



# Master Thesis

im Rahmen des  
Universitätslehrganges „Geographical Information Science & Systems“  
(UNIGIS MSc) am Interfakultären Fachbereich für GeoInformatik (Z\_GIS)  
der Paris Lodron-Universität Salzburg

zum Thema

Welcher Preis ist für die Immobilie angemessen?  
Wie die Verwendung von Open Geo Data bei einem Vergleich helfen kann

vorgelegt von

B. Sc. Maximilian Lohse  
U104337, UNIGIS MSc Jahrgang 2016

Betreuer:  
Prof. Dr. Josef Strobl

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

Berlin, 31.03.2019



---

# Vorwort

Seit Beginn meines vorangegangenen Geographie-Studiums beeindruckten mich die vielfältigen Anknüpfungspunkte unterschiedlicher Themenbereiche an das Thema Geoinformatik/GIS. Die Chance, sich weiterführend intensiv mit anderen Fachgebieten auseinanderzusetzen, ermöglicht es mir seither neue, interessante und spannende Erfahrungen zu sammeln. Im Zuge der vergangenen Etappe haben mich die folgenden Personen begleitet, bei denen ich mich herzlich bedanke möchte:

Ich danke dem UNIGIS Lehrgangsteam für die ausgesprochen sympathische, kompetente und verlässliche Betreuung während des Studiums. Ein besonderes Dankeschön geht dabei an Prof. Dr. Josef Strobl, der mir mit professionellen Ratschlägen während der Bearbeitung dieser Master Thesis zur Seite stand.

Für die stetige Unterstützung bei der Durchführung meines berufsbegleitenden Studiums bedanke ich mich bei meinen Kolleginnen und Kollegen der interfaceMA GmbH.

Ein besonderer Dank gilt meiner Familie und allen Freunden, die mir bei der Durchsicht der Arbeit kritische Verbesserungsvorschläge unterbreiteten.

Mein allergrößtes Dankeschön möchte ich meiner zukünftigen Frau Maria aussprechen, die mich während des gesamten Studiums stets unterstützt hat und mir immer Kraft für sämtliche Herausforderungen gibt.

# Kurzfassung

Für eine Immobilie ist die Lage ein entscheidender Parameter in Bezug auf Wert und Nutzbarkeit. Standortabhängige Eigenschaften, wie die Anbindung an öffentliche Verkehrsmittel, die Lärmbelastung oder die Nähe zu Grünflächen, spielen eine maßgebliche Rolle für die Attraktivität und den Preis einer Immobilie. In den deutschen Immobilienbewertungsrichtlinien wird die Lage als ein relevanter Aspekt hervorgehoben. Zusätzlich steigt durch unterschiedliche Initiativen und die Umsetzung von europäischen sowie nationalen Richtlinien das Angebot an frei zugänglichen Geodaten.

Diese Arbeit untersucht welche wertbeeinflussenden Indikatoren mithilfe von „Open Geo Data“-Services in Berlin abgerufen und miteinander verschnitten werden können. Hierfür wird die aktuelle Entwicklung freier Geodaten in Deutschland betrachtet und eine Machbarkeitsanalyse vorgenommen. Ferner wird eine dokumentierte Webschnittstelle implementiert, mit deren Hilfe es möglich ist, für ein Referenzobjekt anhand der Koordinaten geeignete Vergleichsobjekte mit ähnlichen Lageeigenschaften zu identifizieren. Der Aspekt der Lage einer Immobilie soll hierdurch beispielsweise bei dem Vergleichsverfahren der Immobilienwertermittlung automatisiert berücksichtigt werden können.

Es wird aufgezeigt, dass relevante Dienste, die sich für eine Einbindung eignen, im Untersuchungsgebiet verfügbar sind. Für diese Arbeit wurde eine REST-Webschnittstelle entwickelt, die empirische Untersuchungen zur Nutzbarkeit und der Performanz ermöglichte. Die Leistung der ausgewählten *Web Feature Services* wird für den Anwendungsfall als ausreichend eingeschätzt, wenngleich Aspekte wie partiell schwer auffindbare Informationen, temporäre Ausfälle und Verzögerungen oder die zeitliche Fortschreibung der Daten in Teilen verbesserungsfähig sind. Außerdem kann mithilfe der Overpass API eine unkomplizierte Einbindung von Daten aus dem Gemeinschafts-Projekt OpenStreetMap (OSM) erreicht werden. In Hinblick auf die angestrebten Open-Data-Entwicklungen der Politik lässt sich feststellen, dass diese noch nicht ausreichend umgesetzt worden, auch wenn sich eine positive Entwicklung hin zu einer Vernetzung offener Geodatendienste abzeichnet.

---

# Abstract

For a real estate property, the location is a crucial parameter for the value and usability. Location-dependent characteristics, such as the access to public transport, noise pollution or proximity to green spaces play a significant role concerning the attractiveness and price of a property. Germany property valuation guidelines highlight the location as a relevant aspect. In addition the availability of freely accessible geodata increases through various initiatives and the implementation of European and national guidelines.

