklungsarbeit erfolgt, findet ein erheblicher Wertschöpfungsabfluss statt. Durch den Einsatz von OS/FS wird den öffentlichen Verwaltungen, unter Berücksichtigung der Vergabevorschriften die Möglichkeit eröffnet, auch die regionalen Softwareanbieter und -dienstleister berücksichtigen zu können. Viele kleine und mittelständige Unternehmen beteiligen sich an der Entwicklung von OS/FS-Projekten. Sie verfügen somit über detaillierte Kenntnisse über die jeweiligen OS/FS-Projekte und bieten Support bei der Entwicklung von Anwendungen auf der Basis spezieller OS/FS-Produkten an.

Investitionssicherheit

Der Vorteil der Herstellerunabhängigkeit wirkt sich auch auf die Investitionssicherheit aus. OS/FS-Produkte werden durch eine Gruppe von Entwicklern und Nutzern gestützt, die das Know-how, die Softwareentwicklung selbst sowie die damit verbundenen Risiken gemeinsam tragen. Damit unterliegen OS/FS-Projekte keiner Unternehmensstrategie.

Es kann durchaus davon ausgegangen werden, dass die Entwicklung und Pflege von größeren OS/FS-Projekten, die sich bereits auf dem Markt etabliert haben, nicht einfach bzw. ohne erkennbaren Vorlauf eingestellt werden. Selbst wenn doch bzw. bei nachlassender Aktivität einer Community kann der Nutzer einer Software diese eigenständig an neue Entwicklungen anpassen bzw. einen Dienstleister damit beauftragen.

Letzteres ist insbesondere bei Gemeinschaftsprojekten im öffentlichen Sektor möglich, bei denen mehrere Verwaltungseinheiten ein OS/FS-Produkt bzw. die auf dessen Basis entwickelte Applikation gemeinsam nutzen.¹²¹

¹²⁰ HILLEMAYER et al, 2006: 28.

¹²¹ Adams et al, 2004: 134 f.

2.2.1.2 *Nachteile und Risiken*

Mangelnde Unterstützung neuer Hardware und fehlende Installationsroutinen

Die Lauffähigkeit von OS/FS-Produkten ist auf neuer Hardwaretechnologie in der Regel zunächst nur eingeschränkt bzw. gar nicht möglich. Dieses ist darauf zurückzuführen, dass Hardware-Hersteller häufig mit Anbietern proprietärer Software kooperieren und damit schon aus eigener unternehmensstrategischer Sicht meist gar kein Interesse an der Verbreitung von OS/FS-Produkten haben. Somit ist neu auf den Markt gekommene Hardware häufig nur auf proprietäre Softwareprodukte ausgelegt, zumal auch die OS/FS-Nutzergruppe noch vergleichsweise klein ist und sich damit den Anbietern von Hardwareprodukten kein ausreichender Absatzmarkt bietet.

Die oben [vgl. S. 39] bereits erwähnte Modularität führt zu sehr individuellen und vielfältigen Softwarelösungen. Der Aufwand, diese unterschiedlichen Konfigurationen auf der neuen Hardware lauffähig zu machen ist in der Regel sehr hoch. Das führt dazu, dass Treiber für neue Hardware zunächst meist noch entwickelt werden müssen.¹²²

Keine Gewährleistung, kein Service und Support

OS/FS-Lizenzen bieten in der Regel keine Herstellergarantie. Sie schließen ferner jegliche Haftbarmachung der an der OS/FS Produktentwicklung Beteiligten explizit aus. Dieses hat in erster Linie zum Ziel die an der Softwareentwicklung Beteiligten, deren Mitarbeit an einem OS/FS-Projekt in der Regel aus individuellen Beweggründen erfolgt und unentgeltlich ist, vor einem haftungsrechtlichen Risiko zu schützen. Somit entfällt die Option für den Nutzer von OS/FS-Produkte, den Lizenzgeber wegen (schwerwiegender) Mängel an der Software und Schäden bei einem Dritten in Regress zu nehmen.

Diese genannten Nachteile sind allerdings, solange es sich um keine schwerwiegenden Fehler in der Software handel, gut durch gesondert vereinbarte Service- und Wartungsdienstleistungen abzufangen, die mit auf die Weiterentwicklung und den Support von Softwareprodukten spezialisierten Dienstleistern vertraglich festgelegt werden.

Problematisch ist eine fehlende Gewährleistung (Herstellergarantie) jedoch bei schwerwiegenden Mängeln, wie Sicherheitslücken oder Programmierfehlern, die zu Schäden geführt haben. Hierbei muss der Einsatz von proprietärer Software jedoch nicht unbedingt vorteilhaft gegenüber OS/FS sein, da die Vertragsbedingungen häufig zugunsten des Softwareanbieters ausformuliert sind.

Keine Entwicklungsgarantie

Grundsätzlich ist festzuhalten, dass bei OS/FS-Projekten, im Gegensatz zu weitverbreiteten Softwareprodukten großer proprietärer Softwarehersteller, die eine quasi Entwicklungsga-

¹²² HILLEMAYER et al, 2006. S. 30 f.

rantie genießen, eine solche Garantie nicht gegeben werden kann. Zum Teil kann dieses jedoch durch Entwicklungsaufträge an Dienstleister abgefangen werden.¹²³

Es ist allerdings zu erwarten, dass kommerzielle Softwareanbieter praktisch immer ein Interesse am zukünftigen Bestand und der Weiterentwicklung ihrer Software, egal ob proprietär oder frei, haben, um wettbewerbsfähig bleiben bzw. weiter expandieren zu können.

Einschränkungen in der Weiterentwicklung der verfügbaren Software können allerdings durch den offen gelegten Quellcode bereits gut kompensiert werden. So können Anwender in dem Fall, dass ein Produkt nicht mehr weiterverfolgt wird, die Software eigenständig weiterentwickeln [vgl. *Investitionssicherheit S. 40*].

Mangelnde Dokumentation

Die ausreichende Dokumentation einer Software hinsichtlich ihrer Funktionen und ihrer Architektur ist eine wichtige Voraussetzung dafür, dass Projekte weiterentwickelt werden können. Bei Software im Allgemeinen und somit natürlich auch bei OS/FS-Produkten kommt die Dokumentation allerdings häufig zu kurz. Das Interesse vieler Entwickler liegt häufig primär auf der Lösung programmtechnischer Herausforderungen als in der anschließenden lückenlosen Dokumentation. Vergleichbaren Problem begegnen aber auch die Hersteller von proprietärer Software, insbesondere dann, wenn an bestehender Software Änderungen erforderlich werden, der einst federführende Programmierer das Unternehmen allerdings schon verlassen hat und das Know-how unternehmensintern nicht gehalten werden konnte.

Umschulungsaufwand

Im Zusammenhang mit großen Softwareprojekten wird als ein Nachteil von Softwareumstellungen häufig der Umschulungsaufwand der Anwendern genannt.¹²⁴ Dies trifft insbesondere bei Client-Anwendungen zu. Die Umstellung auf OS/FS-Produkte bedeutet für viele Nutzer ferner ein Abrücken von den bisherigen Arbeitsgewohnheiten und -verfahren. Der Schulungsaufwand ist vielfach erheblich, da Mitarbeiter häufig nur im Umgang mit den verbreiteten proprietären Anwendungen (z.B. Office-Anwendungen von Microsoft oder den Produkten von ESRI oder Intergraph im GIS-Bereich) vertraut sind. Gleiches gilt im Prinzip auch für den Investitionsaufwand in das Know-how von Fachadministratoren, die sich in die Bedienung der neuen Software einarbeiten müssen. Die hier beschriebenen Nachteile sind aber nicht nur OS/FS spezifisch, sondern können sich auch bei der Umstellung auf proprietäre Software eines anderen Anbieters einstellen.

¹²³ HILLEMAYER et al, 2006. S. 31.

¹²⁴ GEORGIEV et al 2004.

Insbesondere im GIS-Bereich, wo an vielen Arbeitsplätzen zurzeit häufig noch komplexe GIS-Systeme lediglich zu Auskunftszwecken bzw. einfachen Anwendungen eingesetzt werden, könnten aufwendige Anwenderschulungen zukünftig durch den Aufbau von einfachen Web Mapping-Applikationen entbehrlich werden.

Soziale Faktoren

Eines der größten Hindernisse ist allerdings ein immer noch stark ausgeprägtes Misstrauen bei den Mitarbeitern einer Organisation gegenüber OS/FS-Produkten,¹²⁵ insbesondere aber im Bereich des Managements, das letztlich die Entscheidung zugunsten des Einsatzes von OS/FS-Produkten treffen muss. Zurückzuführen ist dies auf das noch häufig weit verbreitete „Hacker“-Image der OS/FS-Szene, der derzeit in einigen Bereichen noch ungeklärte rechtliche Status von OS/FS-Lizenzen sowie die fehlende Gewährleistung für OS/FS-Produkte. Beseitigen lassen sich solche soziale Hürden meistens nur durch eine wirksame Kommunikation von nachhaltig tragfähigen OS/FS-Lösungen sowie des Entwicklungsfortschritts im Bereich des OS/FS-Einsatzes. In der öffentlichen Verwaltung hat sich das OSOR.EU (Open Source Observatory and Repository for European Public Administration) als Plattform etabliert. Ziel ist die Verbreitung von Good Practice Beispielen zur Nutzung von OS/FS in der öffentlichen Verwaltung.¹²⁶

Zusammenfassung

Tab. 7 gibt zusammenfassend die im Vergleich der untersuchten Softwarekategorien identifizierten jeweiligen Stärken wieder. Im Umkehrschluss lassen sich daraus die entsprechenden Schwächen ableiten. Es ist anzumerken, dass an dieser Stelle lediglich der allgemeine Trend wiedergegeben werden kann, da Softwareprodukte sehr individuell und vielschichtig sein können. Des Weiteren müssen die unterschiedlichen Rahmenbedingungen des jeweiligen Einsatzes der Software berücksichtigt werden. Auch kann den einzelnen Aspekten je nach Anwendungsfall ein unterschiedlicher Stellenwert (Bewertung) beigemessen werden.

| <i>Stärken von OS/FS</i> | <i>Stärken von proprietärer Software</i> |
|---|---|
| Sicherheit und Stabilität durch offenen Source-Code | Unterstützung neuer Hardware |
| Interoperabilität und Kompatibilität | Existenz von Installationsroutinen |
| Modularität (Skalierbarkeit) | Gewährleistung von Service und Support |
| Unabhängigkeit von Softwareanbietern | Quasi Entwicklungsgarantie |
| Investitionssicherheit | Höhere Akzeptanz bei den Nutzern |
| Rechte und Lizenzen | |

Tab. 7: Vergleich der Stärken von OS/FS und proprietärer Software.

¹²⁵ HILLEMAYER et al, 2006. S. 37.

¹²⁶ <http://www.osor.eu/> (Stand 24.08.2008).

Die oben aufgeführten Aspekte bilden die Grundlage für die Bewertung der in der Fallstudie eingesetzten Software. Dabei gilt es, die identifizierten Stärken bzw. Schwächen durch den Anwendungsfall zu belegen oder ggf. zu widerlegen.

2.2.2 Einsatz von Open Source und Freier Software im öffentlichen Sektor

In einem Antrag des Deutschen Bundestags aus dem Jahr 2001 begrüßt dieser ausdrücklich „die Förderung von Open-Source-Produkten und fordert die Einführung von unter Open-Source-Lizenzen erstellten Produkten in der Bundesverwaltung“.¹²⁷ Seitdem hat OS/FS in der öffentlichen Verwaltung kontinuierlich an Bedeutung gewonnen. So zeigt beispielsweise eine Studie des Fraunhofer IAO aus dem Jahr 2005, dass 79 Prozent der befragten öffentlichen Einrichtungen sich mit den Einsatzmöglichkeiten von OS/FS auseinander setzen.¹²⁸ GÜNTHER zeigt wesentliche Motive für diese Entwicklung auf [vgl. Abb. 14]:

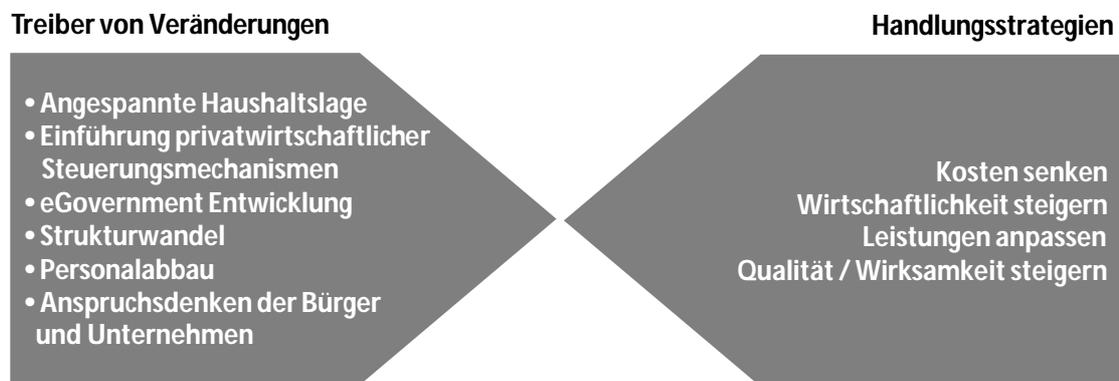


Abb. 14: Handlungsfelder öffentlicher Verwaltungen¹²⁹

2.2.2.1 Rahmenbedingungen für den Einsatz von OS/FS im öffentlichen Sektor

Im Vergleich zum privaten und privatwirtschaftlichen Bereich bietet der öffentliche Sektor besondere Rahmenbedingungen, die es erlauben, OS/FS-Produkte effizient einzusetzen. Die föderale Struktur der öffentlichen Verwaltung der Bundesrepublik ist mit 16 Bundesländern und insgesamt 313 Kommunen¹³⁰ vergleichsweise kleinteilig.

So stehen öffentliche Verwaltungen im Vergleich zu Unternehmen der privaten Wirtschaft nicht miteinander im Wettbewerb. Der Einsatz bestimmter Software, der bei den nach marktwirtschaftlichen Mechanismen agierenden Unternehmen in der Regel nach strategischen und wettbewerbserheblichen Gesichtspunkten erfolgt, hat bei einer öffentlichen Verwaltung keine kommerzielle Bedeutung.¹³¹

¹²⁷ BT Drucksache 14/5246, 2001: 5.

¹²⁸ GÜNTHER 2008: 207.

¹²⁹ nach GÖLZ und HOFMANN 2005, in: GÜNTHER 2008: 208.

¹³⁰ <http://www.kreise.de/landkreistag> (Stand 24.08.2008).

¹³¹ ADAMS et al 2004: 22

Aus gleichen zentralen und bundeseinheitlichen Vorschriften, wie z.B. der Informationspflicht gegenüber dem Bürger [vgl. z.B. *IFG und UIG S. 12*] oder den Vorgaben aus der INSPIRE-Richtlinie [vgl. *Abschnitt 1.2.1*] entstehen ähnliche Anforderungen und Fragestellung, die eine gemeinsame Lösungsfindung nahe legen. Aufgrund der oben beschriebenen Kleinteiligkeit können viele Themenstellungen, beispielsweise im Bereich des Katastrophenschutzes, der Verkehrsinfrastrukturen sowie des Naturschutzes nur über Verwaltungsgrenzen hinweg sinnvoll gelöst werden.¹³²

In der Regel haben fast alle öffentlichen Verwaltungen ähnliche Anforderungen. Dieses gilt insbesondere für den Bereich der Bereitstellung von Geoinformationen und räumlichen Basisdaten, auch wenn z.B. ein küstenbehaffeter Flächenstaat wie Schleswig-Holstein natürlich andere Schwerpunkte bei der Geodatenverwaltung setzen muss, wie das (vor-)alpine Bayern oder ein Stadtstaat wie Berlin.¹³³

Gleichzeitig wurden in den verschiedenen öffentlichen Einrichtungen, bedingt durch die kleinteiligen Strukturen in Deutschland und dem damit einhergehenden Fehlen eines zentralen IT-Rahmenkonzeptes, unterschiedliche Schwerpunkte beim Einsatz von Soft- und Hardware gesetzt. Dieses hat auch dazu geführt, dass die Qualifikationen und die Fachkenntnisse der Mitarbeiter in den jeweiligen IT-Bereichen sehr differenziert und heterogen ausgeprägt sind. Da die Verwaltungen bereits flächenweise intensiv zusammenarbeiten, lässt sich das (Fach-)Wissen praktisch verlustfrei und damit kosteneffizient weiter vermitteln sowie austauschen. Ferner können die auf Basis von OS/FS entwickelten Applikationen einfach weiter gegeben werden. Dies ist beispielsweise für kleinere Verwaltungseinheiten, denen es an personeller und finanzieller Ausstattung fehlt, von Bedeutung.

Hinzu kommt, dass in der Regel fast jede Verwaltungseinheit über entsprechendes Personal verfügt, das sich zumindest in Grundzügen, bei größeren Einrichtungen häufig jedoch tiefgehend mit Software auskennt. Somit steht in der öffentlichen Verwaltung ein gewaltiges Entwicklerpotential zur Verfügung.¹³⁴ Insbesondere in größeren Verwaltungen oder durch die Kooperation mehrerer Verwaltungseinheiten könnten auf Basis von OS/FS eigene Anwendungen bedarfsgerecht entwickelt werden, die denen kommerzieller Softwarehersteller, unabhängig davon, ob es sich um proprietäre Software oder OS/FS handelt, in nichts nachstehen müssen.

¹³² ADAMS et al 2004: 21 f

¹³³ ADAMS et al 2004: 22

¹³⁴ ADAMS et al 2004: 23

2.2.2.2 *Bereitstellung von Geoinformationen im öffentlichen Sektor*

Als einen wesentlichen Trend beim IT-Einsatz öffentlicher Dienstleister identifiziert GÜNTHER¹³⁵ die vermehrte Entwicklung webbasierter Anwendungen. Neuentwicklungen oder die Ablösung bestehender (meist proprietärer) Anwendungen stellen dabei eine Chance für den Einsatz von OS/FS dar. Teilweise werden sogar Release-Zyklen oder geänderte Wartungsbedingungen gezielt dazu genutzt, „eine Plattformunabhängigkeit von Applikationen durch deren webbasierte Umsetzung zu erreichen.“¹³⁶

Auch die in der vorliegenden Arbeit thematisierten GDIs stellen webbasierte Anwendungen und damit Bereiche dar, in denen sich OS/FS Lösungen besonders gut entwickeln können. Durch das Inkrafttreten der INSPIRE-Richtlinie zur Schaffung einer europäischen GDI sowie durch zahlreiche Bestrebungen zum Aufbau von nationalen GDIs, wie z.B. der deutschen GDI-DE, wird dieser Trend weiter gestärkt und erfährt einen Zuwachs an Aufmerksamkeit. Dies zeigt sich beispielsweise an zahlreichen Publikationen in Fachzeitschriften aus dem Bereich der Geoinformatik, aber auch durch die Thematisierung von GDI-Aspekten auf Fachtagungen (z.B. GDI-Tage Nord 2007) und Anwenderkonferenzen (z.B. FOSSGIS 2008).

Mittlerweile sind bereits verschiedene GDI-Projekte auf Basis von OS/FS Software realisiert worden bzw. werden derzeit entwickelt. Die hier gemachten Erfahrungen stellen eine solide Basis für die Entwicklung neuer GDIs dar. In diesem Zusammenhang sei das GeoPortal-RP¹³⁷ erwähnt. Hier wurde auf Landesebene bewusst die Entscheidung getroffen, ausschließlich Software aus dem Bereich OS/FS einzusetzen, um eine Kooperation mit Verwaltungseinheiten auf unteren Ebenen zu ermöglichen. Durch die Applikationsentwicklung auf Basis von OS/FS-Produkten sollte die Möglichkeit geschaffen werden, die Anwendungen an kleinere Verwaltungen weiterzugeben, denen das Personal und damit das Know-how zum Aufbau von eigenen Lösungen fehlen. Auch eine Kooperation, z.B. durch den Austausch einzelner Funktionalitäten oder Module, über die Grenzen von Bundesländern hinweg ist grundsätzlich möglich.¹³⁸

2.3. **GDI Systemarchitektur mit Open Source und Freier Software**

In diesem Abschnitt erfolgt die Entwicklung eines Szenarios für die interoperable Bereitstellung von Erdbebeninformationen („naturbedingten Risiken“; Annex III), wie sie in der INSPIRE-Richtlinie gefordert ist. GDIs können ausschließlich mit OS/FS-Produkten umge-

¹³⁵ GÜNTHER 2008: 206 f.

¹³⁶ GÜNTHER 2008: 207.

¹³⁷ <http://www.geoportal.rlp.de/> (Stand 24.08.2008).

¹³⁸ RETTERATH, Armin (2008): Gespräch auf der FOSSGIS 2008 in Freiburg.