This thesis investigates which value-influencing indicators can be queried and blended using „Open Geo Data“-Services in Berlin. For this purpose, the current development of free geodata in Germany is considered and a feasibility study is done. A web-API is implemented to find appropriate comparison objects for a reference object that have similar location characteristics using their coordinates. The aspect of the location of a property should thereby be able to be taken into account automatically when determining a comparative value.

It could be shown that relevant services that are suitable for integration are available in the study area. A REST web interface was developed that allowed empirical studies on usability and performance. The performance of the chosen *Web Feature Services* was considered sufficient for the use case, although aspects such as partially hard to find information, temporary downtime and delays or time series of the data in parts can be improved. In addition, an uncomplicated integration of data from the community project OpenStreetMap could be achieved via the Overpass API. With regard to the intended open data developments in politics, it can be stated that these have not yet been sufficiently implemented, even though there is a clear trend towards the networking of open spatial data services.

# Inhaltsverzeichnis

<b>Vorwort</b>	<b>I</b>
<b>Kurzfassung</b>	<b>II</b>
<b>Abstract</b>	<b>III</b>
<b>Abbildungsverzeichnis</b>	<b>VI</b>
<b>Tabellenverzeichnis</b>	<b>VII</b>
<b>Abkürzungsverzeichnis</b>	<b>VIII</b>
<b>1 Einführung</b>	<b>1</b>
1.1 Ausgangslage . . . . .	1
1.2 Zielsetzung und Forschungsfragen . . . . .	2
1.3 Lösungsansatz . . . . .	3
1.4 Abgrenzung der Arbeit . . . . .	3
1.5 Struktur der Arbeit . . . . .	4
<b>2 Grundlagen</b>	<b>6</b>
2.1 Der Berliner Wohnungsmarkt . . . . .	6
2.2 Immobilien . . . . .	7
2.3 Das Vergleichswertverfahren . . . . .	12
2.4 Open Geo Data und deren Bedeutung . . . . .	16
2.4.1 Open (Geo) Data . . . . .	16
2.4.2 INSPIRE . . . . .	22
2.4.3 PSI und IWG, IFG und UIG . . . . .	26
2.4.4 OpenStreetMap . . . . .	27
2.5 Nutzungsregelungen und Lizenzen . . . . .	29
2.6 Standardisierung von Geoinformationen . . . . .	34
2.6.1 Open Geospatial Consortium . . . . .	34
2.6.2 Web Map Service (WMS) . . . . .	35
2.6.3 Web Feature Service (WFS) . . . . .	36
2.6.4 Web Processing Service (WPS) . . . . .	39
2.6.5 Formate . . . . .	39
<b>3 Dienste und Datensätze</b>	<b>51</b>
3.1 Dienste und Datensätze . . . . .	51
3.2 Overpass API . . . . .	56
<b>4 Schnittstellenimplementierung</b>	<b>58</b>

4.1	RESTful HTTP . . . . .	58
4.1.1	Alternative SOAP . . . . .	59
4.1.2	HTTP Methoden . . . . .	60
4.1.3	HTTP Status Codes . . . . .	62
4.2	JSON Schema Validator . . . . .	64
4.3	Verwendete Tools . . . . .	66
4.4	API-Aufbau . . . . .	72
4.5	Funktionsweise . . . . .	75
<b>5</b>	<b>Ergebnisse</b>	<b>80</b>
<b>6</b>	<b>Fazit</b>	<b>85</b>
6.1	Zusammenfassung . . . . .	85
6.2	Diskussion . . . . .	86
6.3	Ausblick und Ansätze für weitere Forschungen . . . . .	88
	<b>Literaturverzeichnis</b>	<b>98</b>
<b>A</b>	<b>Anhang</b>	<b>I</b>
A.1	Auswahl der implementierten Endpunkte . . . . .	I
A.2	JSON Schemata . . . . .	II
A.2.1	ComparableObject . . . . .	II
A.2.2	Rate-Request . . . . .	III
A.2.3	Beispielinstanz eines Rate-Requests . . . . .	IV
	<b>Eidstattliche Erklärung</b>	

# Abbildungsverzeichnis

1.1	Vereinfachtes Schema der Schnittstelle . . . . .	3
2.1	Immobilienarten nach Nutzung . . . . .	9
2.2	Der Preis der Aussicht . . . . .	11
2.3	Ablaufschema Vergleichswertverfahren . . . . .	15
2.4	Open Geo Data als Venn-Diagramm . . . . .	18
2.5	Open Data Potenzial . . . . .	20
2.6	Akteure eines Open Data Systems . . . . .	21
2.7	Domänen von wiederverwendeten Open Data . . . . .	22
2.8	Grundsätze der Nationalen Geoinformations-Strategie . . . . .	25
2.9	Entwicklung der OpenStreetMap Mitwirkenden . . . . .	28
2.10	Service orientierte Architektur . . . . .	35
2.11	Beispiel eines WMS Dienstes . . . . .	36
2.12	UML-Diagramm des Beispiel XML-Schemas . . . . .	42
2.13	JSON Objekt . . . . .	45
2.14	JSON Array . . . . .	45
3.1	Grünversorgung . . . . .	53
3.2	Lärmkarte . . . . .	55
3.3	Luftbelastung . . . . .	55
3.4	Doppelte Vorkommnisse in OSM . . . . .	57
4.1	Beispiel mehrerer Git-Banches . . . . .	67
4.2	Endpunkte der Schnittstelle in Swagger UI . . . . .	69
4.3	Architektur der Schnittstelle . . . . .	72
4.4	Visualisierung von Luftliniendistanzen . . . . .	77
4.5	Rating-Resultate der implementierten API . . . . .	79
5.1	Abrufzeiten der Overpass Daten . . . . .	81
5.2	Abrufzeiten der WFS Daten . . . . .	82
6.1	4D-Daten im Format NetCDF . . . . .	86