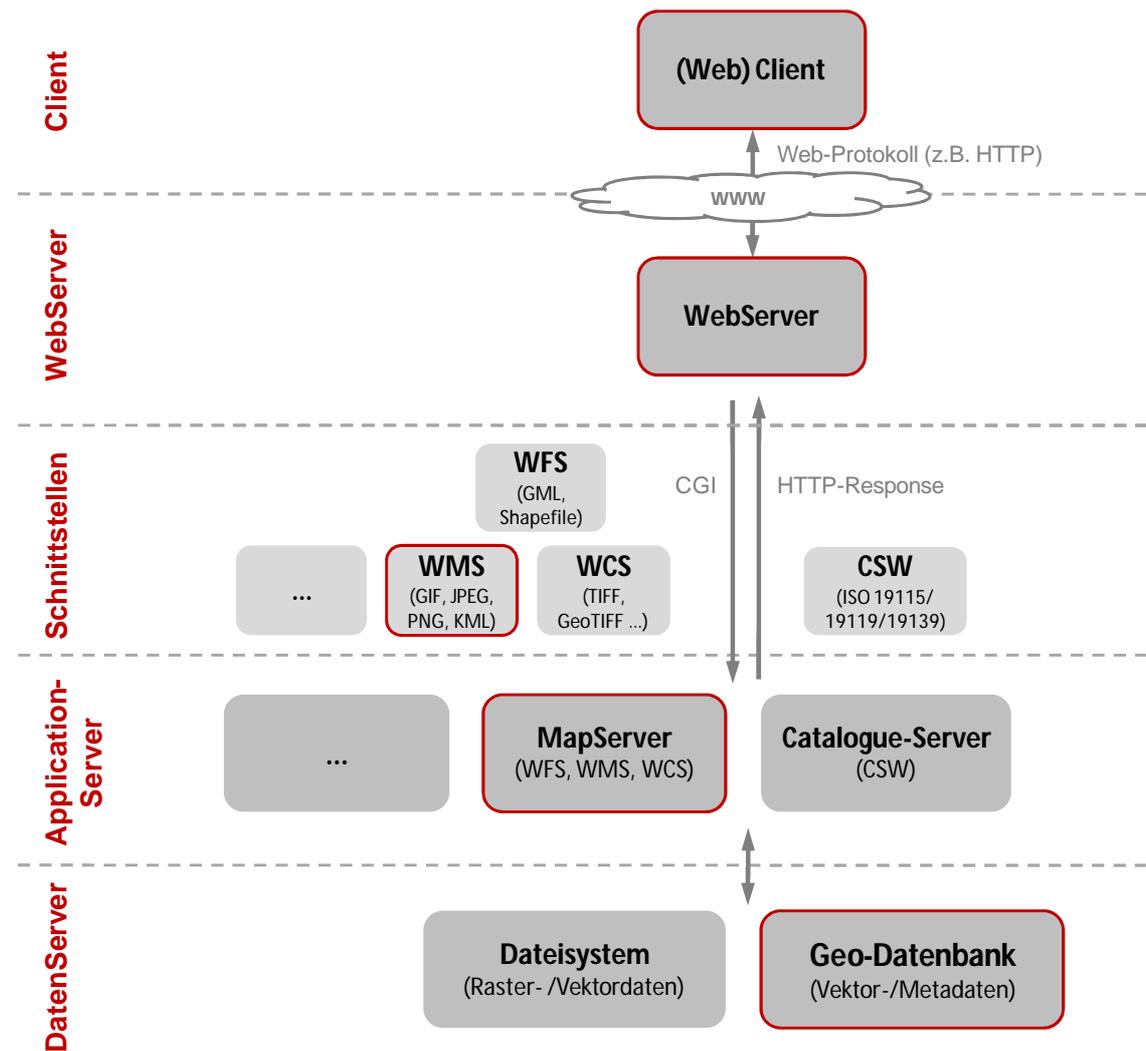
setzt werden [vgl. oben – z.B. *GeoPortal-RP*]. Solange sie die Schnittstellen die seitens des OGC und der ISO definierten Normen und Standards implementieren, ist eine interoperable Kommunikation zwischen den einzelnen Komponenten einer GDI gewährleistet [vgl. *Abschnitt 1.3.1*]. Eine flexible Erweiterung oder ein Austausch einzelner GDI-Komponenten zu einem späteren Zeitpunkt ist dann problemlos möglich. Die hinter einer GDI stehende technische Umsetzung baut auf einer serverseitigen Mehrschichten-Architektur auf, wie sie in Abb. 15 exemplarisch dargestellt ist.

2.3.1 Beispiel einer einfachen Kartenanfrage

Der Nutzer fordert mithilfe von Werkzeugen (z.B. zoom in/out), die über eine graphische Benutzeroberfläche, dem Web-Client, zur Verfügung gestellt werden, einen neuen Kartenausschnitt an. Der Request (Anfrage) wird in Form einer URL über das HTTP-Protokoll (Hypertext Transfer Protocol) an den Webserver übertragen. Hier wird der Request verarbeitet und die mitgelieferten Parameter an dem MapServer weitergeleitet. Dies geschieht über eine CGI-Schnittstelle (Common Gateway Interface – Allgemeine Vermittlungsrechner-Schnittstelle) zwischen dem Webserver und dem MapServer, deren vorrangige Aufgabe es ist, dynamische Inhalte bereitzustellen. Der MapServer bereitet die Eingabeparameter auf und greift über eine Netzwerkverbindung oder lokal auf die entsprechenden Geodatenbestände zu. Diese können entweder dateibasiert oder in einer relationalen Datenbank vorgehalten werden. Das Resultat ist eine neue Karte in Form eines PNG- oder GIF-Bildes, welches über das HTTP-Protokoll an den Webserver und von diesem an den Web-Client ausgeliefert wird (Response).¹³⁹

Auch der MapServer selbst kann zum Client werden und anstelle des Zugriffs auf ein Dateisystem oder eine Datenbank beispielsweise einen anderen WMS oder WFS anfragen. Gleiches gilt im Fall einer Metadatenrecherche für einen Catalogue-Server. Ein Geodatenkatalog kann eine Anfrage nach einem Suchbegriff an alle angeschlossenen Kataloge weiterleiten, die über eine standardkonforme CSW-Schnittstelle [vgl. *Abschnitt 1.3.4.1*] für die Metadatenrecherche verfügen. Der Datenkatalog fasst den Rücklauf aus den angeschlossenen Einrichtungen tabellarisch zusammen und übermittelt diese über das HTTP-Protokoll an den Clienten.

¹³⁹ ADAMS 2004: 41 f.; <http://de.selfhtml.org/servercgi/cgi/cgihtml.htm> (Stand: 29.06.2008).

Abb. 15: Einfache Client-Server Architektur¹⁴⁰

Im Folgenden werden nun die in Abb. 15 dargestellten technischen GDI-Komponenten gemeinsam mit jeweils einem ausgesuchten OS/FS-Produkt der entsprechenden Kategorie vorgestellt. Die Erläuterung der Implementierung erfolgt dann am Beispiel einer vorläufigen GDI für seismische Parameterdaten der Bundesanstalt für Geowissenschaften und Rohstoffe (BGR) in Abschnitt 3.2. Der Auswahl der OS/FS-Produkte wurden folgende Kriterien zugrunde gelegt:

- Entsprechend der Aufgabenstellung der Arbeit muss es sich um Produkte aus dem Bereich OS/FS handeln.
- Die Softwareprodukte müssen sich in die bestehende Systemlandschaft bei der BGR integrieren.

¹⁴⁰ Die rot umrandeten Komponenten werden im Rahmen der Systemimplementierung in Abschnitt 3 umgesetzt.

- Die Softwareprodukte müssen über geeignete Schnittstellen verfügen, so dass eine Kommunikation zwischen den GDI-Komponenten ohne Anpassungsaufwand möglich ist.
- Die OS/FS-Produkte müssen über eine aktive Community verfügen, so dass beim Aufbau der vorläufigen GDI die typischen Kommunikations- und Entwicklungswerkzeuge [vgl. Abschnitt 2.1.4] von OS/FS-Projekten genutzt werden können sowie eine Investitionssicherheit und damit die Möglichkeit einer Weiterentwicklung der vorläufigen GDI durch einen Fortbestand des jeweiligen OS/FS-Projekts gegeben ist.

Den ausgewählten OS/FS-Projekten gemeinsam ist ihre weltweite Anwendergemeinschaft mit zahlreichen Beispielanwendungen, was einen erfolgskritischen Faktor darstellt.

2.3.2 Datenhaltungskomponente

Die Vorhaltung von Geodaten verschiebt sich zunehmend von strukturierten Dateisystemen hin zu relationalen Datenbanken.¹⁴¹ Das ist insbesondere der Fall bei Vektordaten [vgl. *seismische Parameterdaten*, Abschnitt 3.2.3]. Für verschiedene Datenbanksysteme gibt es auf dem Markt räumliche Aufsätze, die eine Verwaltung von Geodaten und den Zugriff auf diese mithilfe von Datenbankwerkzeugen ermöglichen. Im OS/FS-Bereich sind hier insbesondere MySQL und PostGIS (als Aufsatz für PostgreSQL) zu nennen.¹⁴² Im Folgenden wird die von der Firma Refrations Research entwickelte räumliche Erweiterung PostGIS kurz vorgestellt. Diese hat sich im Bereich der OS/FS-GIS-Tools zu einem Standard „Backend“ entwickelt.¹⁴³ Sie wird auch im Implementierungsbeispiel der vorliegenden Arbeit (Abschnitt 3) vor allem deshalb eingesetzt, da sie über eine Schnittstelle zum UMN Map-Server verfügt.

PostGIS

PostGIS ist der OGC-konforme räumliche Aufsatz für die derzeit leistungsstärkste relationale Open Source-Datenbank PostgreSQL. Mit PostGIS können Geodaten über SQL-Anweisungen erzeugt und verändert werden. In einer serverseitigen Datenbanklösung stehen dem Anwender eine Vielzahl an Funktionen zur Verfügung, mit denen er beispielsweise Verschneidungen und Umprojektionen durchführen sowie Nachbarschaftsbeziehungen oder Flächen berechnen kann. PostGIS ermöglicht den simultanen Zugriff (Mehrbenutzer-

¹⁴¹ ADAMS 2004: 50

¹⁴² Ein wesentlicher Unterschied zwischen den räumlichen Komponenten von MySQL und PostGIS besteht darin, dass PostGIS eine größere Funktionsvielfalt als MySQL aufweist. Ferner bietet MySQL keine Transaktionsintegrität für Geometrien (CHRISTL 2008a: 41).

¹⁴³ RAMSEY 2007: 26

fähigkeit) durch mehrere Programme, wodurch die Daten gleichzeitig bearbeitet und in einer Karten- bzw. GIS-Anwendung visualisiert werden können.

Eine der besonderen Stärken von PostGIS liegt in der vollständigen Implementierung der OGC Simple Features Specification for SQL (SFS), welche eine wesentliche Voraussetzung für die Interoperabilität räumlicher Daten bildet. Dieser Aspekt war mit ausschlaggebend für die Verwendung von PostGIS für die in Abschnitt 3.2 beschriebene Lösung. Viele andere Anwendungen implementieren nur Teile des SFS-Standards.¹⁴⁴ Die Simple Feature Schnittstelle ermöglicht die Interoperabilität zwischen der PostGIS Datenbank und der jeweiligen Anwendung, z.B. dem UMN MapServer, auf der Basis von Geometrie-Objekten, den Simple Features [vgl. Tab. 8]. Bei diesen handelt es sich um zweidimensionale Vektorgeometrien, die aus Punkten (Point), Linien (Curve), Flächen (Surface) und Sammlungen von Geometrien (GeometryCollections) bestehen und einem räumlichen Bezugssystem zugeordnet sind.¹⁴⁵ Neben den Geometrie-Objekten definiert die SFS Standards für Analyse- und Manipulations-Funktionen von Geo-Objekten sowie Metadatentabellen¹⁴⁶. Die OGC SFS definiert zwei Standards zur Speicherung von räumlichen Objekten: die Formate Well-Known Text (WKT) und Well-Known Binary (WKB). Tab. 8 enthält Beispiele für die Definition räumlicher Objekte im WKT-Format.

| <i>Geometry Type</i> | <i>Text Literal Representation</i> | <i>Comment</i> |
|----------------------|---|--|
| Point | Point (10 10) | a Point |
| LineString | LineString (10 10, 20 20, 30 40) | a LineString with 3 points |
| Polygon | Polygon ((10 10, 10 20, 20 20, 20 15, 10 10)) | a Polygon with 1 exteriorRing and 0 interiorRings |
| Multipoint | MultiPoint ((10 10), (20 20)) | a MultiPoint with 2 points |
| MultiLineString | MultiLineString ((10 10, 20 20), (15 15, 30 15)) | a MultiLineString with 2 linestrings |
| MultiPolygon | MultiPolygon (((10 10, 10 20, 20 20, 20 15, 10 10)), ((60 60, 70 70, 80 60, 60 60))) | a MultiPolygon with 2 polygons |
| GeomCollection | GeometryCollection (POINT (10 10), POINT (30 30), LINESTRING (15 15, 20 20)) | a GeometryCollection consisting of 2 Point values and a LineString value |

Tab. 8: Beispiele für im WKT-Format dargestellte Geometrie-Objekttypen¹⁴⁷

¹⁴⁴ CHRISTL 2008a: 313.

¹⁴⁵ GREVE / KIEHLE 2006b: 29.

¹⁴⁶ Adams 2004: 51.

¹⁴⁷ Quelle: OGC 2006: 62.

2.3.3 Webserver

Anwendungen im Intra- und Internet basieren auf einem Client-Server-Konzept. Aufgabe von Webservern ist es, einem Clienten über das HTTP-Protokoll HTML-Seiten, CSS-Dateien und Bilddateien bereitzustellen. Werden dynamische Inhalte, wie z.B. dynamische Kartenbilder angefordert, leitet der Webserver den Request an einen Interpreter oder ein CGI-Script weiter. Somit stellt ein Webserver im oben beschriebenen Fall einer einfachen Kartenanwendung [vgl. S. 46] die Verbindung zwischen einem Clienten und einem Map-Server her, der im Script-Verzeichnis des Webserver abgelegt ist.

Apache HTTP Server

Am seismologischen Zentralobservatorium der BGR [vgl. Abschnitt 3.2] wird der Apache HTTP Server für die Bereitstellung des Webauftritts (www.seismologie.bgr.de) mit dynamischen Inhalten eingesetzt. Insofern liegt es nahe, den Apache Webserver auch im Rahmen der vorliegenden Arbeit zu verwenden. Weitere Gründe, die für den Einsatz des Apache sprechen sind

- die weite Verbreitung, mit 49,1 % Marktanteil ist der Apache derzeit der am häufigsten eingesetzte Webserver weltweit [vgl. Abb. 16],¹⁴⁸
- die hohe Stabilität bzw. Leistungsfähigkeit (kurze Antwortzeiten auch unter hoher Last)
- die Plattformunabhängigkeit (Unix, Linux, SunOS etc.) sowie
- die Möglichkeit des Einsatzes unter unterschiedlichen Hardwareumgebungen, ein kritischer Aspekt für die Sicherheit des gesamten Systems.

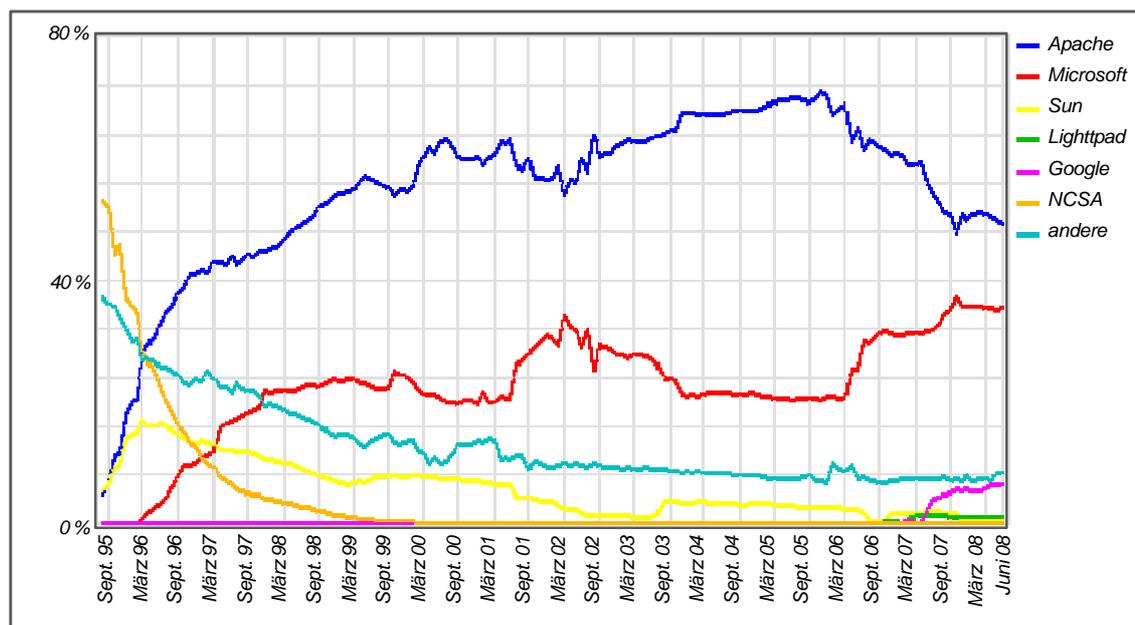


Abb. 16: Marktanteile der Top Webserver in allen Branchen (August 1995 bis Juni 2008)¹⁴⁹

¹⁴⁸ http://news.netcraft.com/archives/web_server_survey.html (Stand: 28.06.2008).

Der Apache Webserver ist in der Regel eine verhältnismäßig kleine ausführbare Datei und wird über eine zentrale Textdatei, der sog. `httpd.conf` konfiguriert [vgl. *Abschnitt 3.2.4*]. Je nach Konfiguration kann diese eine Vielzahl an Modulen mit zum Teil recht umfangreichen, weiteren Funktionen ansprechen.¹⁵⁰ Der modulare Aufbau des Apache gilt als eine seiner großen Stärken, da der Webserver so flexibel an eine Vielzahl von Plattformen sowie an unterschiedliche Umgebungen angepasst werden kann.¹⁵¹

Es gibt unterschiedliche Gruppen von Modulen. Eine übliche Gruppierung erfolgt entsprechend dem Status der Module, d.h. entsprechend der Art, wie sie kompiliert werden: Multiprocessing-Module (MPM), Kernmodule (Basismodule), Erweiterungsmodule, experimentelle und externe Module. Von besonderer Bedeutung sind die drei Kernmodule, die zur Laufzeit des Webserver immer zur Verfügung stehen:¹⁵²

- *core (nur Apache 2.x.x)*

Ermöglicht den Einsatz von Containern in der `httpd.conf` und damit das Ansprechen von Serververzeichnissen und Dateien, die Festlegung von `ServerRoot` und `DocumentRoot`, das Anlegen von Protokolldateien, das Ansprechen eines Ports, Aliasnamens etc. und somit die Festlegung zentraler Funktionen.

- *http_core*

Dieses Modul ist für die Lebensdauer von HTTP-Verbindungen zuständig (KeepAlive). (In Apache 1.3.x werden hier auch alle `core`-Anweisungen gehandelt.)

- *mod_so*

Der Zugriff auf andere Module, welche mit der `LoadModule`-Anweisung als DSO verwendet werden sollen, wird über `mod_so` ermöglicht.

2.3.4 MapServer

Für die Erzeugung von Kartenbildern und Vektordaten aber auch interaktiven Kartenanwendungen im Intra- und Internet wird auf Serverseite neben dem Webserver noch ein MapServer benötigt. Seine Hauptaufgabe besteht darin, Geodaten aus verschiedenen Quellen zu lesen und bestimmten Anforderungen entsprechend in einem Kartenbild zusammenzustellen und auszuliefern.

¹⁴⁹ Quelle: http://news.netcraft.com/archives/web_server_survey.html (Stand: 28.06.2008).

¹⁵⁰ <http://de.selfhtml.org/servercgi/server/apache.htm> (Stand 19.8.2008).

¹⁵¹ <http://httpd.apache.org/docs/2.2/mpm.html> (Stand 19.8.2008).

¹⁵² <http://de.selfhtml.org/servercgi/server/apache.htm> (Stand 19.8.2008).

UMN MapServer

Sehr verbreitet ist der UMN MapServer, wobei das Buchstabenkürzel UMN auf die University of Minnesota und die Ursprünge des Projekts verweist. Bei dem UMN MapServer handelt es sich um eine schnelle und sehr zuverlässige Software, die in nahezu jeder GIS-Umgebung zum Einsatz kommen kann.¹⁵³ Sie wird von einer großen Community verwendet und unterstützt. Auch in der Bundesanstalt für Geowissenschaften und Rohstoffe wird der UMN MapServer für den Aufbau von WMS und WFS eingesetzt.

Die zentrale Kommunikationsplattform für Entwickler und Anwender ist das UMN MapServer Portal unter <http://mapserver.gis.umn.edu/>. Über dieses Portal wird auch die Software (aktuell 5.2.0) zum Download zur Verfügung gestellt. Durch die Implementierung verschiedener OGC Standards – derzeit sind sowohl WMS (Client/Server) als auch Non-Transactional WFS (Client/Server) implementiert – und durch die Unterstützung einer Vielzahl an Datenformaten kann der UMN MapServer als Bindeglied in heterogenen Systemarchitekturen eingesetzt werden.¹⁵⁴ Im Wesentlichen besteht eine UMN MapServer-Anwendung aus den folgenden drei Komponenten:

- dem MapServer-Executable (CGI-Script),
- der MapServer-Mapdatei (zentrale Konfigurationsdatei – .map) sowie
- HTML-Templates.

Ihr Zusammenspiel ist in Abb. 17 dargestellt.

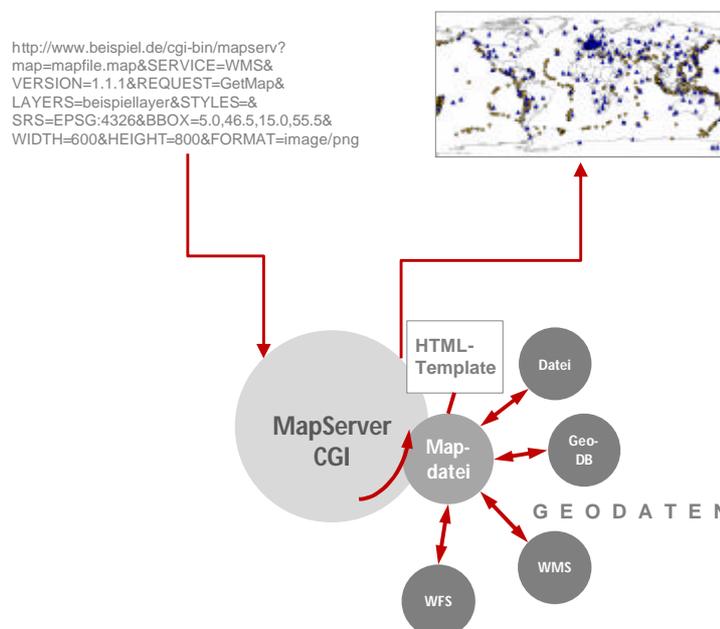


Abb. 17: Hauptbestandteile einer UMN MapServer-Anwendung¹⁵⁵

¹⁵³ <http://www.osgeo.org/mapserver> (Stand: 20.8.2008).

¹⁵⁴ <http://www.umn-mapserver.de/> (Stand: 30.06.2008).

¹⁵⁵ Quelle: CHRISTL 2008a: 46 (verändert).

Der UMN MapServer arbeitet als CGI-Script in der CGI-Schnittstelle des Webservers.¹⁵⁶ Das Script stellt eine Vermittlungsstelle zwischen dem Webserver, der die Karten anfordert, und den Geodaten selbst dar. Das Script wird über einen HTTP-Request in Form einer URL [vgl. Abb. 17 und Abb. 26] angesprochen, welche die vom Benutzer gewünschten Parameter der neuen Karte enthält. Entsprechend den definierten Parametern erstellt das CGI durch Einlesen einer Mapdatei die neuen Kartenbilder als temporäre Graphikdateien zusammen. Welche Mapdatei verwendet werden soll, muss ebenfalls in der URL angegeben sein. Bei der Mapdatei handelt es sich um die textbasierte Konfigurationsdatei einer MapServer-Anwendung [vgl. Abschnitt 3.2.5.1]. In ihr sind in strukturierter Form der Speicherort, das Speichermedium und die Art (Raster-/Vektordaten) der Geodaten angegeben sowie das Aussehen, der Inhalt und die Struktur des zu erstellenden Kartenbildes, aber auch die Interaktion mit diesem definiert. Ferner sind in der Mapdatei Verweise auf sog. HTML-Templates hinterlegt. Dabei handelt es sich um HTML-Seiten, die eine Reihe von Platzhaltern enthalten. Die Platzhalter werden durch die vom UMN MapServer erzeugten temporären Graphikdateien ersetzt. Die in die HTML-Templates eingebetteten Kartenbilder werden dann als HTTP-Response über den Webserver an den Client zurück gegeben. Die von Nutzer gewünschte Karte wurde „on-the-fly“ erstellt. Dieser Prozess wiederholt sich für jede neue Anfrage, d.h. z.B. für jedes hinein- oder heraus zoomen (etc.) in der Kartenanwendung.

2.3.5 Darstellungskomponente / Client / Web-Mapping-Client

Über eine Clientumgebung können die z.B. unter Anwendung des UMN MapServers, erstellten standardisierten OGC konformen Kartendienste angezeigt und in Webseiten integriert werden. Aus der Fülle von Softwareprojekten, die OGC konforme Kartendienste einbinden können, wurde im Rahmen der vorliegenden Arbeit die von der OSGeo graduierte Software OpenLayers ausgewählt.

*OpenLayers*¹⁵⁷

Bei OpenLayers handelt es sich um einen objektorientierten JavaScript basierenden Webmapping Client. Über eine Ajax-Bibliothek wird eine framefreie Integration von Kartendiensten in eine Webseite ermöglicht. Neben eigenen und fremden OGC konformen Kartendiensten – OpenLayers bietet Schnittstellen zu Standard-Formaten wie WMS, WFS, GML, KML etc. an – können auch Kartendienste proprietärer Anbieter wie z.B. Google Maps, MS Virtual Earth oder Yahoo! Maps eingebunden, kombiniert und angesprochen

¹⁵⁶ Neben seine Funktionsweise als CGI-Script (auch ein OGC-konformer MapServer funktioniert als CGI) kann die Funktionalität des UMN MapServers auch in einer Bibliothek zusammengefasst und von verschiedenen Programmiersprachen aus erschlossen werden (MapScript) (FISCHER 2003: 5).

¹⁵⁷ <http://openlayers.org/> (Stand: 30.06.2008).

werden. Dieses bietet aus wirtschaftlichen Gesichtspunkten den Vorteil, dass eine nahezu stufenlos zoombare Hintergrundkarte kostenlos¹⁵⁸ genutzt werden kann und somit die aufwendige Aufbereitung einer geeigneten Basistopographie entfällt. Darüber hinaus wird OpenLayers von einer sehr aktiven Community genutzt und entwickelt. Insofern werden Fehler in der Software schnell behoben, kontinuierlich neue Funktionen entwickelt und auftretende Schwierigkeiten können in der Regel mithilfe der User-Mailing-Liste innerhalb kurzer Zeit gelöst werden. Ein weiterer Aspekt, der bei der Entscheidung für OpenLayers von Bedeutung war, ist die bedientechnische Nähe [vgl. Abb. 30] zu vielen weiterverbreiteten Internetkartenanwendungen, so dass von einer intuitiven Bedienung durch den Nutzer auszugehen ist.

Als JavaScript-Client wird OpenLayers serverseitig konfiguriert und über einen Webserver bereit gestellt. Der Aufbau eines OpenLayers-Projekts wird in Abschnitt 3.2.6 vorgestellt.

2.3.6 Metadaten Management

Wie in Abschnitt 3 der vorliegenden Arbeit noch gezeigt wird, ist mit den bisher beschriebenen Komponenten eine digitale Bereitstellung und Nutzung von Geodaten- und Diensten im Intra- und Internet möglich. Damit die bereitgestellten Informationen vom Nutzer auch gefunden werden können, müssen die GDI-Komponenten über Katalogdienste (CSW) beschrieben werden, die wiederum über Metadatenkataloge bereitgestellt werden. Somit kommt dem Handling von Metadaten insbesondere im Umfeld von Geodateninfrastrukturen eine besondere Bedeutung zu.

Dieser – wenngleich wichtig Teil einer GDI – ist nicht Bestandteil der vorliegenden Arbeit, die im Rahmen eines Projektes im Fachbereich Seismologie der Gesellschaft für Geowissenschaften und Rohstoffe (BGR) in Hannover entstanden ist [vgl. Abschnitte 3.1 und 3.2]. Am Geozentrum Hannover (GZH), zu dem auch die BGR gehört, ist als ein erster Bestandteil der aufzubauenden Geofachdateninfrastruktur (GfDI-BGR) bereits ein OGC- und ISO19115/ISO19119-konformer Metadatenkatalog, unter Verwendung der proprietären Software g.business MDM der Firma GeoTast, realisiert worden. Hier werden Datenbestände und Dienste, wie von der GDI-DE vorgeschrieben und im Architekturkonzept GDI-DE festgehalten, mit ihren Metadaten nachgewiesen. Da es sich bei der am Geozentrum eingesetzten Software aber um ein proprietäres Produkt handelt, wird der Aufbau des Datenkataloges hier nicht weiter thematisiert.

¹⁵⁸ Die Google Maps API ermöglicht die Integration von Google Maps in die eigene Anwendung. Benötigt wird der so genannte API-Key, der kostenlos auf der Google Maps API-Homepage bezogen werden kann (<http://www.google.com/apis/maps/>).

3 Praxisbeispiel – Implementierung des Systementwurfs

3.1. Nutzerorientierte Bereitstellung von Geodaten bei der Bundesanstalt für Geowissenschaften und Rohstoffe

Bei der Bundesanstalt für Geowissenschaften und Rohstoffe (BGR) wird derzeit eine mittel- und langfristige Strategie zur Bereitstellung von Geofachdaten für eine breite Öffentlichkeit erarbeitet. Hintergrund für diese Bemühungen sind die in Abschnitt 1.2 der vorliegenden Arbeit beschriebenen Aktivitäten auf europäischer (INSPIRE) und deutscher (GDI-DE) Ebene.

Ziel ist es, die bei der BGR erarbeiteten Geofachdaten und Produkte aus Forschung und Entwicklung für externe und interne Nutzer besser verfügbar zu machen, um

- neue Nutzergruppen zu erschließen,
- eine effizientere Beratung gewährleisten zu können und
- nationale und internationale Verpflichtungen zur Datenbereitstellung zu erfüllen.¹⁵⁹

Darüberhinaus strebt die BGR die Integration nationaler Geofachdaten in internationalen Datenzentren und globalen Überwachungssystemen an.

Durch den Aufbau einer Geofachdateninfrastruktur der BGR (GfDi-BGR) soll auch BGR-intern eine weitreichende Datenintegration und -harmonisierung erzielt werden. Die Umsetzung der GfDi-BGR erfolgt mittelfristig in mehreren Arbeitsschritten und unter Berücksichtigung bereits bestehender GDI-Komponenten.

3.1.1 Rechtliche und politische Rahmenbedingungen – Auswirkungen für die Bundesanstalt für Geowissenschaften und Rohstoffe

Die rechtlichen Rahmenbedingungen auf europäischer und nationaler Ebene wurden in den Abschnitten 1.2.1 und 1.2.2 bereits dargestellt. Im Folgenden wird deren Bedeutung für die BGR kurz thematisiert.

3.1.1.1 *INSPIRE-Richtlinie*

Durch die INSPIRE-Richtlinie sollen per Gesetz Umweltinformationen, die in staatlichen Institutionen der europäischen Mitgliedstaaten gesammelt und aufbereitet werden, allgemein verfügbar gemacht werden. Der Schwerpunkt der aufzubauenden Geodateninfrastrukturen liegt auf der Informationsbereitstellung für den Umweltbereich und umfasst eine Viel-

¹⁵⁹ BGR, AG MoI 2008: 6.

zahl unterschiedlicher Fachgebiete. Für den durch die BGR auf nationaler und zum Teil internationaler Ebene vertretenen Bereich der Geowissenschaften sind dies die Themen

- *Geologie (Annex II der INSPIRE-Richtlinie)*
Geologische Beschreibung anhand von Zusammensetzung und Struktur
- *Boden (Annex III)*
Beschreibung von Boden und Unterboden
- *Gebiete mit naturbedingten Risiken (Annex III)*
Gefährdete Gebiete, eingestuft nach naturbedingten Risiken, z.B. Überschwemmungen, Erdbeben und Bodensenkungen, Lawinen, Waldbrände, Erdbeben oder Vulkanausbrüche
- *Energierohstoffe (Annex III)*
- *Mineralische Rohstoffe (Annex III)*

Für die genannten Themengruppen müssen Daten und Metadaten innerhalb eines strengen Zeitplans und entsprechend den noch im Erarbeitungsprozess befindlichen Durchführungsbestimmungen zur Datenspezifikation bereitgestellt werden:

- bis 2010 Metadaten für den Bereich Geologie,
- bis 2013 Metadaten zu den Bereichen Boden, natürliche Risikozonen, Energierohstoffe und mineralische Rohstoffe,
- bis 2014/2019 neu erhobene, restrukturierte bzw. aktuell verwendete thematische Geodaten zu den Themenbereichen Geologie, Boden, natürliche Risikozonen, Energierohstoffe und mineralische Rohstoffe.

Die zu diesen geowissenschaftlichen Themen erforderlichen Datenspezifikationen¹⁶⁰ müssen bis Mai 2011 erarbeitet werden. Als Mitglied des Entwurfsteams Datenspezifikationen („Data Specifications“) ist die BGR aktiv an der Erstellung der Umsetzungsrichtlinien beteiligt. Darüberhinaus wirkt sie (unter Einbeziehung der Länder) als Legally Mandated Organisation (LMO)¹⁶¹ am Review, der Kommentierung und der Modifikation aller Entwurfskriterien mit. Damit leistet die BGR einen Beitrag zur Erarbeitung der gesetzlichen Grundlagen für geowissenschaftliche raumbezogene Daten.¹⁶²

¹⁶⁰ Anforderungen an Qualität, Genauigkeit, Aktualität, Anwendungsschemata und Objektartenkatalog, Begriffe, Hierarchien etc.

¹⁶¹ Im Organisations- und Prozessmodell für die Erarbeitung der INSPIRE-Umsetzungsrichtlinien repräsentieren Legally Mandated Organisations auf Ebene der Mitgliedstaaten sowie auf EU-Ebene die Organisationen, die für die eigentliche Implementierung von INSPIRE verantwortlich sind sowie Nutzerinteressen und Ressourcen stellen [Bernhard 2006: 85].

¹⁶² BGR, AG MoI 2008: 10.

3.1.1.2 GDI-DE

Parallel zur Erarbeitung der Durchführungsbestimmungen muss für die Umsetzung der durch INSPIRE vorgegebenen Ziele ein Netzwerk von Diensten zur Bereitstellung der dezentral vorliegenden Daten sowie der dazugehörigen Metadaten aufgebaut werden. Dafür ist in Deutschland die GDI-DE verantwortlich. Als nachgeordnete Bundesbehörde des Bundesministerium für Wirtschaft (BMWI) und maßgeblich für Geodaten zuständige Einrichtung berät die BGR die Bundesressorts in allen georelevanten Fragestellungen. Über den IMAGI ist sie in die Prozesse zum Aufbau der GDI-DE (Fragen zur Architektur der GDI-DE, Beantwortung von INSPIRE-relevanter Fragestellungen) eingebunden.¹⁶³ Ferner ist die Geschäftsstelle der Kommission für Geoinformationswirtschaft des Bundeswirtschaftsministerium (GIW-Kommission) bei der BGR angesiedelt [vgl. Abb. 6].

3.1.2 Aufbau einer Geofachdateninfrastruktur bei der BGR

Bedingt durch die Vielfalt der oben genannten Annex-Themen [vgl. Abschnitt 3.1.1.1], sind beim Aufbau einer GfDI-BGR eine Vielzahl an fachlich zuständigen Organisationseinheiten einzubinden. Diese haben zum Teil bereits eigene Lösungen für die Bereitstellung ihrer Geodaten über das Internet und mithilfe von Webtechnologien erarbeitet. Auch sind die fachspezifischen Anforderungen an die Datenbereitstellung mitunter sehr heterogen, was z.B. die Aktualität der Daten, den Detaillierungsgrad der benötigten Hintergrundtopographie oder die angesprochenen Nutzergruppen¹⁶⁴ angeht. In der Praxis heißt das, dass sich alle Beteiligten auf eine gemeinsame Vorgehensweise beim Aufbau einer GfDI-BGR verständigen müssen, ein Prozess der sich nur mittelfristig realisieren lässt. Konsens besteht darin, dass der Aufbau einer GfDI-BGR unter Einbeziehung der bereits vorhandenen Komponenten erfolgen soll. Solange diese unter Anwendung der einschlägigen Standards und Normen (OGC, ISO) entwickelt wurden, werden sie sich in der Regel ohne großen zusätzlichen Aufwand zu einer gemeinsamen Lösung zusammenschließen lassen.

Beim Aufbau einer GDI-BGR ist also eine adaptionsfähige und dynamische Vorgehensweise gefragt, im Gegensatz zu einer statischen, bei der alle Softwarekomponenten im Vorfeld bekannt sind und verbindlich festgelegt werden. Im Rahmen einer solchen dynamischen Vorgehensweise werden alle Fragestellungen auf kleinere Unterprojekte verteilt und dort in einem kleinem Kreis von Beteiligten gelöst.“¹⁶⁵

¹⁶³ BGR, AG MoI 2008: 9.

¹⁶⁴ Die wichtigsten Nutzergruppen sind Universitäten und Forschungseinrichtungen, die Staatlichen Geologischen Dienste (SGDs) sowie die allgemeine Öffentlichkeit. Auf ministerieller Ebene ist insbesondere das BMWI vertreten.

¹⁶⁵ ADAMS et al. 2004: 16.

Komponenten einer GfDI-BGR – eine Bestandsaufnahme

Der erste Schritt beim Aufbau einer GfDI-BGR war die Bestandsaufnahme aller bereits bestehenden Komponenten, die zukünftig über ein zentrales Geodatenportal bereitgestellt werden sollen. Um auch der Öffentlichkeit bereits zum jetzigen Zeitpunkt einen Überblick über die bei der BGR vorhandenen Geoinformationen zu vermitteln, wurden sie unter einem gemeinsamen Navigationspunkt ‚Geoinformationen‘ im BGR-Internetauftritt zusammengefasst. Im Wesentlichen sind dies die nachfolgend vorgestellten Komponenten, ihre Beschreibungen wurden den Unternavigationspunkten der Rubrik ‚Geoinformationen‘¹⁶⁶ entnommen:

- *Der Geodatenkatalog – GeoDaK*

Über den GeoDaK können Informationen (Metadaten) zu den Datenbeständen der verschiedenen Fachbereiche der BGR bzw. des Geozentrums Hannover¹⁶⁷ recherchiert werden. Dazu gehören z.B. Karten und Fachdatenbestände aus den Bereichen Geologie, Geophysik, Geochemie, Hydrogeologie, Ingenieurgeologie, Seismologie und Bodenkunde. Der GeoDaK die durch das Produkt ‚g.business MDM‘ der Firma GeoTask bereitgestellt.

- *Fachanwendungen*

Die Fachbereiche *Bodenkunde*, *Geologie* und *Hydrogeologie* haben jeweils an ihre Belange angepasste Fachanwendungen entwickelt, die über das Internet verfügbar sind.

Bodenkunde: Das Fachinformationssystem *Bodenkunde* (FISBo) der BGR enthält eine große Bandbreite bundesweit abgestimmter Informationen zum Thema Boden. Für die Dokumentation und interaktive Präsentation des derzeitigen Arbeitsstandes in Form von Karten und Datenbankauszügen ist der ‚Bodenkundliche Kartenserver‘ entwickelt worden.

Geologie: Die BGR hat unter der Schirmherrschaft der Weltkartenkommission (CGMW) und in Zusammenarbeit mit den Geologischen Diensten von insgesamt 50 Ländern in den letzten Jahren eine Internationale Geologische Karte von Europa und angrenzender Gebiete im Maßstab 1 : 5 000 000 (IGME 5000) erarbeitet.

Grundwasser: Das Fachinformationssystem *Hydrogeologie* (FIS Hy) der BGR hat die Aufgabe, analoge und digitale Grundwasserdaten aus nationalen und internationalen BGR-Projekten an zentraler Stelle zusammen zu führen, zu verarbeiten und bereit zu

¹⁶⁶ http://www.bgr.bund.de/nn_322882/DE/Themen/Geoinformationen/geoinformationen__node.html?__nnn=true (Stand: 15.08.2008, ohne Zeilenumbruch).

¹⁶⁷ Im Geozentrum Hannover haben sich die folgenden drei geowissenschaftlichen Einrichtungen zusammengeschlossen: die Bundesanstalt für Geowissenschaften und Rohstoffe (BGR), das Landesamt für Bergbau, Energie und Geologie (LBEG) und das Institut für Geowissenschaftliche Gemeinschaftsaufgaben (GGA).

stellen. Die hydrogeologischen Karten werden als Internet-Kartenanwendungen und Webdienste bereitgestellt.

Seismologie: Im Zuge der vorliegenden Arbeit wurde eine Anwendung für den Fachbereich Seismologie entwickelt. Diese wird in Abschnitt 3.2 detailliert vorgestellt.

- *Webdienste*

Die Geofachdaten der BGR werden auch in Form von Webdiensten bereitgestellt. Zurzeit sind verschiedene Web Map Services (WMS) sowie ein Catalogue Service for the Web (CSW) realisiert.

- *GeoViewer*

Zukünftig sollen die in den einzelnen Fachanwendungen bereitgestellten Inhalte sowie die verschiedenen Web-Dienste über eine gemeinsame Viewer-Lösung angeboten werden. Dabei soll dem Nutzer zwecks Übersichtlichkeit die Möglichkeit gegeben werden, verschiedene Themenschwerpunkte auszuwählen. Eine solche integrierte Lösung kann jedoch, bedingt durch die heterogenen fachlichen Anforderungen der Fachbereiche, nicht kurzfristig realisiert werden. Deswegen wurde zunächst für einen befristeten Zeitraum auf eine im Rahmen des GIW-Leitprojektes ‚Rohstoffe‘ entwickelte und der BGR von der GIW-Kommission zur Verfügung gestellt WebGIS-Anwendung zurückgegriffen. Diese wurde an die zuvor gemeinsam definierten Minimalanforderungen der BGR angepasst und unter der Bezeichnung ‚GeoViewer‘ veröffentlicht. Das Konzept für eine umfassende fachbereichsübergreifende Lösung wird derzeit erarbeitet. Der GeoViewer basiert auf dem Mapbender Framework (OS/FS). Denkbar ist, dass auch der zukünftig zu entwickelnde Viewer auf Basis von Mapbender konzipiert wird.

Das Gesamtkonzept zum Aufbau und Betrieb einer GfDI-BGR wird derzeit erarbeitet und soll bis Ende 2008 vorliegen.¹⁶⁸

3.2. Fallbeispiel – Bereitstellung von seismischen Daten am seismologischen Zentralobservatorium der BGR (SDAC)

Erdbeben werden in Deutschland und weltweit mithilfe eines Netzwerks von seismologischen Messstationen kontinuierlich überwacht. In Deutschland wird diese Aufgabe vom seismologischen Zentralobservatorium (Seismic Data Analysis Center – SDAC) der BGR wahrgenommen. Vor allem bei größeren seismischen Ereignissen wird diesem Monitoring eine große Aufmerksamkeit seitens der Medien aber auch von Betroffenen und einer interessierten Öffentlichkeit zuteil. Vor diesem Hintergrund sowie den bereits beschriebenen politischen und rechtlichen Rahmenbedingungen [vgl. Abschnitte 1.2 und 3.1.1] kommt der

¹⁶⁸ BGR, AG MoI 2008: 33

Bereitstellung von seismischen Daten innerhalb einer GfDI-BGR eine besondere Bedeutung zu.

Der Aufbau einer GfDI-BGR ist jedoch, bedingt durch die große Anzahl der beteiligten Fachbereiche, nur mittel- bis langfristig möglich. Deswegen wird durch das SDAC eine Fachanwendung aufgebaut, die wesentliche Komponenten einer GDI enthält und deren Elemente zu einem späteren Zeitraum in eine GfDI-BGR integriert werden können. Bei der Bereitstellung von seismischen Daten werden im Wesentlichen folgende Ziele verfolgt:

- Visualisierung von aktuellen seismischen Ereignissen im Internet;
- Bereitstellung eines einfachen Zugangs für einen breiten Nutzerkreis;
- Schaffung von Mehrwerten für den Nutzer durch eine Präsentation in einem fachlichen Kontext (vertiefende Hintergrundthemen und -informationen) sowie eine interaktive Anwendung (Abfragemöglichkeit zu einzelnen Ereignissen);
- OGC-konforme Realisierung, um eine spätere Integration in eine GfDI-BGR zu ermöglichen.

3.2.1 Identifizierung von Nutzergruppen

Im Vorfeld des Systementwurfs und der Implementierung wurde eine Identifizierung und Analyse der potenziellen Nutzergruppen durchgeführt. Dazu wurden die im Zeitraum von März bis einschließlich November 2007 eingegangenen E-Mails und telefonischen Anfragen analysiert und ausgewertet. Auf dieser Basis konnten folgende Nutzergruppen identifiziert werden, die entsprechend der Häufigkeit ihrer Anfrage sortiert sind:

- Betroffene
- interessierte Laien
- Universitäten / Forschung
- Medien (Fernsehen, Printmedien)
- Schulen und Schulbuchverlage
- Ingenieurbüros / Consulting
- andere Behörden (Archäologie, Bergbau, ...)
- Sonstige

Die Nutzeranalyse hat gezeigt, dass Informationen zu Erdbeben für eine breite Öffentlichkeit von Interesse sind, jedoch größtenteils keine fachspezifischen Kenntnisse vorausgesetzt werden können. Insofern muss der Systementwurf vor allem eine einfache und intuitive Nutzung ermöglichen [vgl. *Abschnitt 2.3.5*].