# Tabellenverzeichnis

2.1	Anzahl INSPIRE-betroffener Geodatensätze und -dienste . . . . .	24
2.2	Creative Commons Lizenzen . . . . .	31
2.3	Formate . . . . .	40
2.4	Validierungs-Schlüsselwörter im JSON Schema . . . . .	48
3.1	Auswahl relevanter WFS Dienste und OpenStreetMap POIs . . . . .	52
4.1	HTTP Methoden . . . . .	62
4.2	HTTP Status Codes . . . . .	64
4.3	Java JSON Schema Validatoren . . . . .	65
4.4	Ergebnisse des Benchmark Tests . . . . .	65
4.5	Ergebnisse des verifizierenden Benchmark Tests . . . . .	66
4.6	Entfernungsvergleich nach Klassen . . . . .	78
4.7	Entfernungsvergleich nach Differenzen . . . . .	78
5.1	Deskriptive Statistik der Abfragezeiten . . . . .	81
5.2	Deskriptive Statistik der Parsingzeiten . . . . .	82

# Abkürzungsverzeichnis

<b>API</b>	Application Programming Interface, auch Anwendungsprogrammierschnittstelle
<b>BelWertV</b>	Beleihungswertermittlungsverordnung
<b>EU</b>	Europäische Union
<b>EGovG</b>	E-Government Gesetz
<b>GDI</b>	Geodateninfrastruktur
<b>GIS</b>	Geoinformationssystem
<b>GML</b>	Geography Markup Language
<b>HTTP</b>	Hypertext Transfer Protocol
<b>ImmoWertV</b>	Immobilienwertermittlungsverordnung
<b>INSPIRE</b>	Infrastructure for Spatial Information in Europe
<b>JSON</b>	JavaScript Object Notation
<b>NGIS</b>	Nationale Geoinformations-Strategie
<b>OGC</b>	Open Geospatial Consortium
<b>OKFN</b>	Open Knowledge International
<b>OSM</b>	OpenStreetMap
<b>POI</b>	Point Of Interest
<b>REST</b>	REpresentational State Transfer
<b>SOAP</b>	Simple Object Access Protocol
<b>TEGoVA</b>	The European Group of Valuers' Associations
<b>URL</b>	Uniform Resource Locator
<b>URI</b>	Uniform Resource Identifier
<b>WebGIS</b>	webbasiertes Geoinformationssystem
<b>WFS</b>	Web Feature Service
<b>WMS</b>	Web Map Service
<b>WPS</b>	Web Processing Service
<b>WSDL</b>	Web Service Description Language
<b>XML</b>	eXtensible Markup Language

---

# 1 Einführung

## 1.1 Ausgangslage

Für eine Immobilie ist die Lage ein entscheidender Parameter für den Wert und die Nutzbarkeit. Lageabhängige Eigenschaften, wie z. B. die Anbindung an öffentliche Verkehrsmittel, die Nähe zu Grünflächen oder Versorgungseinrichtungen sowie die Bevölkerungsstruktur der Nachbarschaft, spielen eine entscheidende Rolle für die Attraktivität und den Preis einer Immobilie (PLATZ, 2013; DIN, HOESLI und BENDER, 2001). So macht die Kriteriengruppe „Standort“ bei dem Europäischen Objekt- und Markt-rating von The European Group of Valuers' Associations (TEGoVA) einen Anteil von 30% bei der Immobilienart Wohngebäude aus (TEGoVA 2003; TEGoVA 2016). Geoinformationen spielen in der Immobilienwirtschaft somit eine entscheidende Rolle und gewinnen durch die verstärkte Nutzung von Geoinformationssystemen (GIS) an immer größerer Bedeutung in Bezug auf effiziente Datenverarbeitung (STINGLWAGNER und NEUNDÖRFER, 2007).

Zeitgleich steht durch das Aufkommen unterschiedlicher Open Data Initiativen ein stetig wachsendes Angebot an frei verfügbare Datenquellen mit geographischem Bezug zur Verfügung – beispielsweise mit Kennzahlen zur Analyse der Mikro- und Makrolage (KALASEK und WENINGER, 2015). In der Bundeshauptstadt Berlin werden seit dem 1. Oktober 2013 die amtlichen Geodaten der Berliner Vermessungsverwaltung im Berliner Open Data Portal kostenfrei zur Verfügung gestellt (FRIEDT und LUCKHARDT, 2014; SEUSS, 2015). Während sich mit der öffentlichen Bereitstellung die Abrufe von Kauffällen aus der Automatisierten Kaufpreissammlung des Gutachterausschusses für Grundstückswerte in Berlin verachtfacht haben, wurden Bodenrichtwerte etwa drei Mal so häufig abgerufen (Stand 2014) (FRIEDT und LUCKHARDT, 2014). Nicht ohne Grund werden Open Data als der Rohstoff des 21. Jahrhunderts bezeichnet, denn von ihnen geht ein großes wirtschaftliches Potenzial aus – nach KUZEV (2016) liegt dieses je nach verfolgter nationaler Strategie zwischen 12,1 und 131,1 Mrd. € jährlich.