3.2.2 Einsatz von OS/FS-Produkten am SDAC

Der Ansatz, die GDI größtenteils bzw. ausschließlich durch dem Einsatz von OS/FS-Produkten aufzubauen, ist neben den in Abschnitt 2.2.1 beschriebenen Vorteilen, auch durch die positiven Erfahrungen mit anderen Softwareprodukten aus dem OS/FS-Bereich zu begründen. Die Grundlage für den Einsatz von OS/FS-Produkten wurde 2007 mit der Umstellung der am SDAC betriebenen Hardware auf UNIX-artige Betriebssysteme geschaffen. Seitdem werden verstärkt OS/FS-Produkte eingesetzt, u.a. wurden bereits Erfahrungen mit dem Apache Webserver und dem UMN MapServer gesammelt.

Entsprechend der Aufgabenstellung dieser Master Thesis wurden für den Aufbau der Geodateninfrastruktur ausschließlich OS/FS-Produkte eingesetzt. Bevor in den folgenden Abschnitten die eingesetzte Software und deren Konfiguration näher erläutert werden, gibt Tab. 9 einen ersten Gesamtüberblick:

| <i>Systemkomponente</i> | <i>Software</i> |
|-------------------------|--|
| Betriebssystem: | SunOS / Solaris |
| Webserver: | Apache HTTP Server Project |
| Datenbank: | PostgreSQL / PostGIS |
| Kartenserver: | UMN MapServer (über WMS Schnittstelle) |
| Kartenclient: | OpenLayers |

Tab. 9: Eingesetzte Software

3.2.3 Die Datenbasis

3.2.3.1 *Der Datenbestand am SDAC*

Ein Netz von hochempfindlichen Seismometerstationen registriert kontinuierlich Bodenbewegungen, zeichnet diese in digitaler Form auf und übermittelt die Daten fortlaufend an das Datenzentrum der BGR. Hier kommen neben den Daten der von der BGR (SDAC) betriebenen Stationen auch die von anderen seismologischen Einrichtungen in Deutschland sowie von den nationalen Datenzentren anderer Staaten (ISC¹⁶⁹, USGS¹⁷⁰) zusammen und werden in einer Datenbank bzw. auf Fileservern archiviert.

¹⁶⁹ International Seismological Centre; <http://www.isc.ac.uk/> (Stand: 24.08.2008).

¹⁷⁰ United States Geological Survey, Geologischer Dienst der Vereinigten Staaten; <http://earthquake.usgs.gov/> (Stand: 24.08.2008).

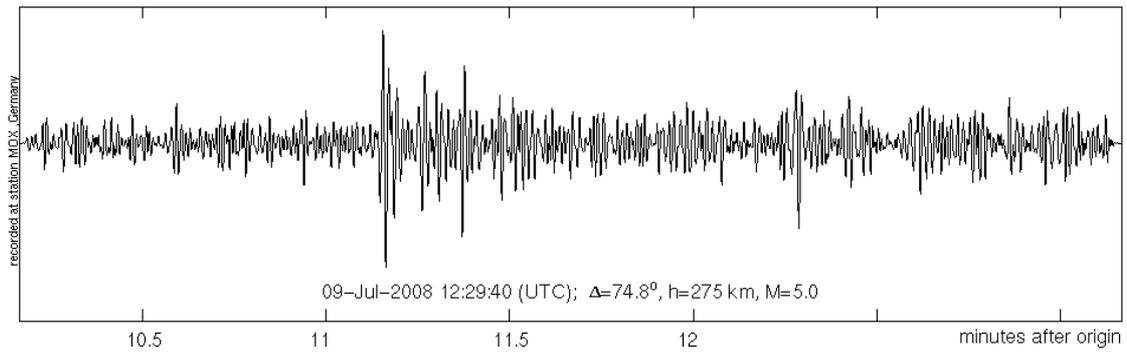


Abb. 18: Seismogramm eines Erdbebens

Die binären Rohdaten (seismische Wellenformdaten, vgl. Abb. 18) werden in der BGR ausgewertet. Die daraus abgeleiteten seismischen Parameterdaten werden zunächst in einen vorläufigen monatlichen Datenkatalog und später in einem endgültigen Jahreskatalog kompiliert und veröffentlicht. Diese Datenkataloge beinhalten in tabellarischer Form die ermittelten Herd- und Stationsparameter zu einzelnen Erdbebenereignissen. Einen Ausschnitt aus einem Datenkatalog zeigt Abb. 19:

| 1. Event 80317301 Zug, Schwyz/Ch, Zuger See ← ausgewertetes Ereignis | | | | | | | | | | | | | | | | | | |
|---|-------------|-------|--------|--------------|-----------|-------|------|------|-------|-----|------|------|------|-----------|-------|---------|--------|----------|
| Date | Time | Err | RMS | Latitude | Longitude | Smaj | Smin | Az | Depth | Err | Ndef | Nsta | Gap | mdist | Mdist | Qual | Author | OrigID |
| 2008/03/17 | 00:08:23.10 | 0.05 | 0.47 | 47.0286 | 8.6423 | 1.0 | 0.7 | 105 | 10.0 | 0.0 | 76 | 24 | 066 | | | m i k e | LDG | 80317321 |
| 2008/03/17 | 00:08:25.29 | 1.71 | | 47.1520 | 8.6210 | 16.7 | 7.8 | 020 | 10.0f | | 6 | 4 | 291 | 1.19 | 2.67 | m i k e | BGR | 80317311 |
| 2008/03/17 | 00:08:22.70 | | | 47.0400 | 8.6600 | | | | 7.0 | | | | | | | m i u k | SED | 80317301 |
| (#PRIME) | | | | | | | | | | | | | | | | | | |
| Magnitude | Err | Nsta | Author | OrigID | | | | | | | | | | | | | | |
| MD | 2.5 | 0.1 | 3 | LDG | 80317321 | | | | | | | | | | | | | |
| ML | 2.4 | 0.3 | 24 | LDG | 80317321 | | | | | | | | | | | | | |
| ML | 2.3 | | | BGR | 80317311 | | | | | | | | | | | | | |
| ML | 2.1 | | | SED | 80317301 | | | | | | | | | | | | | |
| 2. Sta | | | | | | | | | | | | | | | | | | |
| Dist | EvAz | Phase | Time | TRes | Azim | AzRes | Slow | SRes | Def | SNR | Amp | Per | Qual | Magnitude | | | | |
| BFO | 1.31 | 350.3 | Pg | 00:08:47.721 | 0.3 | | | | | | | | m_e | ML | 2.1 | | | |
| BFO | 1.31 | 350.3 | Sg | 00:09:03.417 | -0.9 | | | | | | | | m_e | | | | | |
| STU | 1.77 | 11.5 | Sg | 00:09:18.975 | 0.1 | | | | | | | | m_e | ML | 2.2 | | | |
| FUR | 2.09 | 56.6 | Sg | 00:09:30.826 | 1.7 | | | | | | | | m_e | ML | 2.6 | | | |
| GRCL | 2.74 | 43.3 | Pg | 00:09:14.285 | 0.0 | | | | | | | | m_e | ML | 2.5 | | | |
| GRCL | 2.74 | 43.3 | Sg | 00:09:50.019 | 0.4 | | | | | | | | m_e | | | | | |

1. – Herdparameter
2. – Stationsparameter

Abb. 19: Ausschnitt aus dem vorläufigen monatlichen Datenkatalog vom März 2008

3.2.3.2 Datenauswahl

Die Datenbasis bildet den Kern der Implementierung. Insofern kommt der Datenauswahl eine hohe Bedeutung zu. Hinsichtlich der für die Anwendung relevanten Daten wird zwischen drei Datenebenen unterschieden:

- Seismische Fachdaten
- Ergänzende Fachdaten
- Basistopographie

Seismische Fachdaten

Den Kerninhalt der Anwendung bilden die seismischen Fachdaten, genauer die aus den Rohdaten abgeleiteten seismischen Parameterdaten [vgl. Abb. 19]. Entsprechend der identifizierten Zielgruppe [vgl. Abschnitt 3.2.1] soll nur eine Untermenge der vorhandenen Parameterdaten über einen OWS bereitgestellt werden. Konkret sind das Informationen zur

geographischen Lage (Länge/Breite), Ort / Region, Datum, Uhrzeit und Magnitude. Bei besonderen Ereignissen, in deren Folge eine besonders hohe Energie freigesetzt wurde oder es viele Opfer gab, soll auch ein Kommentar bereitgestellt werden.

Ergänzende Fachdaten

Ergänzend zu den Kerninhalten sind eine Vielzahl weiterer Informationen denkbar, die zum Teil aus verwandten Disziplinen stammen. Bis auf die Daten zu den Seismometerstationen ist die Bereitstellung allerdings erst mit späteren Ausbaustufen der Anwendung vorgesehen.

- *Punkt- und Liniendaten:*
Seismometerstationen, Kernexplosionen, Katastrophenbeben, Störungslinien, Plattengrenzen, seismo-tektonische Regionen
- *Flächendaten:*
Geologie, Bevölkerungsdichte, Erdbebengefährdungsgebiete

Basistopographie

Als Orientierungsebene kommt der Basistopographie eine grundlegende Bedeutung zu. Um eine leichte Orientierung zu ermöglichen, müssen relevante Städte, administrative Grenzen, Gewässer, Verkehrsinfrastruktur sowie möglichst auch ein Relief gegeben sein.

Eine weltweit verfügbare Basistopographie bietet die kommerzielle Anwendung Google Maps. Auch wenn deren Integration [vgl. Abschnitt 2.3.5] in die aufzubauende (vorläufige) GDI im Gegensatz zum Ansatz der ausschließlichen Verwendung von OS/FS-Produkten steht, überwiegen an dieser Stelle praktische Überlegungen. Bei der im Rahmen der Arbeit entwickelten GDI handelt es sich um eine vorläufige Lösung, für die die Erarbeitung einer eigenen, in verschiedenen Zoomstufen vorliegenden, Hintergrundtopographie nicht wirtschaftlich ist. Die genauen Rahmenbedingungen einer GfDI-BGR werden derzeit in verschiedenen Arbeitsgruppen der BGR erarbeitet. Eines der zu klärenden Themen ist auch die Bereitstellung einer einheitlichen Basistopographie.

3.2.3.3 Datenhaltung in einer PostGIS Datenbank

Bei den seismischen Parameterdaten (Kerndaten) handelt es sich um einfache Punktdaten, deren Lokation über geographische Koordinaten beschrieben wird und denen Attributinformationen zugeordnet sind. Bisher wurden sie in einer relationalen Open-Source Datenbank verwaltet, die jedoch keine Unterstützung räumlicher Objekte bot. Deshalb wurde für die Bereitstellung der Erdbebendaten über einen OWS eine objekt-relationale PostGIS Datenbank [vgl. Abschnitt 2.3.2] aufgesetzt, welche über eine Schnittstelle zum UMN Map-Server verfügt [vgl. Code 4].

Wie in Abschnitt 2.3.2 erläutert, unterstützt PostGIS die OGC Spezifikation Simple Features for SQL. Diese enthält ein Schema für die Speicherung von räumlichen Daten, welches auch von PostGIS verwendet wird und in Abb. 20 dargestellt ist.

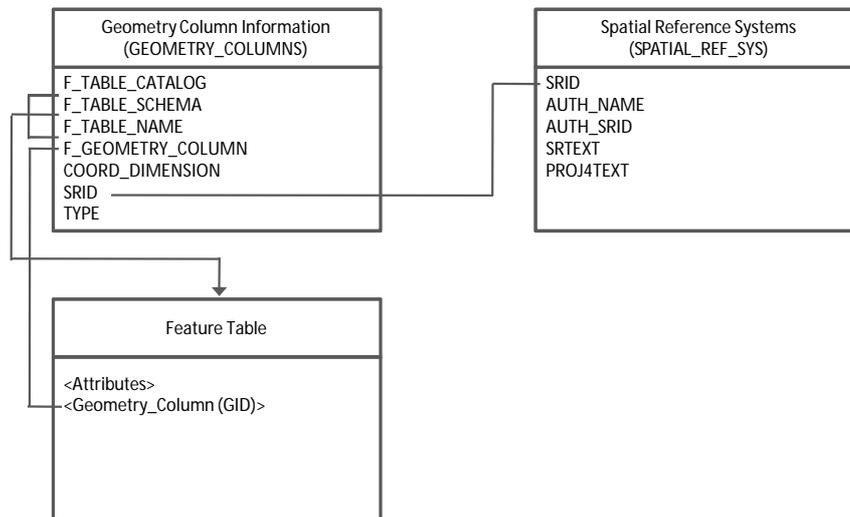


Abb. 20: Speicherschema für die Geodatenhaltung in PostGIS¹⁷¹

In ihrer Grundstruktur besteht eine PostGIS Datenbank somit aus (mindestens) drei Tabellen, den Metadatatabelle, 'GEOMETRY_COLUMNS' und 'SPATIAL_REF_SYS' sowie den eigentlichen Datentabellen.

- *Tabelle 'SPATIAL_REF_SYS':*

In der Tabelle 'SPATIAL_REF_SYS' werden Informationen über Koordinatenreferenzsysteme und Projektionen gespeichert. Der Inhalt der Tabelle 'SPATIAL_REF_SYS' wird beim Einrichten der Datenbank über ein SQL-Script (spatial_ref_sys.sql) geladen.

- *Tabelle 'GEOMETRY_COLUMNS':*

Die Tabelle 'GEOMETRY_COLUMNS' enthält für jeden 'Feature Table' einen Eintrag. Gespeichert werden der Name des Schemas,¹⁷² der Tabellename, der Name der Geometriespalte, das verwendete Koordinatenreferenzsystem, der Geometrietyp und die Dimension der Daten.

- *Feature Table*

In der hier als Feature Table bezeichneten Tabelle erfolgt die Speicherung der Attributinformationen zu einem räumlichen Objekt. Des Weiteren wird hier auch eine Geometriespalte angelegt, in der die jeweiligen Geometrieinformationen festgehalten werden. Innerhalb der Geometriespalte müssen entweder alle Features vom selben Geometrie-

¹⁷¹ Quelle: OGC 2006: S. 21 (verändert).

¹⁷² Eine Datenbank kann 1 bis n Schemata enthalten, welche der Strukturierung in der Datenbank dienen.

typ sein (z.B. ‚POINT‘) oder es muss der Geometrietyp ‚GEOMETRY‘ verwendet werden.

Feature Tables werden in zwei Schritten erzeugt. Zunächst wird eine Tabelle über den SQL-Befehl *CREATE TABLE* erstellt. Dann wird die Geometriespalte über die Funktion *AddGeometryColumn* hinzugefügt:

```
AddGeometryColumn
(<schema_name>, <table_name>, <column_name>, <srid>, <type>, <dimension>)
```

Die im Zuge der vorliegenden Arbeit aufgebaute Datenbankstruktur ist in Abb. 21 dargestellt. Entsprechend der definierten Zielgruppen [vgl. Abschnitt 3.2.1] wird über den OWS zunächst nur eine Untermenge der insgesamt beim SDAC vorhandenen Daten bereitgestellt. Neben den beiden Metadatatabelle wurden zwei weitere Tabellen für die seismischen Parametradaten und die Daten zu den Seismometerstationen angelegt. In beiden Fällen handelt es sich um Punktgeometrien. Eine Bereitstellung weiterer Daten gemäß Abb. 21 ist geplant. Entsprechend der Prämisse höchster Datenaktualität ist eine kontinuierliche Aktualisierung der Datenbank erforderlich.

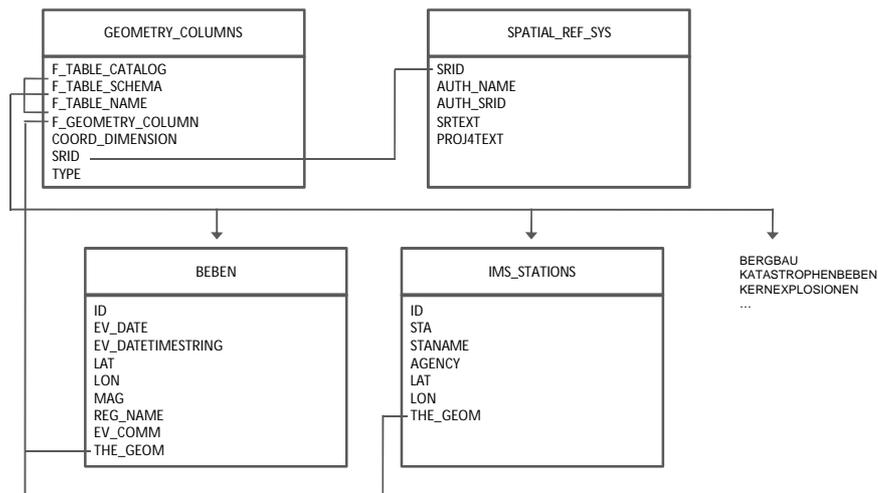


Abb. 21: Datenbankstruktur von „seis_ows“

Tab. 10 zeigt einen Ausschnitt aus der Tabelle „beben“ der Datenbank „seis_ows“. Damit die Geometrien in einer für den Betrachter lesbaren Form ausgegeben werden, wird für die Datenbankabfrage die Funktion *ST_AseWKT* verwendet.

```
seis_ows=# select id, ev_date, ev_datetimestring, lat, lon, mag, reg_name, ST_AseWKT(the_geom) from beben;
```

| id | ev_date | ev_datetimestring | lat | lon | mag | reg_name | st_asewkt |
|-------|------------|-------------------------|--------|--------|-----|---|--------------------------------|
| 33653 | 2007-10-19 | 2007-10-19 09:23:23.800 | -44.82 | 167.38 | 3.8 | South Island, New Zealand | SRID=4326:POINT(167.38 -44.82) |
| 33665 | 2007-10-19 | 2007-10-19 16:06:31.700 | -40 | 176.88 | 3.9 | North Island, New Zealand | SRID=4326:POINT(176.88 -40) |
| 33686 | 2007-10-19 | 2007-10-19 23:36:51.200 | -36.57 | 177.71 | 4.3 | Off east coast of North Island, New Zealand | SRID=4326:POINT(177.71 -36.57) |
| 33789 | 2007-10-21 | 2007-10-21 16:05:20.100 | -37.44 | 176.2 | 4.5 | North Island, New Zealand | SRID=4326:POINT(176.2 -37.44) |
| 33847 | 2007-10-22 | 2007-10-22 16:27:40.200 | -39.29 | 173.57 | 4.1 | Off west coast of North Island, New Zealand | SRID=4326:POINT(173.57 -39.29) |
| 33848 | 2007-10-22 | 2007-10-22 16:35:39.800 | -40.69 | 173.8 | 4.8 | Cook Strait, New Zealand | SRID=4326:POINT(173.8 -40.69) |
| 33871 | 2007-10-23 | 2007-10-23 05:11:48.300 | -44.81 | 167.36 | 4 | South Island, New Zealand | SRID=4326:POINT(167.36 -44.81) |
| 33892 | 2007-10-23 | 2007-10-23 14:19:39.500 | -47.97 | 99.15 | 4 | Southeast Indian Ridge | SRID=4326:POINT(99.15 -47.97) |
| 33896 | 2007-10-23 | 2007-10-23 17:35:29.400 | -36.49 | 177.18 | 4 | Off east coast of North Island, New Zealand | SRID=4326:POINT(177.18 -36.49) |
| 33914 | 2007-10-24 | 2007-10-24 01:12:30.900 | -42.17 | 172.04 | 5.1 | South Island, New Zealand | SRID=4326:POINT(172.04 -42.17) |
| 33941 | 2007-10-24 | 2007-10-24 09:29:30.300 | -37.64 | 177.21 | 3.9 | Off east coast of North Island, New Zealand | SRID=4326:POINT(177.21 -37.64) |
| 33945 | 2007-10-24 | 2007-10-24 11:09:39.200 | -44.88 | 167.2 | 4.5 | South Island, New Zealand | SRID=4326:POINT(167.2 -44.88) |
| 33946 | 2007-10-24 | 2007-10-24 12:02:44.100 | -40.51 | 175.96 | 3.9 | North Island, New Zealand | SRID=4326:POINT(175.96 -40.51) |

Tab. 10: Ausschnitt aus der Tabelle „beben“ der Datenbank PostGIS- „seis_ows“ (seismische Parameterdaten)

3.2.4 Serverseitige Implementierung I – der Webserver

Wie in Abschnitt 2.3.3 erwähnt, erfolgt die Konfiguration des Apache http Servers über die Textdatei httpd.conf. In der Regel ist der Apache bereits vorkonfiguriert, so dass nur geringfügige Anpassungen notwendig werden. Im Folgenden werden einige wichtige Einstellungen angesprochen.

- *Die Portnummer*

Üblicherweise wird der Apache als Stand-alone-Version konfiguriert, damit der WWW-Daemon nicht bei jedem Zugriff erneut die Konfiguration einlesen muss, sondern nur bei einem Neustart. In diesem Fall muss eine Portnummer angegeben werden. Voreingestellt ist der Port 80. In der Testphase ist jedoch eine abweichende Portnummer zu empfehlen, z.B. 8080, damit kein Internetnutzer unbeabsichtigt auf die Testversion zugreift (Beispiel: <http://www.seismologie.bgr.de:8080/>).

- *Das DocumentRoot Verzeichnis*

Hierbei handelt es sich um das Verzeichnis, das der Webserver für Zugriffe über das Inter- und Intranet freigibt. In diesem Verzeichnis wird z.B. ein temp-Verzeichnis erstellt, in dem der Mapserver die bereitzustellenden Kartenbilder ablegt. Auch werden hier u.a. die für die OpenLayers-Anwendung benötigten Javascript-Bibliotheken abgelegt.

- *Das Script (oder CGI-) Verzeichnis*

Hier dürfen Programme, wie z.B. der Mapserver ausgeführt werden. Es ist frei wählbar, muss aber als Script-Verzeichnis definiert werden.

- *Virtuelle Verzeichnisse*

Ergänzend zum DocumentRoot Verzeichnis können weitere, virtuelle Verzeichnisse für den Zugriff aus dem Internet freigegeben werden. Sie sind wählbar, müssen aber explizit definiert werden.