Durch die automatisierte Verarbeitung offener Geodaten-Schnittstellen könnte der Research-Aufwand bei der Analyse von Objekten in Immobilienbeständen in Bezug auf vergleichbare Lageeigenschaften, beispielsweise bei einer Immobilienwertermittlung, stark reduziert werden (STINGLWAGNER und NEUNDÖRFER, 2007). Voraussetzung für eine solche Optimierung ist die ausreichende Umsetzung der Konzepte offener Datenportale, deren Standards sowie eine gute Qualität der Daten und Performanz der Webdienste (M. JANSSEN, CHARALABIDIS und ZUIDERWIJK, 2012; BERNARD, FITZKE und R. M. WAGNER, 2004).

Aus diesem Grund soll sich in dieser Arbeit mit den wertbeeinflussenden Indikatoren der Lage einer Immobilie sowie der Nutzbarkeit und Leistungsfähigkeit der offenen Geodatendienste auseinander gesetzt werden. Der Nutzen für die Immobilienbranche liegt darin, dass das häufig genutzte Vergleichswertverfahren (siehe Kapitel 2.3) durch die Automatisierung effizienter und detaillierter durchgeführt werden könnte. Die Güte der Resultate kann optimiert werden, da eine Vielzahl

an Datensätzen herangezogen und automatisiert schneller miteinander verglichen werden kann, als dies im manuellen Vergleich möglich wäre. Direkte Immobilienmarktdaten (Verkaufsfälle) sind oft unzureichend vorhanden, wodurch der Vergleich der relevanten Standortqualität eine hohe Bedeutung erhält (VOSS und BAKKER, 2017). Aus der Sicht der Geoinformatik ist der Anwendungsfall interessant, um Erkenntnisse zu einer produktive Nutzbarkeit von standardisierten Geodatendiensten in Berlin zu gewinnen und diese in die aktuellen politischen Entwicklungen einzuordnen.

## 1.2 Zielsetzung und Forschungsfragen

Das Hauptziel der Arbeit ist es, eine Webschnittstelle zu implementieren, die innerhalb eines übertragenen Immobilienbestands für ein Referenzobjekt geeignete Vergleichsobjekte ermittelt. Hierbei sollen (halb-)automatisch anhand der Position die wertbeeinflussenden Standortindikatoren der Mikro- und Makrolage verglichen werden, um vergleichbare Immobilien zu identifizieren. Dabei soll auf offene Geodaten-Quellen für das Untersuchungsgebiet der Stadt Berlin zurückgegriffen werden. Durch das stetig wachsende Angebot an offenen Geodaten der Stadt Berlin steht eine verwertbare Anzahl an Diensten zur Verfügung.

- Welche Kriterien lassen sich für einen Vergleich auf Grundlage der geographischen Lage heranziehen, um geeignete Vergleichsobjekte innerhalb von Immobilienbeständen zu ermitteln?
- Welche relevanten Datensätze für den Wohnmarkt sind hierfür über offene Geodatendienste abrufbar?
- Wie sind die Leistung, die Verfügbarkeit und die Qualität dieser Open Geo Data Services für den konkreten Anwendungsfall einzuschätzen?

Diese Fragen sollen im Zuge dieser Arbeit beantwortet werden, um in der Immobilienwirtschaft mit den daraus resultierenden Ergebnissen eine Datengrundlage zu schaffen, die beispielsweise innerhalb des Vergleichswertverfahrens der Immobilienwertermittlung ergänzend genutzt werden kann. Im Bereich der Geoinformatik sind Erkenntnisse in Hinblick auf die aktuelle Open Geo Data Situation und die produktive Nutzbarkeit von standardisierten Geodatendiensten zu erwarten.

Durch die Verschneidung von wertrelevanten Indikatoren mit Raumbezug soll die Qualität der Lage besser vergleichbar und bewertbar gemacht werden. Hierdurch sollen z. B. in kaufpreisarmen Lagen auch solche Standorte für Vergleichswerte herangezogen werden, die andernfalls beispielsweise auf Grund der Distanz bei einer üblichen Betrachtung vermutlich nicht herangezogen werden würden, obwohl Ähnlichkeiten in Hinblick auf wertbeeinflussende Indikatoren gegeben sind. Durch die Automatisierung der Verarbeitung und die Verwendung von Diensten statt Rohdaten soll die Nutzbarkeit erleichtert, der benötigte Ressourcenbedarf reduziert sowie eine redundante Datenhaltung vermieden werden.

### 1.3 Lösungsansatz

Die Qualität der Lage, ihre Ermittelbarkeit mithilfe von Open Geo Data und ihre Vergleichbarkeit in Hinblick auf den Immobilienpreis stehen im Fokus dieser Arbeit. Um das soeben vorgestellte Ziel zu erreichen und die Forschungsfragen zu beantworten, wird vorerst durch eine Literaturrecherche analysiert, welche Indikatoren für eine Immobilie als wertbeeinflussend gelten und welche lagerelevanten Aspekte für das Vergleichsverfahren eine Rolle spielen. Im Anschluss wird die Open Geo Data Situation in Deutschland und Berlin sowohl allgemein als auch in Hinblick auf diese Kennzahlen untersucht und dabei existierende Lizenzregelungen und Standardisierungen in der Geoinformatik berücksichtigt. Es wird empirisch ermittelt, welche Dienste die zuvor herausgestellten Standortindikatoren bereitstellen um diese im Anschluss in einer neu entwickelten Webschnittstelle zu integrieren. So kann die Nutzbarkeit und die Performanz des Angebots in Hinblick auf die Zielsetzung untersucht werden. Das in Abbildung 1.1 gezeigte vereinfachte Schema visualisiert die Konzeptidee der zu entwickelnden Schnittstelle:

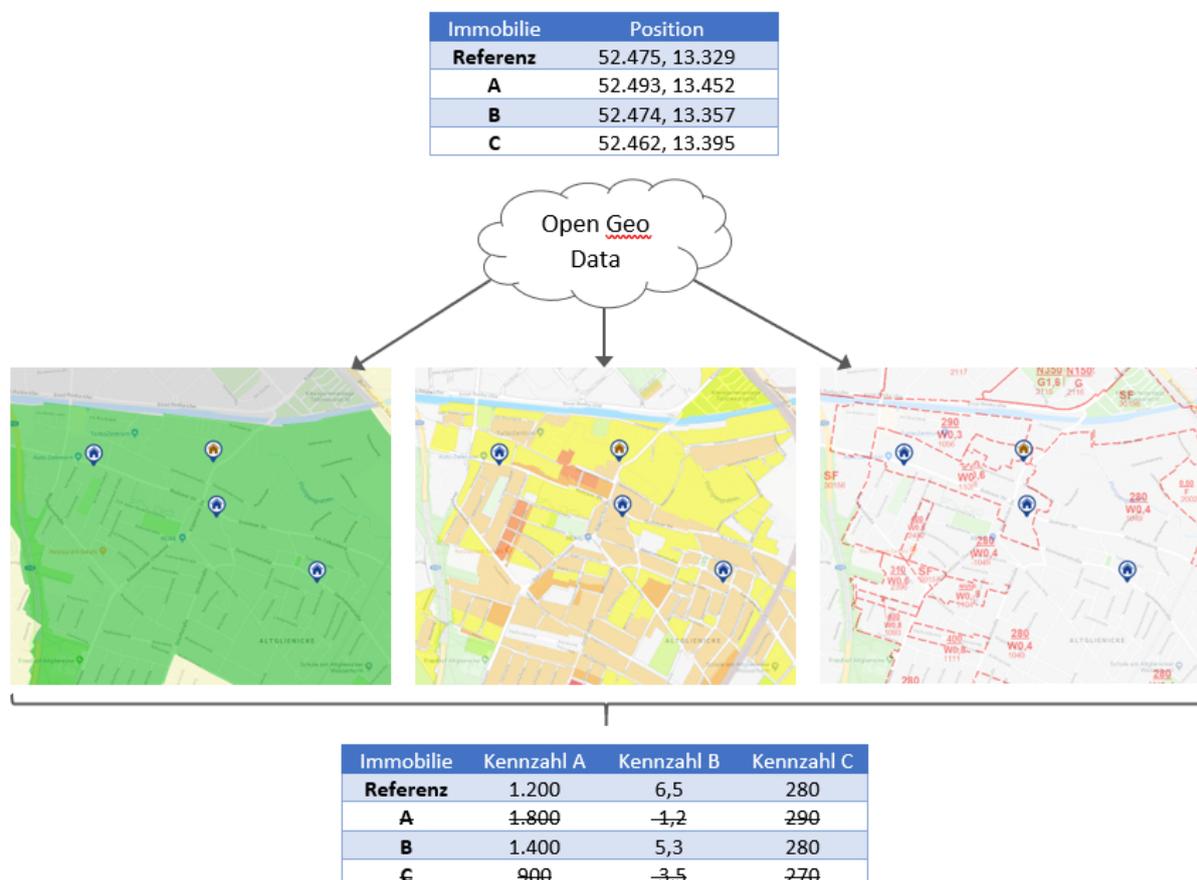


Abbildung 1.1: Vereinfachtes Schema der zu implementierenden Schnittstelle

### 1.4 Abgrenzung der Arbeit

Das Thema dieser Arbeit ist in den Fachbereich der Geoinformatik sowie in den der Immobilienwirtschaft einzuordnen. Da die Lage in Bezug auf eine Immobilie und deren Wert teilmaktabhängig ist (ebd.), wird sich in der vorliegenden Master Thesis primär auf Berliner Wohnimmobilien,

deren Eigenschaften und hierfür relevante Dienste bezogen. Eine vollumfassende Einbeziehung aller verfügbaren Datenquellen und lagerelevanten Indikatoren wird nicht vorgenommen und auch nicht als sinnvoll eingeschätzt. Eine methodische Übertragung auf weitere Immobilienarten wie Gewerbeimmobilien wird grundsätzlich als möglich erachtet, jedoch ist hier in den meisten Bewertungsfällen auch eine detailliertere Einzelfallbetrachtung notwendig und die relevanten Indikatoren unterscheiden sich zu Teilen. Aus diesem Grund werden nur Wohnimmobilien betrachtet und andere Immobilienarten vorerst von der automatisierten Vergleichsobjektanalyse exkludiert.