3.2.5 Serverseitige Implementierung II – der UMN MapServer

Die Implementierung des OGC-konformen Web-Dienstes erfolgt unter Nutzung des UMN MapServers. Dieser wurde im Vorfeld mit folgenden Einstellungen konfiguriert:

```
MapServer version 5.0.2 OUTPUT=GIF OUTPUT=PNG OUTPUT=JPEG OUTPUT=WBMP OUTPUT=SVG
SUPPORTS=PROJ SUPPORTS=FREETYPE SUPPORTS=WMS_SERVER SUPPORTS=WMS_CLIENT
SUPPORTS=WFS_SERVER SUPPORTS=WFS_CLIENT SUPPORTS=WCS_SERVER SUPPORTS=THREADS
INPUT=TIFF INPUT=EPPL7 INPUT=POSTGIS INPUT=OGR INPUT=GDAL INPUT=SHAPEFILE
```

Abb. 22: UMN MapServer Version und unterstützte Daten- und Ausgabeformate

Die Lauffähigkeit der UMN MapServer-Installation kann durch den Aufruf des `mapserv-Executables` im Browser getestet werden:

```
http://localhost/cgi-bin/mapserv
```

Als Antwort sollte der UMN MapServer folgenden String zurückliefern:

```
No query information to decode. QUERY_STRING is set, but empty.
```

Die Meldung zeigt, dass der UMN MapServer erfolgreich installiert wurde und nun einsatzbereit ist.

3.2.5.1 Konfiguration des UMN MapServers

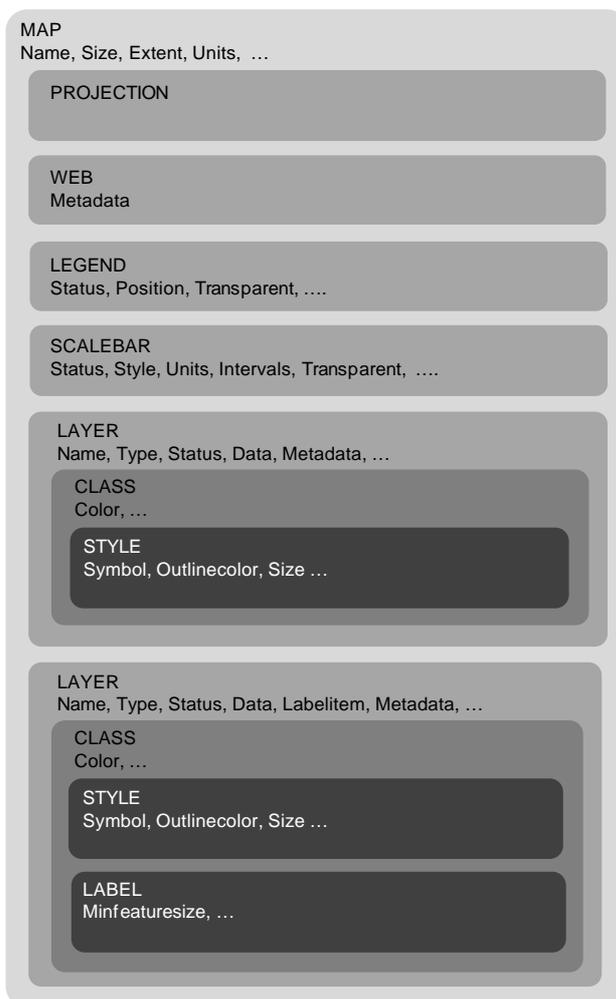


Abb. 23: Die Mapdatei – wesentliche Objekte und dazugehörige Parameter (Auswahl)¹⁷³

Wie bereits in Abschnitt 1.3.4.2 angesprochen, wird eine UMN MapServer-Anwendung über die sogenannte Mapdatei (`.map`) konfiguriert. Hier werden Angaben zur Darstellung der auszuliefernden Karte, ihren Inhalten sowie zu weiteren Kartenelementen wie z.B. Kartenmaßstab, Legende oder Übersichtskarte sowie zu implementierten OGC-Funktionalitäten getroffen. Bei einer Mapdatei handelt es sich um eine Textdatei, die sich aus verschiedenen

¹⁷³ Quelle: CHRISTL et al. 2008: 173 (erweitert).

hierarchisch verschachtelten Blöcken (Objekten) zusammensetzt. Jeder Block beginnt mit einem Schlüsselwort und wird mit einem END abgeschlossen. Einzig der oberste Block, der sog. Header, beginnt ohne Schlüsselwort. Innerhalb eines Blocks erfolgt die Deklaration von Schlüsselwörtern mit dazugehörigen Werten. Die Blöcke selbst können weitere Blöcke enthalten. Abb. 23 zeigt den grundlegenden Aufbau einer Mapdatei:

Im Zuge der vorliegenden Arbeit ist unter Nutzung des UMN MapServers ein OGC-konformer WMS aufgesetzt worden. Nachfolgend werden wesentliche Blöcke der Mapdatei herausgegriffen und erläutert. Entsprechend der Notation des UMN MapServers werden erläuternde Kommentare durch das #-Zeichen vom eigentlichen Code, d.h. den Schlüsselwörtern mit den ihnen zugewiesenen Werten, abgetrennt.

Der Header

Den obersten Abschnitt der Mapdatei bildet der sogenannte Header [vgl. Code 1]. Hier werden für das gesamte Projekt gültige Angaben getroffen, die das Erscheinungsbild der Karte bestimmen.

```
MAP                                # Beginn des Map-Objekts (optional)
  NAME Erdbebenkatalog            # Kartennamen bei OWS-konformen Aufrufen bzw.
# für die Benennung temporär erzeugter Dateien
  SIZE 600 300                    # Breite und Höhe der Kartengrafik in Pixel
  EXTENT -180 -90 180 90          # Darstellungsbereich (Bounding Box)
  UNITS DD                        # Karteneinheiten (hier: Dezimalgrad)
  STATUS ON                       # Status der Mapdatei (ON, OFF)
  RESOLUTION 96                   # Pixel/Inch für Ausgabe (Maßstabsberechnung)
  IMAGECOLOR -1 -1 -1             # Hintergrundfarbe in GRB (hier: transparent)
  IMAGETYPE gif                   # Format der Bildausgabe
  FONTSET "/home/mapserv_bin/fonts/fonts.fnt" # Schriftendatei
  SYMBOLSET "/home/mapserv_bin/symbset.sym"   # Symboldatei
[...]
```

```
END                                # Ende des Map-Objekts
```

Code 1: Der Header der Mapdatei

Weitere Objekte des Headers

Der Header kann um weitere Blöcke ergänzt werden, die ebenfalls für das gesamte Projekt gültig sind (z.B. Legend, Querymap, Scalebar). Der folgende Ausschnitt aus der Mapdatei zeigt die Verwendung des Projection-Objekts sowie des Outputformat-Objekts [vgl. Code 2].

```
PROJECTION                        # Beginn des Projektions-Objekts
  "init=epsg:4326"                # EPSG Code des Koordinatenreferenzsystems
END                                # Ende des Projektions-Objekts

OUTPUTFORMAT                      # Beginn des Outputformat-Objekts (Definition
  # des Ausgabeformats)
  NAME gif                        # Schlüsselwort
  DRIVER "GD/GIF"                 # Treiber für Generierung des Ausgabebildes
  MIMETYPE "image/gif"            # MIME-Typ, der verwendet werden soll
  IMAGEMODE PC256                 # Ausgabemodus
  EXTENSION "gif"                 # Dateiendung, die verwendet werden soll
END                                # Ende des Outputformat-Objekts
```

Code 2: Weitere Objekte der Header-Sektion

Über das Projection-Objekt kann die gewünschte Projektion einer Karte (sowie der einzelnen Layer, vgl. S. 71) angegeben werden. Hier wurden geographische Koordinaten im Koordinatensystem WGS84 (EPSG-Code: 4326) gewählt. Wird keine Projektion gewählt, verwendet der UMN MapServer eine Standardprojektion (Plate Carée bzw. die äquidistante zylindrische Projektion).¹⁷⁴ Projektionen können in der Mapdatei über den EPSG-Codes¹⁷⁵ und / oder durch Projektionsparameter angegeben werden. Die EPSG-Codes befinden sich in einer Textdatei (epsg.txt), die auf die PROJ.4-Bibliothek [vgl. Abb. 22: „SUPPORTS=PROJ“] zugreift. Letztere ermöglicht die Umprojektion von Koordinaten on-the-fly. Das Outputformat-Objekt ermöglicht die Definition von Einstellungen für das Ausgabeformat. Welches Outputformat verwendet wird, wird dagegen über den Parameter IMAGE-TYPE bestimmt.

Die Web-Sektion

Jeder Header wird durch eine Web-Sektion [vgl. Code 3] ergänzt. Durch die hier gemachten Angaben wird das Verhalten der Mapdatei nach außen definiert. Beispielsweise wird angegeben, wo die vom UMN MapServer generierten Bilder lokal gespeichert werden sollen (IMAGEPATH) bzw. über welche URL der Webbrowser des Benutzers auf diese zugreifen kann (IMAGEURL). Eine weitere Angabe ist der Pfad zum HTML-Template, dem eigentlichen Interface zum Benutzer.

Soll wie im gegebenen Fall ein OGC-konformer Webdienst aufgebaut werden, so muss die Web-Sektion noch um ein Metadata-Objekt ergänzt werden, in dem verschiedene OWS-Parameter spezifiziert werden. Diesen Parametern wird ein Präfix vorangestellt. Auch wenn im Rahmen der vorliegenden Arbeit zunächst ein WMS aufgebaut worden ist, wurde bewusst das Präfix OWS_ anstelle von WMS_ verwendet. Das Präfix OWS_ steht stellvertretend für alle OWS und kann somit sowohl die Metadaten eines WMS als auch die eines WFS kennzeichnen. Damit ist ein Ausbau des WMS zu einem WFS mit lediglich geringfügigen Ergänzungen möglich. Einige der WFS spezifischen Metadaten wurden bereits jetzt in das Metadaten-Objekt aufgenommen.

```

WEB                                     # Beginn des Web-Objekts
  TEMPLATE "/home/wms/templates/seis_wms.html" # HTML-Template als Interface zum
                                                # Benutzer
  IMAGEPATH "/home/wms/tmp/"              # Pfad zum Speicherort der vom UMN
                                                # MapServer generierten Bilder
  IMAGEURL "/tmp/"                        # Mitteilung unter welcher URL
                                                # das Ausgabebild zu finden ist
METADATA                                # Beginn des Metadaten-Objekts
  "OWS_TITLE"                            "BGR Seismology: German Earthquake Catalogue"
  "OWS_ONLINERESOURCE"
    http://www.seismologie.bgr.de:8080/cgi-bin/mapserv?map=seis_wms.map
  "OWS_SRS"                               EPSG:4326 EPSG:900913 EPSG:25832 [...]"

```

¹⁷⁴ CHRISTL et al. 2008: 206.

¹⁷⁵ EPSG-Codes werden durch das Survey & Positioning Committee der International Organisation of Oil and Gas Producers (OGP) verwaltet. Bei einem EPSG-Code handelt es sich um eine Zahl (Code), die auf eine Reihe von Parametern zur Beschreibung einer bestimmten Projektion verweist.

```

"ows_feature_info_mime_type"      "text/html"
"OWS_ABSTRACT"                    "Earthquakes are continuous [...]."
```

| | |
|------------------------------------|---|
| "OWS_KEYWORDLIST" | "seismicity,earthquakes,magnitude, [...]" |
| "OWS_CONTACTPERSON" | "Nina Helle-Mildt" |
| "OWS_CONTACTELECTRONICMAILADDRESS" | "seisinfo@sdac.hannover.bgr.de" |
| "OWS_CONTACTORGANIZATION" | "Federal Institute for Geosciences and Natural Resources (BGR)" |
| "OWS_ADDRESSTYPE" | "postal" |
| "OWS_ADDRESS" | "Stilleweg 2" |
| "OWS_CITY" | "Hannover" |
| "OWS_POSTCODE" | "30655" |
| "OWS_COUNTRY" | "Germany" |
| "OWS_STATEORPROVINCE" | "Lower Saxony" |
| "ows_contactfacsimiletelephone" | "+49-511-643-3229" |
| "OWS_CONTACTVOICETELEPHONE" | "+49-511-643-6363" |
| "OWS_FEES" | "none" |
| "OWS_ACCESSCONSTRAINTS" | "In case of reuse the information made available shall be quoted as follows: Data source: (C) BGR Hannover 2008." |
| "OWS_SERVICE_ONLINERESOURCE" | "http://www.seismologie.bgr.de" |

```

END                                # Ende des Metadaten-Objekts
END                                # Ende des Web-Objekts

```

Code 3: Die Web-Sektion der Mapdatei

Themendefinition – Beispiel eines Layers

Die über eine Mapdatei spezifizierten Karteninhalte werden in Form von Layern (Kartenschichten) übereinander gelegt.

```

LAYER
NAME "q_year_1"                    # Name des Layers
GROUP "q_year"                    # Name der Gruppe, der der Layer angehört
TYPE POINT                        # Typ der Objekte des Layers
STATUS ON                          # Status des Layers (ON, OFF)
MAXSCALE 200000000                # max. Maßstab der Darstellung des Layers
MINSCALE 50000000                 # min. Maßstab der Darstellung des Layers

CONNECTIONTYPE POSTGIS            # externe Verbindung zur Datenbank
CONNECTION "dbname=seis_ows user=xxx host=xxx port=xxx"
                                   # Verbindungsparameter zur PostGIS Datenbank:
                                   # Datenbankname, Benutzer, Rechnername, Port
DATA "the_geom FROM (select id, ev_date, ev_datetimestring, lat, lon,
mag, reg_name, ev_comm, the_geom from beben WHERE ev_date >
current_date - interval '1 year') as foo USING UNIQUE id USING
SRID=4326"                        # SQL-Statement zur Abfrage der Datenbank
PROCESSING "CLOSE_CONNECTION=DEFER"
# aktiviert Connection-Pooling, damit bestehende
# Datenbank-Verbindungen mehrfach genutzt werden
# können
PROJECTION                        # Beginn des Projektions-Objekts
  "init=epsg:4326"                # EPSG Code des Koordinatenreferenzsystems
END                                # Ende des Projektions-Objekts

TOLERANCE 3                       # Toleranz von punktbasierten Abfragen: 3px
TOLERANCEUNITS pixels             # Einheit für punktbasierten Abfragen
DUMP TRUE                         # Rückgabe Daten in GML (für GetFeatureInfo)
CLASSITEM "mag"                  # Attributname aus Datenbanktabelle (-> Class)
TEMPLATE "/home/wms/templates/german_template.html"
                                   # HTML-Template als Interface zum Benutzer

METADATA                          # Beginn Metadata-Objekt
  "ows_title"                     "beben_1"                            # Titel
  "ows_group_title"               "beben"                            # Gruppentitel
  "ows_srs"                       "epsg:4326"                        # EPSG-Code
  "ows_extent"                    "-180 -90 180 90"                  # Darstellungsbereich
  "ows_abstract"                   "Earthquakes with magnitude (ML) greater 5"
  "ows_order"                      "0"                              #
  "gml_include_items"              "all"                          #
  "ows_include_items"              "all"                              #

```

```

"ows_keywordlist"          "seismicity,earthquakes,magnitude, [...]"
END                        # Ende des Metadata-Objekts

CLASS                      # Beginn des Class-Objekts
  NAME "ML > 5"            # Name
  EXPRESSION ([mag] > 5)   # Voraussetzungen für Darstellung der Class
  STYLE                    # Beginn des Style-Objekts
    SYMBOL "circle"       # Symbol für die Darstellung
    SIZE 9                 # Größe in der das Symbol gezeichnet wird
    COLOR 255 0 0         # Füllfarbe des Symbols
    OUTLINECOLOR 0 0 0    # Linienfarbe (Rand) des Symbols
  END                      # Ende des Style-Objekts
END                          # Ende des Class-Objekts
END                          # Ende des Layer-Objekts

```

Code 4: Beispiel eines Layers

Die Daten des hier beschriebenen Layers (bzw. des OWS insgesamt) sind in einer PostGIS Datenbank gespeichert, auf welche über eine SQL-Abfrage zugegriffen wird. Mit dem Schlüsselwort CLASSITEM wird der Feldname (Attributname) aus der PostGIS-Tabelle übergeben, nach dem klassifiziert werden soll. Die Angaben zur Klassifizierung der Daten werden im CLASS-Objekt definiert. Über das Schlüsselwort EXPRESSION werden die darzustellenden Feldinhalte selektiert. Die Zuweisung von Darstellungsparametern erfolgt dann im STYLE-Objekt.

Die Performance des Webdienstes und Übersichtlichkeit der Kartendarstellung kann durch eine maßstabsabhängige Darstellung des Layers erreicht werden (MINSCALE, MAXSCALE). Schließlich wird das Handling der einzelnen Layer durch die Definition von Gruppen (Schlüsselwort GROUP) erleichtert, da über eine solche Gruppe alle dazugehörigen Layer gleichzeitig angesprochen werden können. Dies ist insbesondere für die Einbindung der Layer in der Clientanwendung von Bedeutung.

3.2.5.2 Test der Mapdatei auf OGC-Konformität

Die OGC-Konformität des im Rahmen der Master Thesis aufgesetzten Dienstes konnte durch ein korrektes Antwortverhalten auf die für einen WMS definierten Requests bestätigt werden. Dabei müssen mindestens die GetCapabilities und die GetMap Anfragen ohne Fehlermeldungen beantwortet werden. Des Weiteren kann die OGC-Konformität auch durch das Einladen des Dienstes in ein externes GIS getestet werden. Hier erfolgt der Test mithilfe von Quantum GIS, einem unter der GNU Public Licence herausgegebenen Projekts. Schließlich bestätigt auch die Verwendung des WMS in der aufgebauten Clientanwendung OpenLayers die angestrebte OGC-Konformität.

Der GetCapabilities-Request

Der GetCapabilities-Request wird über den Browser abgesetzt. Abb. 24 zeigt den Aufbau der aufrufenden URL. Die einzelnen Parameter werden jeweils durch ein ‚Kaufmannsund‘

(&) getrennt. Der besseren Lesbarkeit wegen, ist der Request in zwei Zeilen umgebrochen worden.

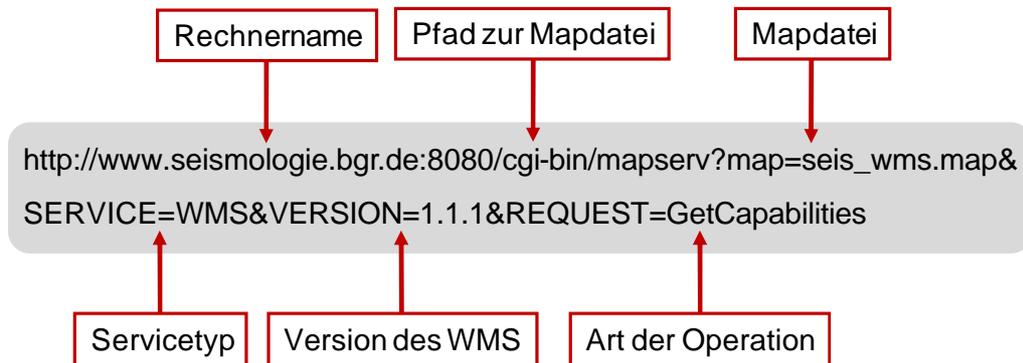


Abb. 24: Der GetCapabilities-Request

Als Ergebnis wird ein XML-Dokument ausgeliefert. Es setzt sich aus verschiedenen Bereichen zusammen [vgl. Abb. 25]. Damit die ausgelieferte XML-Datei keine Warnhinweise (<!--WARNING ... →) enthält, müssen in der Mapdatei alle erforderlichen Mapfile-Parameter und Metadaten gesetzt worden sein.



Abb. 25: Struktur des GetCapabilities-Dokuments

Der GetMap-Request

Auch der GetMap-Request kann als URL über den Browser abgesetzt werden. Nachfolgend sind die benötigten Request-Parameter aufgeführt [vgl. Abb. 26]. Der besseren Lesbarkeit halber wurde die URL in mehrere Zeilen umgebrochen.

```
http://www.seismologie.bgr.de:8080
/cgi-bin/mapserv?
map=seis_wms.map
&SERVICE=WMS
&VERSION=1.1.1
&REQUEST=GetMap
&LAYERS=q_year,q_month,q_week,
        aktuell_welt,countries
&STYLES=
&SRS=EPSG:4326
&BBOX=-180,-90,180,90
&WIDTH=600
&HEIGHT=300
&FORMAT=image/gif
```

- Rechnername
- Pfad zur Mapdatei
- Mapdatei
- Servicetyp: WMS
- Version des WMS
- Art der Operation
- Liste der Layernamen
(Fortsetzung)
- Darstellungsoperationen
- SRS Referenzsystem
- Darstellungsbereich, Bounding Box
- Bildgröße in Pixel (Weite)
- Bildgröße in Pixel (Höhe)
- Ausgabeformat

Abb. 26: Der GetMap-Request

Als Ergebnis wird eine Karte ausgeliefert, die entsprechend dem angeforderten Darstellungsbereich die gesamte Erde umfasst.

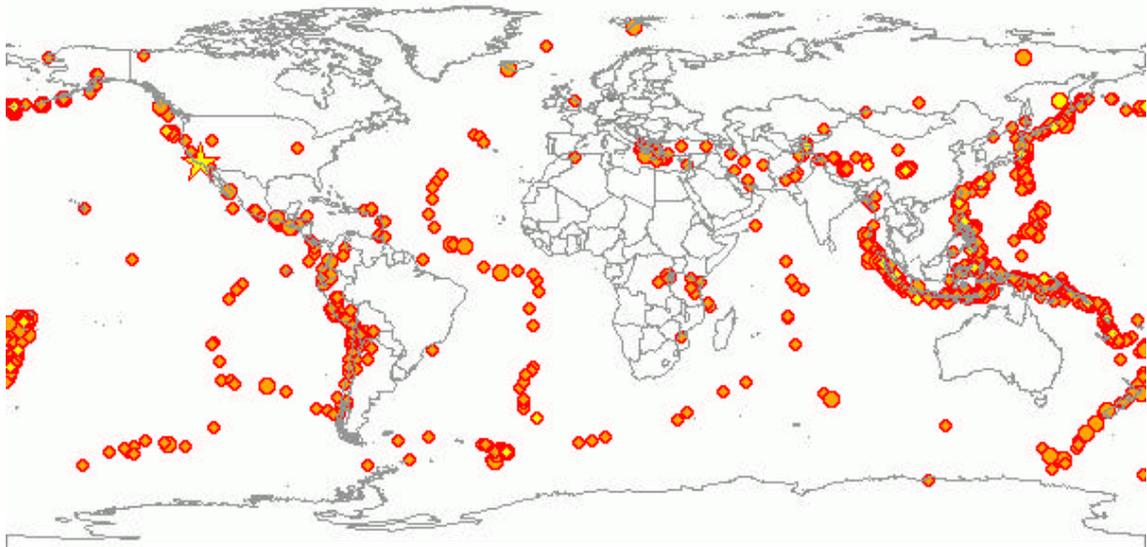


Abb. 27: Antwort auf den GetMap-Request

Der GetFeatureInfo-Request

Entsprechend den im Vorfeld definierten Anforderungen an die aufzubauende GDI, wurde auch der GetFeatureInfo-Request umgesetzt. Abb. 28 zeigt die in mehrere Zeilen umbrochene URL des Requests und das Ergebnis des GetFeatureInfo-Requests.

GetFeatureInfo-Request:

```

http://www.seismologie.bgr.de:8080
/cgi-bin/mapserv
?map=seis_wms.map
&SERVICE=WMS
&VERSION=1.1.1
&REQUEST=GetFeatureInfo
&LAYERS=q_year,countries
&QUERY_layers=q_year
&FORMAT=image/gif
&INFO_FORMAT=text/plain
&SRS=EPSG:4326
&BBOX=-180,-90,180,90
&WIDTH=600
&HEIGHT=300
&X=520&Y=150
&STYLES=
    
```

- Rechnername
- Pfad zur Mapdatei
- Mapdatei
- Servicetyp: WMS
- Version des WMS
- Art der Operation
- Layernamen
- abzufragende Layer
- Format
- Rückgabeformat für Feature-Informationen
- SRS Referenzsystem
- Darstellungsbereich, Bounding Box
- Bildgröße in Pixel (Weite)
- Bildgröße in Pixel (Höhe)
- Pixelkoordinaten (X,Y)
- Darstellungsoperationen

GetFeatureInfo results:

```

Layer 'q_year_8'
Feature 33199:
id = '33199'
ev_date = '2007-10-13'
ev_datetimestring = '2007-10-13 09:42:42.300'
lat = '-0.689'
lon = '132.989'
mag = '3.9'
mc = '4'
reg_name = 'Irian Jaya, Indonesia, region'
ev_comm = "
    
```

Abb. 28: Der GetFeatureInfo-Request

Quantum GIS

Um den WMS in Quantum GIS zu laden, muss über die Eingabe des GetCapabilities-Requests eine Verbindung zum Kartendienst hergestellt werden. Wie der nachfolgende Screenshot [vgl. Abb. 29] zeigt, ist ein GetFeatureInfo-Request auch aus dem OS/FS-Projekt Quantum GIS heraus möglich.

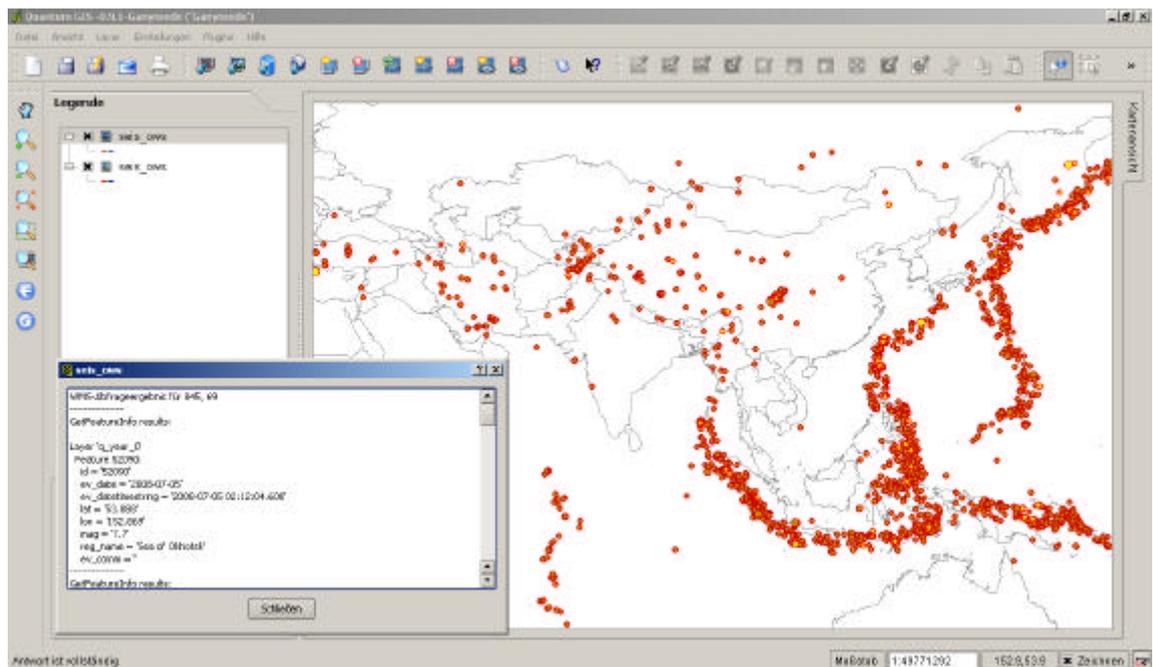


Abb. 29: Darstellung des WMS in Quantum GIS

3.2.6 Der Client – Schnittstelle zum Nutzer

Die Clientanwendung bildet die Schnittstelle zwischen dem serverseitig implementierten Kartendienst und dem Nutzer. Auf Seiten des Nutzers wird dabei lediglich ein Webbrowser vorausgesetzt. Ziel ist der Aufbau einer übersichtlichen für den Großteil der Nutzer intuitiv zu bedienenden Clientanwendung. Aufgrund der hohen Verbreitung von Google Maps und anderen kommerziellen Kartenanwendungen ist heute eine Vielzahl der Internetbenutzer mit deren Nutzung vertraut. Diese Tatsache macht sich das OS/FS-Projekt OpenLayers zu nutze [vgl. Abschnitt 2.3.5]. Es ermöglicht eine einfache Gestaltung der Nutzerschnittstelle mit einem Wiedererkennungseffekt, der den Umgang mit der Anwendung für viele Nutzer erleichtern sollte [vgl. Abb. 30].

Die Komponenten der Clientanwendung

Für die Gestaltung der Benutzeroberfläche bietet OpenLayers eine Vielzahl an unterschiedlichen Komponenten, mit deren Hilfe der Nutzer mit der Karte interagieren kann. Nachfolgend sind wesentliche Komponenten kurz beschrieben und in Abb. 30 dargestellt.¹⁷⁶



Abb. 30: Komponenten der Clientanwendung OpenLayers

- *Hauptkarte*

Die Hauptkarte enthält eine oder mehrere übereinander gelagerte Kartenschichten (Vektor- oder Rastergrafiken). Auch können weitere Komponenten, wie z.B. Symbole (Marker) in der Hauptkarte platziert werden. Deren Auswahl erfolgt über die Ebenübersicht [vgl. unten].

¹⁷⁶ Einen umfassenden Überblick gibt die OpenLayers ‚Class Documentation‘: <http://dev.openlayers.org/releases/OpenLayers-2.6/doc/apidocs/files/OpenLayers-js.html> (12.8.2008).

- *Übersichtskarte*

In der Übersichtskarte wird der in der Hauptkarte jeweils angezeigte Kartenausschnitt visuell hervorgehoben. Sie dient der leichteren Orientierung des Benutzers und ermöglicht gleichzeitig das Navigieren in der Hauptkarte.

- *Ebenenübersicht*

Die Ebenenübersicht zeigt die zur Verfügung stehenden Hintergrundkarten (Base Layer) und sonstigen Kartenebenen (Overlays, z.B. seismische Ereignisse) an. Der Nutzer kann hier eine eigene Auswahl treffen.

- *Pan-Zoom-Navigationsleiste*

Über die vier „Navigationspfeile“ kann der Nutzer den Kartenausschnitt verschieben sowie über das Plus- bzw. Minuszeichen in die Karte hinein oder aus der Karte heraus zoomen. Ein Klick auf das kleine Weltkugel-Symbol führt zur Darstellung der maximalen Kartensicht.

- *Werkzeugleiste*

In die Werkzeugleiste können verschiedene Buttons zur Aktivierung von verfügbaren GIS-Funktionen integriert werden. Hier wurden die Pan- sowie die Zoom-Box-Funktion gewählt.

- *Maßstabsbalken*

Der Maßstabsbalken ist an die jeweilige Zoomstufe der Hauptkarte gekoppelt.

- *Koordinatanzeige*

Eine Koordinatanzeige zeigt die jeweilige Maus-Position in geographischen Koordinaten an.

OpenLayers auf dem eigenen Server

Um den OpenLayers Kartenviewer in den eigenen Webauftritt zu integrieren, genügt im einfachsten Fall der Verweis auf das OpenLayers API (Application Programming Interface). Im Rahmen der vorliegenden Arbeit soll OpenLayers jedoch über den eigenen Webserver bereitgestellt werden. Dafür muss der OpenLayers-Quellcode¹⁷⁷ heruntergeladen und die JavaScript-Bibliotheken über den Apache Webserver so verfügbar gemacht werden, dass über eine einfache HTML-Seite auf Bibliotheken zugegriffen werden kann.

¹⁷⁷ <http://www.openlayers.org/download/> (Stand: 12.08.2008).

Aufbau der OpenLayers Kartenanwendung

Der Aufbau des OpenLayers Viewers erfordert die Erstellung und Anpassung einer HTML-Seite, in welcher der Viewer angezeigt wird. Im Folgenden werden die wesentlichen Ausschnitte aus der HTML-Seite herausgegriffen und erläutert.

Der OpenLayers Viewer kann nahezu in jedes Element der HTML-Seite integriert werden. Zunächst muss allerdings über einen ‚script tag‘ die OpenLayers Bibliothek in die Seite eingebunden werden. Sollen – wie im vorliegenden Beispiel – die von Google Maps (bzw. einem anderen proprietären Drittanbieter) angebotenen Karten als Hintergrundtopographie genutzt werden, muss auch die Google Map API über einen ‚script tag‘ in die HTML-Seite eingebunden werden [vgl. Code 5].

```
<html>
<head>
  <base href="http://www.seismologie.bgr.de" />
  <title>BGR/Seismologie/Aktuelle Seismische Ereignisse in Deutschland
    und weltweit</title>
  [...]
  <!-- OpenLayers API -->
  <script
src="http://www.seismologie.bgr.de/sdac/erdbeben/kartenanwendung/OpenLayers.js">
  </script>
  <!--Google Maps API -->
  <script src='http://maps.google.com/maps?file=api&v=2&key=[...]'>
  </script>
  [...]
</head>
[... ]
</html>
```

Code 5: Integration der API von OpenLayers und Google Maps

Aufgrund der objektorientierten Programmierweise gliedert sich die OpenLayers-API in eine Vielzahl von Klassen. Für den grundlegenden Aufbau einer OpenLayers-Anwendung sind die beiden folgenden Klassen von zentraler Bedeutung:¹⁷⁸

- *OpenLayers.Map*

Über die Klasse *OpenLayers.Map* werden die Hauptkarte der Anwendung generiert und verschiedene Methoden zur Verwaltung der Kartenanzeige bereitgestellt. Zu nennen sind u.a. die Darstellung von Ebenen, verschiedenen Bedienelementen sowie die Möglichkeit, die Karte zu verschieben oder in diese hinein oder aus dieser heraus zu zoomen. Ferner beinhaltet die ‚Map‘ Informationen über die Standardprojektion, die Ausmaße und Einheiten etc. der darzustellenden Karte.

- *OpenLayers.Layer*

Die eigentlichen Karteninhalte werden über ‚Layer‘ bezogen. Dabei wird über einen Layer spezifiziert, wie OpenLayers die Informationen abfragen und darstellen soll.

¹⁷⁸ <http://openlayers.org/doc/> (Stand: 17.08.2008); SCHÜTZE 2007: 49 ff.

`OpenLayers.Layer` bildet eine Basisklasse, die verschiedene Unterklassen enthalten kann, z.B. `.Layer.Google` oder `.Layer.WMS.Untiled`.

Der folgende Ausschnitt [vgl. Code 6] aus der HTML-Datei zeigt den grundlegenden Aufbau des Kartenviewers. Die Hauptkarte wird über die zentrale Klasse `Openlayers.Map` generiert und über die Variable `options` mit den für die ganze Karte geltenden Einstellungen versehen. Unter Verwendung der Klasse `Openlayers.Map` werden dann die eigentlichen Karteninhalte, d.h. der Hintergrund- und die Darstellungsebenen, eingebunden und die entsprechenden Darstellungparameter festgelegt. Eine Darstellungsebene unterscheidet sich von einer Hintergrundebene durch die Angabe des Parameters `,transparent'`, der den Wert `,true'` erhält. Schließlich werden die Kartenebenen über `map.addLayers` in die Hauptkarte geladen. Damit die Karte im Viewer dargestellt wird, muss allerdings noch das Kartenzentrum und die Ausgangs-Zoomstufe festgelegt werden (`map.setCenter`).

```
//Festlegungen von Optionen für die Hauptkarte:
var options = {
  projection: new OpenLayers.Projection("EPSG:900913"),
  units: "m",
  maxResolution: 156543.0339,
  maxExtent: new OpenLayers.Bounds(
    -20037508.34, -20037508.34, 20037508.34, 20037508.34),
  restrictedExtent: restrict};

//Generierung der Hauptkarte:
map = new OpenLayers.Map("map", options);

//Googel Maps Hintergrundebenen (Base Layers):
var gphy = new OpenLayers.Layer.Google(
  "Google Physical", {"sphericalMercator": true, type: G_PHYSICAL_MAP});
var ghyb = new OpenLayers.Layer.Google(
  "Google Hybrid", {"sphericalMercator": true, type: G_HYBRID_MAP});
var gsat = new OpenLayers.Layer.Google(
  "Google Satellite", {"sphericalMercator": true, type: G_SATELLITE_MAP});

//Kerninhalte / WMS (Overlay Layers):
var seis = new OpenLayers.Layer.WMS(
  "Erdbeben weltweit ",
  "http://www.seismologie.bgr.de:8080/cgi-bin/mapserv?map=seis_ger_wms10.map",
  {transparent: 'true', layers: ['q_year', 'q_month', 'q_week', 'aktuell_welt']},
  {singleTile: true, ratio: '1.0'},
  {'isBaseLayer': false, 'gutter': 17});
var seis_ger = new OpenLayers.Layer.WMS(
  "Erdbeben Deutschland",
  "http://www.seismologie.bgr.de:8080/cgi-bin/mapserv?map=deutschland.map",
  {transparent: 'true', layers:
    ['q_year_ger', 'q_month_ger', 'q_week_ger', 'aktuell_ger', 'box']},
  {singleTile: true, ratio: '1.0'},
  {'isBaseLayer': false, 'gutter': 17});
var sta = new OpenLayers.Layer.WMS(
  "Stationen in Deutschland (Datenbezug)",
  "http://www.seismologie.bgr.de:8080/cgi-bin/mapserv?map=deutschland.map",
  {transparent: 'true', layers: 'stations_deutschland'},
  {singleTile: true, ratio: '1.0'},
  {'isBaseLayer': false, 'gutter': 4});

//Hinzufügen der Layer zur Map (Hauptkarte):
map.addLayers([gphy, ghyb, gsat, seis, seis_ger, sta]);

//Kartenzentrum und Zoomstufe definieren
map.setCenter(new OpenLayers.LonLat(1122707.07130, 6654591.36368), 4);
```

Code 6: Einbindung von Hauptkarte und Kartenebenen über die Klassen `,OpenLayers.Map'` und `,OpenLayers.Layer.Google'` bzw. `,OpenLayers.Layer.WMS'`

Damit der Nutzer der Anwendung in der Karte navigieren kann, werden über die Klasse *OpenLayers.Control* Bedienelemente eingebunden. Die Klasse *Control* dient dabei als Basisklasse für alle Bedienelemente. Ihre Verwendung ist im folgenden Codebeispiel (vgl. Code 7) zusehen. Das Layout (Position, Farbigkeit etc.) der einzelnen Bedienelemente wird über ein Stylesheet (.css) angepasst.

```
//Hinzufügen von Bedienelementen:
//Ebenenübersicht
map.addControl(new OpenLayers.Control.LayerSwitcher({'ascending':false}));
//Maßstabsleiste
map.addControl(new OpenLayers.Control.ScaleLine());
//Fortschrittsanzeige
map.addControl(new OpenLayers.Control.LoadingPanel());
//Werkzeugleiste (Panfunktion und Zoombox)
map.addControl(new OpenLayers.Control.NavToolbar());

//Koordinatenlaufleiste
mp=new OpenLayers.Control.MousePosition();
mp.displayProjection = new OpenLayers.Projection("EPSG:4326");
map.addControl(mp);
```

Code 7: Einbindung von Bedienelementen

Im Vorfeld der Anwendungsentwicklung wurde als wesentliches Ziel festgelegt, dass durch die Möglichkeit der Abfrage einzelner seismischer Ereignisse ein Mehrwert für den Nutzer geschaffen werden soll. Diese Anforderung wird durch die Schaffung einer interaktiven Abfragemöglichkeit der über den WMS generierten Erdbebenpunkte umgesetzt [vgl. Code 8].

```
//support WMS GetFeatureInfo-Request
map.events.register('click', map, function (e) {
OpenLayers.Util.getElement('nodeList').innerHTML = "Einen Moment bitte ...";
var url = seis.getFullRequestString({
REQUEST: "GetFeatureInfo",
EXCEPTIONS: "application/vnd.ogc.se_xml",
BBOX: seis.map.getExtent().toBBOX(),
X: e.xy.x,
Y: e.xy.y,
INFO_FORMAT: 'text/html',
QUERY_LAYERS: seis.params.LAYERS,
WIDTH: seis.map.size.w,
HEIGHT: seis.map.size.h});
OpenLayers.loadURL(url, '', this, setHTML);
OpenLayers.Event.stop(e);
});
function setHTML(response) {
OpenLayers.Util.getElement('nodeList').innerHTML = response.responseText;}
```

Code 8: Abfrage des GetFeatureInfo-Requests

Der oben abgebildete Code-Ausschnitt [vgl. Code 8] bewirkt Folgendes. Zunächst wird die Funktion registriert, damit sie ausgewählt wird, wenn ein Click-Event stattfindet. ,e‘ ist dabei das Event, das den GetFeatureInfo-Request auslöst. Als erstes ändert die Funktion den ,innerHTML‘ der ,nodeliste‘ zu "Einen Moment bitte ...". So wird der Nutzer über den Ladevorgang informiert. Dann wird eine neue URL zusammengestellt, die auf der in den

Karten-Layern angegebenen URL [vgl. Code 6] basiert. Dabei wird der Request in einen *GetFeatureInfo*-Request [vgl. Abb. 28] umgewandelt, die *BoundingBox* (BBOX) mit dem aktuellen Kartendarstellungsbereich gleichgesetzt und die Werte für ‚WIDTH‘ und ‚HEIGHT‘ mit der aktuellen Kartengröße abgeglichen. Dann wird der URL noch die aktuelle X- und Y-Position des Click-Events angehängt sowie die zu durchsuchenden Layer und das ‚INFO-FORMAT‘ angegeben. Abschließend wird die ‚loadURL‘ Funktion gestartet und definiert, dass die Funktion ‚setHTML‘ aufgerufen werden soll, um das Ergebnis des *GetFeatureInfo*-Requests entgegenzunehmen. Letztlich wird dem Event-Prozessor mitgeteilt, dass die Funktion gestoppt werden kann. Die letzten beiden Code-Zeilen geben an, wie die Funktion ‚setHTML‘ mit dem zurückgelieferten Ergebnis umgehen soll. Hier ist festgelegt worden, dass sie die Response in das ‚nodelist‘ Element des HTML-Dokuments kopiert. Somit erfolgt die Ausgabe der Response, wie in Abb. 31 zu sehen ist, neben dem Kartenfenster.

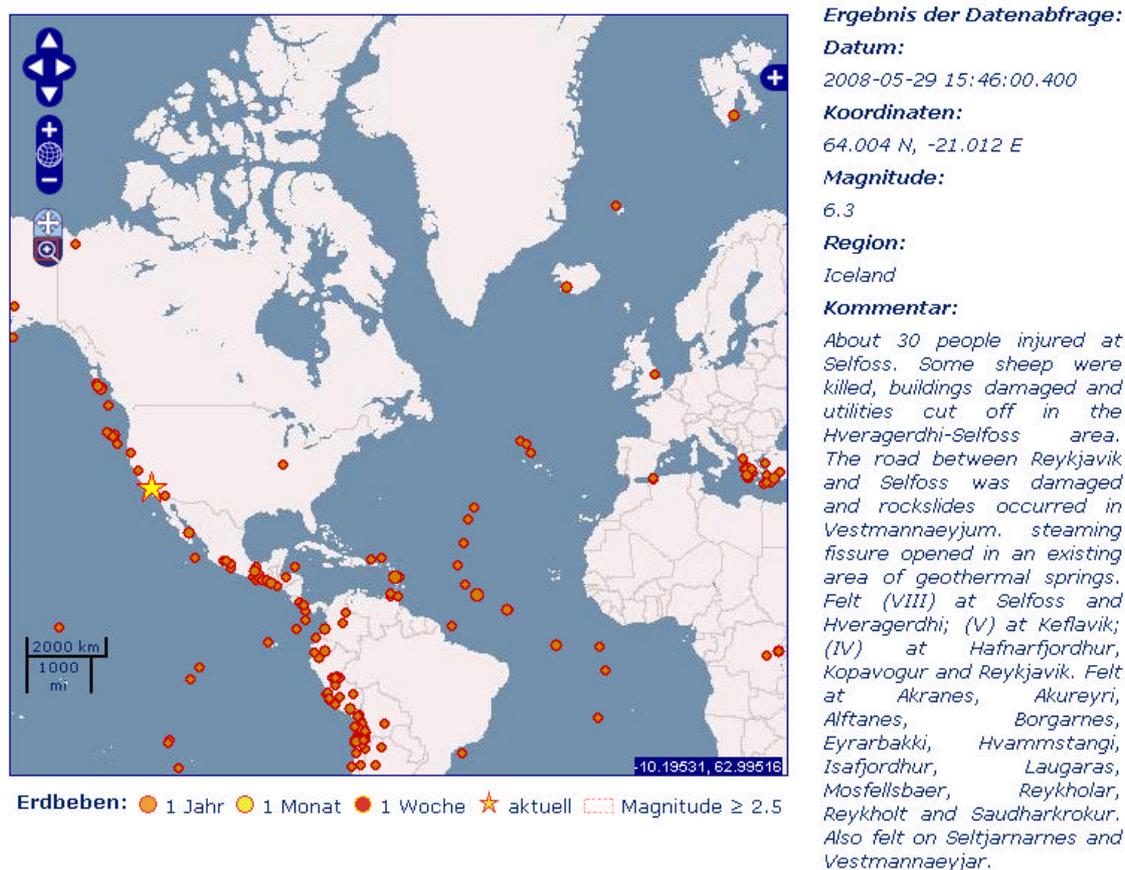


Abb. 31: OpenLayers-Anwendung: Seismische Ereignisse in Deutschland und weltweit¹⁷⁹

Die OpenLayers-Anwendung ist unter folgender URL im Internet aufrufbar:
<http://www.seismologie.bgr.de/sdac/erdbeben/kartenanwendung/unigis.html>.