Im Zuge dieser Ausarbeitung werden keine Aspekte des Themenfelds *Big Data* behandelt, da weder die verarbeiteten Datenmengen dementsprechenden Umfang besitzen, noch Aspekte wie die Analyse oder automatische Ableitung von Informationen aus gleich- oder verschiedenartigen Datenbeständen mithilfe künstlicher Intelligenz für den Anwendungsfall als relevant angesehen werden.

Die Entwicklung dieser Arbeit soll keine automatische Kalkulation eines Vergleichswerts für die Wertermittlung liefern, da die Übermittlung eines Kaufpreises zur Immobilie zum einen optional ist und gegebenenfalls individuelle Zu- und Abschläge vorgenommen werden müssen, die automatisiert nicht ermittelt werden (können). Zum anderen soll es sich bei dieser Arbeit vielmehr um eine Methodenentwicklung und Konzepterprobung handeln, die in weiteren Untersuchungen weiter validiert und ausgebaut werden kann. Auf Grund mangelnder offengelegter Kaufpreise, die zu einer Verifizierung herangezogen werden könnten, und die fehlende Information, welche Vergleichsobjekte gegebenenfalls für eine Vergleichswertanalyse verwendet wurden und zum jeweiligen Stichtag vorlagen, wird von einer konkreten Validierung eines Vergleichswerts und dem erzieltem Immobilienpreis abgesehen.

In einer weiteren Entwicklungsstufe wäre es denkbar, dass eine beliebige Anzahl an optionalen mit dem Objekt verknüpften Parametern über die API vergleichbar gemacht werden und eigene Regeln für das Rating übermittelt werden können (mehr dazu im Ausblick in Kapitel 6.3). Solche Primärdaten werden in dieser Arbeit jedoch nicht berücksichtigt.

Die in den folgenden Kapiteln aufgezeigten allgemeinen Codebeispiele oder Ausschnitte des implementierten Quellcodes dienen der besseren Nachvollziehbarkeit und wurden zu Teilen angepasst oder komprimiert.

## 1.5 Struktur der Arbeit

Im Anschluss an das erste Kapitel, welches in das Thema einleitet, folgt in Kapitel 2 eine komprimierte Vorstellung des Berliner Wohnungsmarkts. Anschließend erhält der Leser eine Einführung in die notwendigen Grundlagen. Neben der Definition einer Immobilie, werden die wertbeeinflussenden Lageeigenschaften einer Immobilie sowie das Vergleichswertverfahren der Immobilienbewertung vorgestellt. Nachfolgend wird auf die aktuelle Open Geo Data Situation im Untersuchungsgebiet eingegangen und in diesem Zusammenhang relevante Lizenzen und Standardisierungen aus dem Bereich der Geoinformationen vorgestellt. Die in diesem Kapitel zusammengetragenen Informationen basieren auf einer Literaturrecherche.

Im dritten Kapitel dieser Arbeit folgt der empirische Teil, bei dem eine Analyse ausgewählter kostenfreier Geodatendienste, die für einen preisbestimmenden Vergleich der Lageparameter Verwendung finden können, durchgeführt wird. Im Anschluss folgt eine kurze technische Einführung in die Welt der REST-APIs und die Präsentation der Schnittstellen-Implementierung zum Auslesen und Verarbeiten der analysierten Datenquellen mithilfe von GIS-Analysen. Codebeispiele untermauern in diesem Kapitel die technischen Hintergründe.

In Kapitel 5 werden die Resultate dokumentiert und analysiert, die sich aus den vorangegangenen Schritten ergeben haben. Diese Ergebnisse werden im letzten Kapitel in Hinblick auf die Forschungsfragen zusammengefasst und diskutiert. Ein Ausblick bezüglich möglicher aufbauender Forschungen sowie genereller Entwicklungen schließt die vorliegende Master Thesis ab.