¹⁷⁹ Für den Screenshot wurde aus lizenzrechtlichen Gründen der von OpenLayers zur Verfügung gestellte WMS als Kartenhintergrund verwendet.

3.2.7 Aufnahme von Metainformationen in den Geodatenkatalog des Geozentrums Hannover

Damit die erstellte OpenLayers-Anwendung sowie der WMS über das Internet gefunden werden können, müssen sie abschließend im Geodatenkatalog (GeoDak) des Geozentrums Hannover publiziert werden. Die nachfolgende Abbildung [vgl. Abb. 32] zeigt eine entsprechende Abfrage des GeoDak. Da es sich beim GeoDak nicht um OS/FS-Software handelt, wird der Aufbau des Metadatenkatalogs an dieser Stelle nicht weiter thematisiert.



The screenshot shows the GeoDak search results page. At the top left is the GeoDak logo, which consists of a square icon with concentric circles and the text 'GeoDak' in a blue, handwritten-style font. Below the logo are four navigation buttons: 'START', 'SUCHE', 'INFO', and 'HILFE'. The main content area is titled 'Suchergebnis' and includes a red '< zurück' link. The search results are for 'BGR Seismologie: Erdbebenkatalog'. The 'Thema:' section lists 'Geowissenschaften'. The 'Schlüsselwörter:' section lists 'Seismik, Seismologie, Naturkatastrophen, Erdbeben, Erdbebendaten, Epizentrum, Magnitude, WMS, OWS'. The 'Zusammenfassung:' section provides a detailed description of the data, mentioning the German Regional Network (GRSN) and arrays like GERESS and Gräfenberg (GRF). The 'Ausdehnung:' section indicates the map extent covers the data set. The 'Grafische Vorschau:' section shows a world map with numerous red dots representing seismic events, primarily concentrated in the Pacific and Indian Ocean regions.

Abb. 32: Eintrag des WMS in den GeoDak¹⁸⁰

¹⁸⁰ Der Eintrag des WMS in den GeoDak ist zum Zeitpunkt der Einreichung der vorliegenden Master Thesis lediglich intern publiziert. Abb. 32 zeigt die nur intern verfügbare Vorschau.

4 Bewertung und Ausblick

4.1. Bewertung

4.1.1 Bewertung der Entwicklung eines Szenarios zur Umsetzung der INSPIRE-Richtlinie im Rahmen der GfDI-BGR auf Basis von OS/FS (GDI-Systemarchitektur)

Mit der in Abschnitt 2.3 entwickelten Systemarchitektur lassen sich die zu Beginn der Arbeit [Abschnitt 1.1] identifizierten Komponenten einer GDI problemlos umsetzen. Abb. 33 stellt einen Bezug zwischen den einzelnen GDI-Komponenten und den eingesetzten OS/FS-Produkten sowie den verwendeten offenen Standards her. Deutlich wird die grundlegende Bedeutung der Einhaltung von offenen Standards, um eine Interoperabilität zu gewährleisten. Ferner baut die Systemarchitektur, entsprechend der Aufgabenstellung der Arbeit, (fast) ausschließlich auf OS/FS-Produkten auf. Eine kleine Abweichung stellt die Integration des proprietären Online-Kartendienstes Google Maps dar. Diese wurde notwendig, da der eigene Aufbau einer qualitativ vergleichbaren Basistopographie nicht wirtschaftlich gewesen wäre. Die Möglichkeit, externe Dienste in die eigene Anwendung einzubinden, zeigt jedoch beispielhaft die Vorteile, die sich durch die hohe Kompatibilität von OS/FS-Projekten ergeben.

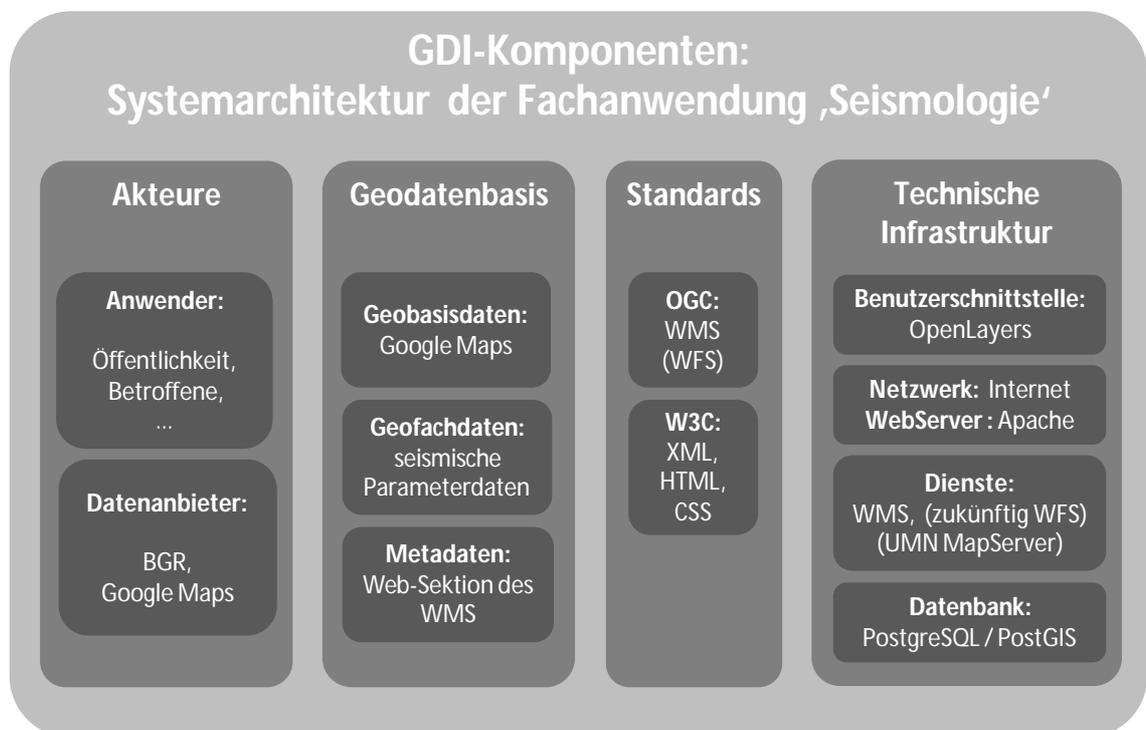


Abb. 33: Entwickelte Systemarchitektur auf Basis von OS/FS-Produkten

Die Systemarchitektur der Fachanwendung ‚Seismologie‘ klammert den für GDIs zentralen Aspekt der Metadatenhaltung zumindest teilweise aus. Dieser wurde allerdings bereits übergreifend für alle BGR-Fachbereiche durch die Umsetzung eines CSW sowie der Metadatenkatalog-Anwendung GeoDak realisiert. Metadaten werden im Rahmen der Fachanwendung ‚Seismologie‘ über die Web-Sektion des WMS bereitgestellt und sind über den GetCapabilities-Request abrufbar.

4.1.2 Bewertung der eingesetzten OS/FS-Produkte

Im Folgenden werden die im Rahmen der Anwendungsimplementierung mit den eingesetzten OS/FS-Produkten gesammelten Erfahrungen bewertet. Anzumerken ist, dass die Autorin im Vorfeld der Implementierung mit den einzelnen Softwareprodukten nicht vertraut war. Auftretende Probleme konnten in der Regel jedoch durch eine Recherche im Internet, vorzugsweise auf der Produkt-Homepage der jeweiligen Software, gelöst werden. Ebenfalls wurde auf die Userforen und Mailinglists der eingesetzten OS/FS-Projekte zurückgegriffen. Auch wenn die Rückmeldungen auf die gestellten Fragen in den meisten Fällen innerhalb kürzester Zeit erfolgten und die Hilfestellungen zu effektiven Problemlösungen beigetragen haben, besteht bei Foren und Mailinglisten naturgemäß keine Gewähr auf Support. Dies ist sicherlich ein wesentlicher Unterschied zu einer Support-Hotline eines proprietären Softwareanbieters, deren Verfügbarkeit in der Regel aber auch vertraglich geregelt sein muss und kostenpflichtig ist.

PostGIS

Im Zuge der Arbeit wurde eine PostgreSQL/PostGIS Datenbank eingerichtet, da die vorhandene OS/FS-Datenbank keine Unterstützung räumlicher Objekte geboten hat. Voraussetzung für den Aufbau einer PostGIS Datenbank ist, dass auf einer bestehenden PostgreSQL Datenbank aufgebaut werden kann. Bedingt durch eine gute Dokumentation verlief die Einrichtung der PostGIS Datenbank¹⁸¹ problemlos, zumal keine komplexe Datenbank-Struktur aufgebaut worden ist [vgl. Abb. 21] Die Funktionalität der Datenbank entspricht den gesetzten Erwartungen.

Die Abfrage der Datenbank durch den UMN MapServer (Mapdatei) funktioniert zuverlässig. Allerdings wurden bei Performance-Tests deutliche Unterschiede zwischen der Abfrage der seismischen Parameterdaten über die PostGIS Datenbank und über ein zu Testzwecken erstelltes Shapefile festgestellt. Dabei schnitt die Datenbank deutlich schlechter ab. Recherchen ergaben, dass es sich dabei um ein bekanntes Phänomen handelt und dass das Ant-

¹⁸¹ CHRISTL et al. 2008: 318 ff.

wortverhalten der PostGIS Datenbank unter einem Linux-Betriebssystem erheblich schlechter ist (40 % langsamer), als unter Windows.¹⁸²

Apache Webserver

Der Apache Webserver wird im SDAC der BGR schon über einen längeren Zeitraum eingesetzt und läuft seitdem stabil. Für allgemeine Entwicklungsarbeiten, zu denen auch die hier behandelten GDI-Komponenten gehören, wird ein Test-Server betrieben. Somit mussten im Rahmen der vorliegenden Arbeit lediglich kleine Anpassungen in der Konfigurationsdatei vorgenommen werden [vgl. Abschnitt 3.2.4].

UMN MapServer

Der UMN MapServer wird in der BGR bereits unter Windows-Betriebssystemen eingesetzt. Die Installation unter Windows gestaltet sich vergleichsweise einfach, da eine Setup-Routine genutzt werden kann. Im Laufe der letzten Jahre wurden ausgiebige Erfahrungen mit diesem Produkt gesammelt. Die Software läuft seitdem sehr zuverlässig.

Am SDAC musste für den Aufbau eines WMS jedoch eine neue MapServer Version installiert werden. Bedingt durch den Einsatz eines Unix-artigen Betriebssystems (SunOS) gestaltete sich die Installation im Vergleich wesentlich komplexer. Hierfür waren entsprechende IT-Erfahrungen mit Unix-basierten Systemen, eine genaue Kenntnis von den benötigten Komponenten in Hinsicht auf die gewünschten Funktionen des UMN MapServers sowie eine geeignete Dokumentation der Installationsschritte¹⁸³ erforderlich, da insbesondere vor dem Kompilieren des MapServer-Programms verschiedene andere Optionen konfiguriert werden mussten.

Für den Aufbau der Mapdatei wurde ein bestehender WMS genutzt. Diese Vorgehensweise bot den Vorteil, dass die elementaren Schlüsselwörter nicht erst zusammengestellt werden mussten. Die Angaben in der Metadata-Sektion konnten sogar teilweise mitsamt den gesetzten Werten übernommen werden [vgl. Code 3]. Eine gute Hilfestellung beim Aufbau der Mapdatei bildet die ‚Mapfile Reference‘¹⁸⁴ auf der UMN-MapServer Produktseite. Hier sind die benötigten Informationen ausführlich und übersichtlich zusammengestellt.

4.1.2.1 OpenLayers

Das OS/FS-Projekt OpenLayers bedarf keiner Installation [vgl. Abschnitt 3.2.6, S. 77]. Die notwendigen Schritte für die Nutzung von OpenLayers auf dem eigenen Server sind in einer auf der Projekt-Homepage zur Verfügung gestellten Anleitung gut dokumentiert.

¹⁸² ZETTEL 2007: 43

¹⁸³ CHRISTL et al. 2008: 49 ff.

¹⁸⁴ <http://mapserver.gis.umn.edu/docs/reference/mapfile> (Stand: 26.08.2008)

Einen Überblick über die Möglichkeiten des Programms bieten zahlreiche Beispiel-Anwendungen auf der Projekt-Homepage.¹⁸⁵ Da ein OpenLayers-Script einfach in eine HTML-Seite eingefügt werden kann, hat jeder, der den Link einer Anwendung kennt, die Möglichkeit, deren Funktionsweise durch die Analyse des Seitenquelcodes nachzuvollziehen. Damit kann das Wissen um den Aufbau von OpenLayers-Anwendungen auch einfach und vor allem effizient weiter gegeben werden.

OpenLayers nutzt das Prinzip der Kachelung (engl. tiling), um eine Internetkarte in kleine Kacheln zu unterteilen und damit die „Usability“ einer Kartenanwendung zu steigern. Optisch nimmt der Nutzer nur eine Gesamtkarte wahr. Neben den in der OpenLayers-Anwendung sichtbaren Kacheln werden auch alle nicht sichtbaren Kacheln geladen, die an die im Kartenviewer dargestellte Region angrenzen. Das Ergebnis ist eine verzögerungsfreie Anzeige der Karten beim Verschieben im Kartenviewer. Im Fall der Fachanwendung ‚Seismologie‘ hatte dieses Verfahren jedoch einen entscheidenden Nachteil. Für jede geladene Kachel – sichtbar oder nicht – erfolgte eine Abfrage an den DatenServer, pro „Klick“ wurden so zum Teil bis zu 16 Kartenbilder vom UMN MapServer (16 Datenbankabfragen) angefordert. In der Folge kam es, insbesondere wenn mehrere User (im Testbetrieb) gleichzeitig auf die Anwendung zugegriffen haben, zu spürbaren Performanceeinbußen. Um diesem Verhalten entgegen zu wirken, wurde im OpenLayers-Script explizit festgelegt, dass jeweils nur eine ungekachelte Karte dargestellt werden soll. Auch sollen keine umgebenden Kacheln mehr geladen und vorsorglich bereitgestellt werden. Mit dieser Vorgehensweise wurde zugunsten einer besseren Performance bei Mehrfachanfragen ein weniger gutes Anzeigeverhalten in Kauf genommen.

4.1.3 Bewertung von Mehrwert, Umsetzbarkeit und Grenzen des Einsatzes von OS/FS beim Aufbau von GDIs

Im Folgenden werden die im Abschnitt 2.2.1 erarbeiteten Vorteile und Grenzen von OS/FS-Produkten mit denen im Praxistest [Abschnitt 3] gesammelten Erfahrungen vergleichend bewertet. Der textlichen Erläuterung der einzelnen Bewertungsaspekte folgt eine zusammenfassende Bewertungsmatrix. Hier werden den in Abschnitt 2.2.1 beschriebenen Vorteilen und Grenzen die mit den eingesetzten OS/FS-Produkten gesammelten Erfahrungen gegenüber gestellt.

Sicherheit, Stabilität und Qualität

Die eingesetzte Software läuft im Rahmen der aufgebauten GDI-Komponenten stabil. Während der Arbeit konnten keine Probleme festgestellt werden, die auf eines der verwendeten Produkte zurückzuführen gewesen wären. Für eine abschließende Beurteilung zur Sicher-

¹⁸⁵ <http://openlayers.org/dev/examples/> (Stand: 26.08.2008).

heit und Qualität der Software ist der Entwicklungs- und Anwendungszeitraum von rund sechs Monaten allerdings zu kurz. Auch steht die eigentliche Bewährungsprobe – der Zugriff einer nur schwer einzuschätzenden Anzahl von Nutzern nach einem großen Erdbebenereignis (z.B. in der Größenordnung der Tsunami-Katastrophe im Dezember 2004) – noch aus.

Allerdings wurden im Rahmen der Anwendungsentwicklung die einschlägigen Foren und Mailinglists der jeweiligen Produkte, insbesondere des UMN MapServers und von OpenLayers, in regelmäßigen Abständen konsultiert. Positiv aufgefallen ist der offene Umgang mit Fehlern in der Software. Jeder Anwender und Entwickler kann einen Fehler dokumentieren und/oder zu dessen Behebung beitragen. Ferner dienen die Foren und Mailinglists einem Meinungs- und Erfahrungsaustausch, auch in Fragen der Weiterentwicklung der Software. Ist die Integration einer neuen Softwarefunktionalität zwar für den Release der nächsten Version vorgesehen, diese aber noch nicht ausgereift, so wird ihre Integration in die Software aufgeschoben. Eine solche Vorgehensweise ist aus Anwendersicht zu begrüßen und trägt zur Sicherheit, Stabilität und Qualität einer Software bei.

Interoperabilität bzw. Kompatibilität

Ein wesentliches Kriterium für die Auswahl der eingesetzten Software war deren Integration und Umsetzung der für die Kommunikation zwischen den eingesetzten Softwareprodukten benötigten Schnittstellen (offene Standards). Im Resultat kann eine einwandfreie Interoperabilität der verwendeten Produkte untereinander bestätigt werden. Darüber hinaus wurde auch die Kompatibilität mit anderen Produkten getestet. So konnte der proprietäre Online-Kartendienst von Google Maps problemlos in die OpenLayers-Anwendung integriert werden und der mit dem UMN MapServer aufgebaute WMS ließ sich einfach in ein externes GIS-Programm, wie z.B. das OS/FS-Projekt QuantumGIS, einbinden.

Zum Teil kommt es zu Darstellungsschwierigkeiten noch unbekannter Ursache beim Aufruf der OpenLayers-Anwendung im proprietären Internet Explorer (IE).

Bedarfsgerechte Anpassung durch Modularität

Auch die eingesetzten Softwareprodukte sind modular aufgebaut. So wurden beispielsweise bei der Kompilierung des UMN MapServers gezielt die unter Abb. 22 aufgeführten MapServer-Optionen ausgewählt. Damit der UMN MapServer z.B. Daten von einer PostGIS Datenbank beziehen kann, stellt dieses Produkt explizit eine besondere Option (--with-postgis[=ARG]) zur Verfügung, die bei der Konfiguration anzugeben ist und damit die Schnittstelle zur PostGIS Datenbank ermöglicht. Auch bei OpenLayers besteht die Möglichkeit, die Software in einer reduzierten und speziell auf die Anwendung zugeschnittenen Version auf dem eigenen Server bereitzustellen.

Die Möglichkeit einer bedarfsgerechten Anpassung der jeweiligen Software über den Source-Code wird vor allem bei der zukünftigen Integration der einzelnen GDI-Komponenten in die GfDI-BGR zum Tragen kommen. Hier wird es erforderlich sein, zumindest einen Teil der Software (z.B. den Kartenviewer) zu individualisieren. Nur so kann den Anforderungen unterschiedlicher Fachrichtungen entsprochen werden.

Unabhängigkeit von Softwareanbietern

Die Unabhängigkeit von einem proprietären Softwareanbieter ermöglichte eine freie Auswahl zwischen verschiedenen grundsätzlich für den Aufbau von GDIs in Frage kommenden OS/FS-Produkten. Die einzige Ausnahme bildete der Einsatz von OpenLayers. Dieses OS/FS-Projekt wurde bewusst ausgewählt, weil es die Möglichkeit bietet, Karten des proprietären Online-Kartendienstes Google Maps einzubinden [vgl. Abschnitt 2.3.5]. Die Einbindung von Google Maps bringt einige Anpassungserfordernisse mit sich, so muss der Produktname „Google“ sowie ein Verweis auf die Google-spezifischen Nutzungsbedingungen innerhalb des Kartenrahmens der OpenLayers-Anwendung eingebunden werden.

Durch die Unabhängigkeit von einem Softwareanbieter wird die erforderliche Flexibilität für einen an bereits bestehenden GDI-Komponenten orientierten dynamischen Auf- und Ausbaus der GfDI-BGR erst ermöglicht. Nur so können einzelne GDI-Komponenten flexibel ersetzt bzw. weitere hinzugefügt werden, ohne dass das interoperable Zusammenwirken der einzelnen Komponenten gestört wird.

Wettbewerbsförderung

Das Kriterium der Wettbewerbsförderung wird erst im Rahmen der praktischen Umsetzung einer GfDI-BGR wirklich an Bedeutung gewinnen, wenn bei der fachbereichsübergreifenden Entwicklung einzelner GDI-Komponenten auf den Support von OS/FS-Distributoren zurückgegriffen werden soll. Die BGR wird sich dabei an den Vorgaben der Bundesregierung zum Einsatz von OS/FS-Produkten orientieren.¹⁸⁶

Investitionssicherheit

Aussagen zur Investitionssicherheit lassen sich im Rahmen der vorliegenden Arbeit nur bedingt treffen. Die eingesetzten Softwareprodukte sind schon seit längerem auf dem Markt. Jüngstes Projekt ist OpenLayers, dessen erste offizielle Version (1.0) im Juni 2006 veröffentlicht wurde. Seitdem hat es eine rasante Entwicklung durchlaufen: Im August 2006 wurde bereits die Version 2.0 veröffentlicht, die heute aktuelle Version ist 2.6. Allen eingesetzten OS/FS-Projekten gemeinsam ist eine aktive Entwickler-Community. Damit sind keine Anhaltspunkte erkennbar, die gegen eine mittel- bis langfristige Weiterentwicklung

¹⁸⁶ BT Drucksache 14/5246, 2001: 5.

und Pflege der Softwareprodukte sprechen würden. Die Tatsache wird als „Quasi-Entwicklergarantie“ gewertet.

Darüber hinaus besteht innerhalb und zwischen den einzelnen Organisationseinheiten der öffentlichen Verwaltung, bedingt durch ähnliche Aufgabenstellungen, die Möglichkeit einer Kooperation bei der Weiterentwicklung einzelner Software-Komponenten.

Keine Entwicklergarantie

Eine fehlende Entwicklergarantie wird nicht als besonderer Nachteil von OS/FS-Produkten gewertet [vgl. Ausführungen zur Investitionssicherheit oben].

Mangelnde Unterstützung neuer Hardware und fehlende Installationsroutinen

Von HILLEMIEIER et al¹⁸⁷ wurde darauf hingewiesen, dass sich im OS/FS-Bereich häufig eine mangelnde Hardwareunterstützung zeigt [vgl. S. 41]. Die Unterstützung neuer Hardware durch die eingesetzten OS/FS-Produkte kann im Rahmen der Arbeit nur bedingt beurteilt werden, da sie bereits etablierte Hardware zu Einsatz kam.

Aus sicherheitstechnischen Überlegungen wurde zugunsten eines Webserver eines weniger weit verbreiteten Unix-artigen Betriebssystems (SunOS) entschieden. Dieses hatte jedoch negative Auswirkungen auf den Installationsaufwand des UMN MapServers [vgl. Abschnitt 4.1.2, S. 85]. Das Vorhandensein von Installationsroutinen ist sicherlich auch vom Betriebssystem (Unix-artiges Betriebssystem oder Windows) abhängig, auf dem das OS/FS-Produkt installiert werden soll.

Keine Gewährleistung, Service und Support

HILLEMIEIER et al¹⁸⁸ führen eine fehlende Gewährleistung sowie einen nicht vorhandenen Service und Support seitens des Softwareherstellers als Nachteil von OS/FS an. Für die Implementierung der GDI-Komponenten auf Basis der im Rahmen der vorliegenden Arbeit eingesetzten Softwareprodukte wäre ein Service und Support seitens eines Softwareanbieters nicht notwendig gewesen. Die einzelnen Produkte verfügen über gute Internetdokumentationen sowie aktive Foren und Mailinglists. Im Fall des Apache Webserver sowie des UMN MapServer konnte zudem auf die Erfahrungen des IT-Personals der BGR zurückgegriffen werden.

Auch der Aspekt einer fehlenden Gewährleistung wird nicht als besonderer Nachteil von OS/FS-Produkten gewertet, da – wie in Abschnitt 2.2.1.2 [S. 41] bereits erwähnt – Gewährleistungsverträge mit proprietären Softwareherstellern häufig zugunsten des Softwareanbieters formuliert und schwerwiegende Mängel in der Software von der Gewährleistung ausgeschlossen sind.

¹⁸⁷ HILLEMIEIER et al 2006: 30

¹⁸⁸ HILLEMIEIER et al 2006: 31

Mangelnde Dokumentation

Der in der Literatur¹⁸⁹ hervorgehobene Aspekt einer mangelnden Dokumentation konnte bei den verwendeten OS/FS-Produkten nicht bestätigt werden. Der Einsatz der verschiedenen OS/FS-Produkte erfolgte aus Anwendersicht und unter Nutzung der zum Teil ausgesprochen hilfreichen Projekt-Dokumentationen im Internet. Besonders hervorgehoben sind hier die Seiten zum UMN MapServer mit der ‚Mapfile Reference‘¹⁹⁰ wie auch die zahlreichen Anwendungsbeispiele auf der OpenLayers Homepage¹⁹¹ [vgl. oben].

Allerdings ist zu berücksichtigen, dass bei den verwendeten OS/FS-Produkten keine Änderungen am Source-Code vorgenommen werden mussten. Deshalb kann an dieser Stelle die Dokumentation des Source-Codes nicht bewertet werden.

Umschulungsaufwand

Auch ein ggf. hoher Umschulungsaufwand wird allgemein als ein Nachteil von OS/FS gesehen.¹⁹² Dies kann für die eingesetzten OS/FS-Produkte nicht bzw. nur eingeschränkt beurteilt werden, da diese keine Vorläufer-Produkte ersetzt haben.

Ziel war es, den WMS so aufzusetzen, dass der Datenbezug über eine einfache Datenbankabfrage ermöglicht wird. Bislang wurde zwar eine OS/FS-Datenbank eingesetzt, jedoch ohne räumlichen Aufsatz. Der durch die Einführung einer PostGIS-Datenbank (nicht als Ersatz der bestehenden Datenbanklösung sondern als Ergänzung) entstandene Einarbeitungsaufwand ist grundsätzlich eher nicht als OS/FS spezifisch zu werten, sondern entspricht der ‚normalen‘ Einarbeitung in ein neues Softwareprodukt, wie sie auch in die Produkte UMN MapServer sowie OpenLayers erforderlich war.

Soziale Faktoren (Akzeptanz)

Die zahlreichen Beispiele zum Aufbau von GDIs auf Basis von OS/FS zeigen, dass der Aspekt einer mangelnden Akzeptanz von OS/FS-Produkten im GDI-Umfeld vernachlässigbar ist. Insbesondere im Fachbereich Seismologie der BGR war der Einsatz von OS/FS-Produkten bereits im Vorfeld der hier beschriebenen Anwendungsentwicklung verbreitet.

Zusammenfassende Bewertungsmatrix

Nachfolgend aufgeführte Bewertungsmatrix führt die vorab diskutierten Ergebnisse zusammen.

¹⁸⁹ HILLEMEIER et al 2006: 32

¹⁹⁰ <http://mapserver.gis.umn.edu/docs/reference/mapfile> (Stand: 26.08.2008)

¹⁹¹ <http://openlayers.org/dev/examples/> (Stand: 26.08.2008)

¹⁹² HILLEMEIER et al 2006: 32

| Bewertungsaspekte: | PostGIS | Apache | UMN Map Server | OpenLayers |
|---|--|-------------------|-----------------------|-------------------|
| Als Vorteil bzw. Chance zu bewerten: | | | | |
| <i>Sicherheit, Stabilität und Qualität</i> | (ja) | (ja) | (ja) | (ja) |
| <i>Interoperabilität bzw. Kompatibilität</i> | ja | ja | ja | (x) |
| <i>Bedarfsgerechte Anpassung und Modularität</i> | ja ¹⁹³ | ja ¹⁹⁴ | ja | ja |
| <i>Unabhängigkeit von Softwareanbietern</i> | ja OS/FS kann flexibel gewählt / ersetzt werden | | | |
| <i>Wettbewerbsförderung</i> | 0 | 0 | 0 | 0 |
| <i>Investitionssicherheit</i> | (ja) | (ja) | (ja) | (ja) |
| Als Risiko bzw. Grenze zu bewerten: | | | | |
| <i>Keine Entwicklungsgarantie</i> | nein | nein | nein | nein |
| <i>Mangelnde Unterstützung neuer Hardware</i> | 0 | 0 | 0 | 0 |
| <i>Fehlende Installationsroutinen</i> | ja | ja | ja ¹⁹⁵ | ja |
| <i>Keine Gewährleistung, Service, Support auf Herstellerseite</i> | nein | nein | nein | nein |
| <i>Mangelhafte Dokumentation</i> | <i>allgemein</i> | nein | nein | nein |
| | <i>Source-Code</i> | 0 | 0 | 0 |
| <i>Hoher Umschulungsaufwand</i> | 0 | 0 | 0 | 0 |
| <i>Mangelnde Akzeptanz</i> | nein | nein | nein | nein |

ja = trifft zu; (ja) = trifft mit Einschränkung zu; nein = trifft nicht zu; 0 = keine Bewertung möglich

Tab. 11: Zusammenfassende Bewertung des OS/FS-Einsatzes

4.2. Zusammenfassende Bewertung und Ausblick

Verifizierung der Hypothese

Die eingangs gestellte Hypothese, dass sich *Open-Source und Freie Software besonders gut für den Aufbau von Geodateninfrastrukturen in der öffentlichen Verwaltung eignen*, konnte unter den gegebenen Rahmenbedingungen und durch die im Zuge der Anwendungs-Entwicklung gemachten Erfahrungen bestätigt werden.

Die entwickelte und implementierte Systemarchitektur erwies sich für den Aufbau einer GDI bzw. von GDI-Komponenten als gut geeignet. Somit kann für Erdbebendaten¹⁹⁶ der

¹⁹³ PostGIS stellt bereits eine modulare Erweiterung für PostgreSQL dar, vgl. <http://postgis.refractor.net/> (Stand 28.04.2008)

¹⁹⁴ Vgl. Apache HTTP Server, Abschnitt 3.2.4.

¹⁹⁵ Hier ist anzumerken, dass es unter Windows Installationsroutinen für den UMN MapServer verfügbar sind.

durch die INSPIRE-Richtlinie erhobenen Forderung einer interoperablen und unter einheitlichen Rahmenbedingungen erfolgenden Bereitstellung von behördlichen Geoinformationen für Entscheidungsträger aus Verwaltung, Wirtschaft und Wissenschaft sowie für eine breite Öffentlichkeit entsprochen werden.

Die Implementierung der auf OS/FS basierenden Systemarchitektur wurde durch die organisatorischen, rechtlichen und politischen Rahmenbedingungen in der öffentlichen Verwaltung begünstigt. Insbesondere die Tatsache, dass Softwarelösungen in der öffentlichen Verwaltung nicht, wie in der Privatwirtschaft, zur Erzielung eines Wettbewerbsvorteils eingesetzt werden, trägt zu einer Verbreitung und Weitergabe von Know-how und Nutzungsrechten bei, da keine Verwaltungseinheiten kommerzielle Nachteile befürchten muss. Die beschriebenen Rahmenbedingungen ermöglichen zusammen mit häufig ähnlichen Aufgabenstellungen eine Kooperation bei der Anwendungsentwicklung und beim Einsatz von OS/FS-Produkten.

Neben den günstigen Rahmenbedingungen der öffentlichen Verwaltung waren auch die eingesetzten Softwareprodukte gut für den Aufbau von GDI-Komponenten geeignet. Die erforderlichen Funktionalitäten waren ohne Anpassungen des Source-Codes gegeben. Insgesamt lässt sich feststellen, dass die Konfiguration der einzelnen Produkte ebenso wie die Einrichtung der Schnittstellen zwischen den einzelnen Komponenten ohne über das normale Maß hinausgehende Probleme möglich war.

Hervorzuheben ist aber auch, dass die Umsetzung der Systemlösung durch günstige organisatorische, technische und personelle Rahmenbedingungen erleichtert wurde. Durch eine bereits bis 2007 fast vollständig auf OS/FS umgestellte IT-Infrastruktur und Datenhaltung im SDAC der BGR blieben die mitunter sonst zu erwartenden Mehraufwände in Zusammenhang mit der Umstellung auf OS/FS aus. Bei der Systemimplementierung konnte sogar auf entsprechendes Know-how und Erfahrung beim Einsatz von OS/FS zurückgegriffen werden.

Ausblick

Die im Rahmen der Master Thesis aufgebaute einfache GDI ermöglicht es am SDAC erstmals seismische Parameterdaten und dazugehörige Informationen interoperabel bereitzustellen. Dabei löst es bestehende scriptbasierte Verfahren zur einfachen Kartengeneralisierung ohne Interaktionsmöglichkeit ab. Wie bereits im Abschnitt 3.2.3.2 vorgestellt wurde, soll die GDI zukünftig um weitere seismische und ergänzende Fachdaten aus anderen Disziplinen erweitert werden. Unter Umständen stellen sich dann allerdings, durch das größere Abfragevolumen, erneut Performance-Probleme ein, die es zu lösen gilt.

¹⁹⁶ Gebiete mit naturbedingten Risiken (INSPIRE-Richtlinie, Annex III).

Des Weiteren soll der WMS zu einem WFS ausgebaut werden. Einige der WFS-spezifischen Parameter wurden bereits jetzt berücksichtigt. Durch die Einbindung eines WFS anstelle eines WMS kann auch der OpenLayers-Client um weitere anwenderfreundliche Optionen (z.B. Hervorhebung von abfragbaren Objekten bei Mouseover) erweitert werden.

Quellenverzeichnis

- ADAMS et al. (2004): Praxishandbuch WebGIS mit Freier Software. Bonn
- AK WMS der GDI-DE: GDI-DE. Profil WMS-DE_1.0. Applikationsprofil für Web Map Services innerhalb der Geodateninfrastruktur Deutschland
- Bundesdatenschutzgesetz in der Fassung der Bekanntmachung vom 14. Januar 2003 (BGBl. I S. 66), zuletzt geändert durch Artikel 1 des Gesetzes vom 22. August 2006 (BGBl. I S. 1970)
- BEHNCKE, Kai et.al (2006): Zeichenvorschriftskonforme Darstellung von ALK-Daten im UMN Mapserver unter Verwendung von Open Source Software. Diplomarbeit, Hochschule Vechta
- BEHNCKE, Kai et.al (2007): Die Open Source Geospatial Foundation. In: GIS. Zeitschrift für Geoinformatik. Heft 11/2007, S. 38 – 41
- BERNARD, Lars (2005): Infrastruktur für Geodaten. In: GeoBIT. Heft 9/2005, S. 16 – 19
- BGR, AG MoI (2008): Geofachdateninitiative BGR. Strategie der BGR zur nutzerorientierten Bereitstellung von Geofachdaten der BGR. Teil 1: Beschreibung der IST-Situation (unveröffentlicht)
- BMU, Referat ZG I 6 (2008): Referentenentwurf für ein Gesetz über den Zugang zu digitalen Geodaten (Geodatenzugangsgesetz – GeoZG), Version 2.0 – Stand : 05.02.2008
- BREU, Annette et.al (2007): Geo ohne Grenzen. Statusbericht: GDI nehmen Gestalt an. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 8/2007, S. 28 f.
- CHRISTL, A.; Emde, A. (2008a): Web Mapping mit Open Source GIS-Tools. O'Reilly Verlag, Köln
- CHRISTL, A. (2008b): Open Source Software Entwicklungs- und Kommunikationswerkzeuge. Workshop im Rahmen der FOSSGIS 2008, Freiburg, April 2008
- CHRISTL, A. (2008c): FOSS Geschäftsmodelle. Vortrag im Rahmen der FOSSGIS 2008, Freiburg, April 2008
- CLOS, Patrick et.al. (2007): Standards für OGC-konforme Webdienste der BGR: Pfade und Konventionen für Metadaten. Hannover (unveröffentlicht)
- GDI-DE (2007a): Architektur der Geodateninfrastruktur Deutschland, Version 1.0, 17. August 2007
- GDI-DE (2007b): Architektur der Geodateninfrastruktur Deutschland, Managementfassung, Version 1.0 beta, März 2007
- Deutscher Bundestag (2001): Entschließungsantrag „Nutzung von Geoinformationen in der Bundesrepublik Deutschland“, Drucksache 14/5323
- Deutscher Bundestag (2001): Antrag „Deutschlands Wirtschaft in der Informationsgesellschaft“, Drucksache 14/5246
- GEORGIEV, E et al 2004: Open Source Software – Einsatz in der öffentlichen Verwaltung, veröffentlicht als Schriftenreihe 2 / 2004 des Österreichischen Städtebunds
- FISCHER, Thorsten (2003): UMN MapServer 4.0. Handbuch und Referenz. Map Media GmbH Berlin
- FITZKE, Jens et al Degree-Projekt. Open-Source-Software. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 6/2007, S. 18 f.
- GWOSDZ, Frank (2006): Einsatz von Open Source zur Erstellung von Software. Diplomarbeit, Fachhochschule München
- GRASSMUCK, Volcker (2004): Freie Software. Zwischen Privat- und Gemeineigentum. Bpb, Schriftenreihe Band 458, Bonn
- GREVE, Klaus / KIELE, Christian (2006a): Standards und Normen für die GIS-Welt. Teil 3: Zukünftige Entwicklungen und Standards. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 10/2006, S. 30 – 32
- GREVE, Klaus / KIELE, Christian (2006b): Standards und Normen für die GIS-Welt. Teil 1: Was wird
-

- standardisiert. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 6/2006, S. 28 – 31
- GREVE, Klaus / KIELE, Christian (2006c): Standards und Normen für die GIS-Welt. Teil 1: Was wird standardisiert. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 6/2006, S. 28 – 31
- HEIß, Michael (2005): Wirtschaftsimpulse durch interkommunale Geodateninfrastruktur am Beispiel Bremen und die Region. Vortrag: INTERGEO 2005
- HILLEMEIER, Klaus et al (2006): Open Source Leitfaden Public Sector, “Baden-Württemberg Connected” – Arbeitskreis bwcon:boss Public Sector
- HOGREBE, Daniela (2008): Mehrwertschöpfung durch Integration von OpenGIS in kommunale Prozesse. Master Thesis, Paris Lodron-Universität Salzburg
- Informationsfreiheitsgesetz vom 5. September 2005 (BGBl. I S. 2722)
- Informationsweiterverwendungsgesetz vom 13. Dezember 2006 (BGBl. I S. 2913)
- IMAGI, Interministerieller Ausschuss für Geoinformationswesen (2006): Geodienste im Internet – ein Leitfaden. Frankfurt am Main
- IMAGI, Interministerieller Ausschuss für Geoinformationswesen (2004): Geoinformationen und moderner Staat. Frankfurt am Main
- KIEHLE, Christian: Lokales abstrakt. Geodaten standardisiert integrieren. In: iX. Magazin für professionelle Informationstechnologie. Heft 12/2006, S. 50 – 55
- KLOPFER, Martin (2007): Open Geospatial Consortium. Making location count. Vortrag im Rahmen der Where2B Lösungskonferenz am 29. November 2007 in Bonn
- KLOPFER, Martin / TRAKAS, Athina (2007): Open Geospatial Consortium (OGC). In: WhereGroup: Infobrief Nr. 01/2008, S. 9 – 10
- KÖHLER, Petra / WÄCHTER, Joachim (2004): Lokale Geodateninfrastrukturen: Potentiale und Auswirkungen am Beispiel Brandenburgs. In: SCHRENK, Manfred (Hrsg.): CORP 2004, 25.2.-27.2. Wien, S. 475 – 481
- KUSSBERGER, Jürgen (2005): Konzeption und Implementierung eines OGC-konformen WMS-Dienstes im Client-/Server Umfeld für Aspekte der Bauleitplanung. Master Thesis, Paris Lodron-Universität Salzburg
- MENGE, Falko (2006): Geoinformationsstrukturen – „Geo-Disziplinen“ und ihre Ergebnisse wachsen zusammen. In: Wissenschaftliche Arbeiten der Fachrichtung Geodäsie und Geoinformatik der Universität Hannover, Nr. 258, S. 121 – 132
- MICHEL, Ulrich et. al. (2007): WebGIS kostenlos. Einsatz von Open Source Software. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 1-2/2007, S. 30 – 32
- MOLLER, Erik(2005): Die heimliche Medienrevolution – Wie Weblogs, Wikis und freie Software die Welt verändern. Hannover
- Open Geospatial Consortium Inc. (2003): OGC Reference Model. Version 0.1.3, OGC Doc Nr. 03-040
- Open Geospatial Consortium Inc. (2006): OpenGIS® Implementation Specification for Geographic information - Simple feature access - Part 1: Common architecture, OGC Doc Nr. 06-103r3
- Open Geospatial Consortium Inc. (2007): Styled Layer Descriptor profile of the Web Map Service Implementation Specification. Version 1.1.0, OGC Doc Nr. 05-078r4
- Open Geospatial Consortium Inc. (2006): OpenGIS Web Map Service (WMS) Implementation Specification. Version 1.3.0, OGC Doc Nr. 06-042
- RAMSEY, Paul (2007): The State of OpenSource GIS. Refrations Research Inc. Victoria
- Richtlinie 2007/2/EG des Europäischen Parlamentes und des Rates vom 14. März 2007 zur Schaffung einer Geodateninfrastruktur in der Europäischen Gemeinschaft (INSPIRE). In: ABL EU L 108/1
- SCHÜTZE, Emanuel (2007): Stand der Technik und Potenziale von Smart Map Browsing im Webbrowser am Beispiel der Freien WebMapping-Anwendung OpenLayers. Diplomarbeit, Hochschule Bremen

- SCHWARZBACH, Ines (2007): Software im Test. 1. Teil: Der Testprozess. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 12/2007, S. 25 – 28
- SEIFERT, Markus (2006): Inspire – Geodaten für Europa. 11. Münchener Fortbildungsseminar Geoinformationssysteme. München
- STREUFF, Hartmut J.: EU „inspiriert“ die Geoinformationen. Inspire geht in die nächste Runde. In: GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 1-2/2007, S. 23 – 25
- STROBEL, Josef (2008): Stellenwert von Inspire und GDI. Die Geodatenpolitik ist tot, es lebe die Geodateninfrastruktur. In GIS BUSINESS. Geoinformationstechnologie für die Praxis. Heft 03/2008, S. 45
- Umweltinformationsgesetz vom 22. Dezember 2004 (BGBl. I S. 3704)
- WHEREGROUP GmbH & Co. KG 2008: UMN MapServer Schulung. Schulungsunterlagen. Bonn
- Zettel, Johann (2007): Performance- Analyse und Optimierung von UMN-Mapserverssystemen. Diplomarbeit, Fachhochschule Wiener Neustadt für Wirtschaft und Technik

Internet-Quellenverzeichnis

- APACHE HTTP SERVER PROJEKT: <http://httpd.apache.org>
- BGR: <http://www.bgr.bund.de/>
- DEUTSCHER LANDKREISTAG: <http://www.kreise.de/landkreistag>
- FOSSGIS: <http://www.fossgis.de>
- GDI-DE: <http://www.gdi-de.de>
- GNU: <http://www.gnu.org/>
- GOOGLE MAPS: <http://www.google.com/apis/maps/>
- IMAGI-RP: <http://www.geoportal.rlp.de/>
- INTERNATIONAL SEISMOLOGICAL CENTRE: <http://www.isc.ac.uk/>
- LINUX INFORMATION SYSTEMS AG: <http://debiananwenderhandbuch.de>
- NETCRAFT: <http://news.netcraft.com/>
- OPEN GEOSPATIAL CONSORTIUM INC.: <http://www.opengeospatial.org>
- OPENLAYERS: <http://openlayers.org/>
- OPENSOURCE INITIATIVE: <http://www.opensource.org>
- OPEN SOURCE OBSERVATORY AND REPOSITORY FOR EUROPEAN PUBLIC ADMINISTRATIONS:
<http://www.osor.eu>
- OSGEO: <http://www.osgeo.org/>
- REFRACTIONS: <http://postgis.refrations.net/>
- SELFHTML: <http://de.selfhtml.org/>
- UMN MAPSERVER: <http://www.umn-mapserver.de/>
- UNITED STATES GEOLOGICAL SURVEY: <http://earthquake.usgs.gov/>
- WIKIPEDIA: <http://de.wikipedia.org/>

Anmerkung: Die exakte URL sowie das Besuchsdatum ergeben sich in den Fußnoten der entsprechenden Textabschnitte im Hauptdokument.