## 2 Grundlagen

### 2.1 Der Berliner Wohnungsmarkt

Bevor detaillierter auf den Immobilienbegriff, Open Geo Data sowie Standards und rechtliche Grundlagen eingegangen wird, soll der Berliner Wohnungsmarkt kompakt vorgestellt werden. Die bulwiengesa AG hat im Auftrag des BFW Landesverbandes Berlin/Brandenburg e. V. den Wohnungsmarkt der Hauptstadt analysiert und auf Grundlage von 885 Datensätzen eine Wohnungsmarktstudie für den Zeitraum 2016 bis 2021 veröffentlicht. Es wurden solche Projekte berücksichtigt, die ihr Fertigstellungsdatum in den genannten Jahren haben und zu denen mindestens der Entwicklungsstatus, der Wohnungstyp sowie die Investorenart bekannt ist (BULWIENGESA AG, 2018). Obwohl auf Grund der Baulandverknappung, langen Verfahrensprozessen, Bürgerprotesten und steigenden Investitionskosten die Bautätigkeiten in der Hauptstadt aktuell gebremst werden, zeichnet sich eine enorme Dynamik auf dem Berliner Wohnungsmarkt ab. Die Wachstumsphase Berlins basiert vor allem auf Zuzügen und führt zu einer gesteigerten Nachfrage nach Wohnraum (ebd.). Viele Grundstücke haben zwischen 2012 und 2017 einen bis zu siebenfachen Preisanstieg zu verzeichnen und werden aus diesem Grund häufig mit Baurecht weiterverkauft. Gestiegene Miet- und Kaufpreise sind die Folge, die den Wohnraum für die Bürger weniger bezahlbar machen. Auf Grund dessen und auf Grund einer erhöhten Nachfrage vieler temporär in Berlin wohnender Menschen steigt die Anzahl an Bauprojekten mit kleinen Wohnungen – Einzimmerwohnungen haben zwischen 2011 und 2017 mit ca. 14% den größten Anstieg beim Wohnungsbestand zu verzeichnen. Die Anzahl an Wohnungen mit  $\geq 2$  Zimmern stieg im Durchschnitt lediglich um etwa 3% an (ebd.). Auf Grund der Wohnungssituation in Berlin sind die Fortzüge in das Brandenburgische Umland stark gestiegen und liegen aktuell bei mehr als 10.000 Einwohnern jährlich – darunter hauptsächlich Familien mit Kindern und Senioren. Nach BERLIN HYP & CBRE (2019) gleichen sich die Miet- und Kaufpreise von Wohnimmobilien im Berliner Umland immer mehr dem Preisniveau der Hauptstadt an. Trotz allem wächst die Einwohnerzahl der Berlins stetig und mit ihr die Wohnungsnachfrage. Nach BULWIENGESA AG (2018) werden bis 2021 über 85.000 Geschosswohnungen durch Bautätigkeiten realisiert, wobei das Ziel von 20.000 Neubaueinheiten pro Jahr auf Grund der bereits existierenden Anspannungen vermutlich nicht erreicht wird. Dominierend ist der Mietwohnungsbau, dessen Bedeutung weiter steigt. Räumlich konzentriert sich das primär durch private Akteure bestimmte Baugeschehen vor allem auf die östlichen Bezirke (ebd.). Laut der Studie sind die von der Senatsverwaltung für Stadtentwicklung und Wohnen anberaumten 14.000 Wohnungen pro Jahr nicht ausreichend. Für eine Nachfragedeckung seien bis 2030 mindestens 19.000 Wohnungen jährlich notwendig. Um Angebotsüberhänge zu erreichen, müsse der Wohnungsbau von der Politik gefördert, vereinfacht und beschleunigt werden, da es sonst zu verstärkten Wohnungsengpässen und Marktverzerrungen kommen könne (ebd.).

## 2.2 Immobilien

Der Begriff Immobilie stammt aus dem Lateinischen *im-mobilis* und bedeutet „unbeweglich“. Jedoch gibt es keine einheitliche Definition für eine Immobilie und häufig werden synonyme Begriffe wie „Grundstück“, „Gebäude“, „Liegenschaft“ oder „Grundbesitz“ verwendet (BONE-WINKEL, FOCKE und SCHULTE, 2016). BONE-WINKEL, FOCKE und SCHULTE (ebd.) definieren im Zuge des physischen Immobilienbegriffs eine Immobilie als dreidimensionales Gebilde, welches durch Böden, Wände, Decken und Dächer künstliche Flächen und Räume schafft. Es ist auf einem Segment der Erdoberfläche errichtet und trennt „innen“ und „außen“ durch materielle Barrieren voneinander ab. Die Fokussierung des Begriffs ist stark gebäudeorientiert und schließt das Grundstück als solches nicht mit ein. Auch das Bundesamt für Statistik folgt im eidgenössischen Gebäude- und Wohnungsregister einem ähnlichen Vorbild, wenngleich mit konkreteren Kriterien:

«Als Gebäude im Sinne des eidgenössischen Gebäude- und Wohnungsregisters gilt jedes freistehende oder durch eine Brandmauer von einem andern getrennte Bauwerk mit folgenden Eigenschaften:

- Fundament und Dach sowie geschlossene Bauhülle (Mauern)
- eigener Zugang von aussen
- zusätzlich entweder
  - vorgesehen für das Wohnen, mit/ohne wirtschaftliche oder kulturelle Aktivitäten oder
  - Grundfläche von mindestens 20 m<sup>2</sup>, vorgesehen für wirtschaftliche oder kulturelle Aktivitäten

Bei Doppel-, Reihen- und Terrassenhäusern zählt jeder Gebäudeteil als selbstständiges Gebäude, wenn eine Brandmauer existiert.»

Da nicht nur das Objekt alleine, sondern ebenso dessen Nutzbarkeit für eine Einschätzung von Relevanz ist, ist der rein physische Immobilienbegriff für eine umfassende Betrachtung unzureichend (HAASE, 2011). Nach BONE-WINKEL, FOCKE und SCHULTE (2016, S. 15) ergibt sich eine umfassendere Begriffsdefinition:

«Immobilien sind Wirtschaftsgüter, die aus unbebauten Grundstücken oder bebauten Grundstücken mit dazugehörigen Gebäuden und Außenanlagen bestehen. Sie werden von Menschen im Rahmen physisch-technischer, rechtlicher, wirtschaftlicher und zeitlicher Grenzen für Produktions-, Handels-, Dienstleistungs- und Konsumzwecke genutzt.»

Doch was genau zeichnet eine Immobilie aus? Die wesentlichen Eigenschaften, die nach BONE-WINKEL, FOCKE und SCHULTE (ebd.) eine Immobilie von anderen Wirtschaftsgütern unterscheiden, sind:

- **Immobilität:** Wie bereits erwähnt, ist eine Immobilie unbeweglich und daher an einen Standort gebunden. Diese Gebundenheit bestimmt neben den Nutzungsmöglichkeiten auch den ökonomischen Wert der Immobilie. Nach immobilienökonomischer Definition besteht eine Immobilie aus der Kombination der baulichen Anlage und dem Grundstück. Das gleiche Gebäude auf einem anderen Grundstück wäre demnach eine andere Immobilie.

- **Heterogenität:** Die Standortgebundenheit impliziert bereits, dass es keine identischen Immobilien geben kann.
- **Dauer des Entwicklungsprozesses:** Von der Projektidee bis zur Baufertigstellung und der Übergabe an die Nutzer vergehen üblicherweise zwei bis fünf Jahre.
- **Länge des Lebenszyklus:** Nach dem Entwicklungsprozess folgt eine lange Lebenszeit in physisch-technischer und ökonomischer Hinsicht, was Immobilien zu den langlebigsten Wirtschaftsgütern zählen lässt.
- **Höhe des Investitionsvolumen und der Transaktionskosten:** Neben dem hohen Kapitaleinsatz beim Erwerb einer Immobilie fallen beispielsweise bei der Eigentumsübertragung durch Makler- und Notarkosten hohe Transaktionskosten an. Weitere mögliche Kosten entstehen durch die Akquise, Bewertung oder Überwachung des Managements bei indirekten Anlagen wie Fonds.
- **Begrenzte Substituierbarkeit:** Das Gut „Raum“ lässt sich sowohl für die Nutzung Wohnen als auch die gewerbliche Nutzung nur sehr begrenzt durch andere Güter ersetzen. Ein Bedarf lässt sich sowohl zeitlich als auch qualitativ nur aufschieben.

Durch die Standortgebundenheit einer Immobilie ergeben sich *regionale* Teilmärkte, die sich durch Angebot und Nachfrage in verschiedenen Regionen aber auch innerstädtisch zum Teil stark unterscheiden. Durch divergente Nutzungen der Immobilien ergeben sich zusätzlich auch *sachliche* Teilmärkte, da eine Nutzung nicht einfach durch eine andere zu ersetzen ist. Für den Immobilienmarkt generell, also gemeinsam für die unterschiedlichen Teilmärkte, gilt jedoch, dass eine geringe Markttransparenz vorliegt. Da es aufgrund der erwähnten Heterogenität keine zwei gleichen Immobilien gibt, können keine gültigen objektübergreifenden Aussagen getroffen werden. Hinzu kommt, dass die Daten der Immobilien in der Regel privat sind und wenn überhaupt unvollständig oder mit zeitlicher Verzögerung veröffentlicht werden. Trotz der Bemühungen verschiedener Akteure, wie der bulwiengesa AG, Marklerhäusern und Banken, die Markttransparenz zu erhöhen, besteht im Vergleich zu anderen Märkten wie dem Aktienmarkt weiterhin Verbesserungspotenzial (BONE-WINKEL, FOCKE und SCHULTE, 2016).

Der Immobilienmarkt als solcher ist stark von den Entwicklungen der Volkswirtschaft abhängig. Die Nachfrage nach Wohn- und Handelsflächen hängt von der Höhe des Haushaltseinkommens ab und die benötigten Büroflächen werden durch örtliche Beschäftigte im Dienstleistungssektor determiniert. Bestimmt werden diese Einflussfaktoren durch die konjunkturelle Entwicklung. Der erwähnte lange Entwicklungsprozess und die lange Lebensdauer implizieren, dass nur sehr langsam auf Nachfrageveränderungen reagiert werden kann und das Angebot kurzfristig betrachtet eher eine feste Größe darstellt. Hieraus ergibt sich ein zyklisches Auf und Ab, da beispielsweise die fiktive Erhöhung der Flächennachfrage nach einem ursprünglichen Marktgleichgewicht den Mietpreis steigen lässt, da die Flächenmenge kurzfristig nicht erhöht werden kann. Im Umkehrschluss ergeben sich somit höhere Renditen für die Eigentümer und die Attraktivität für Kapitalanleger steigt, weshalb auch die Kaufpreise steigen. Nach einiger Zeit sind die Resultate der verstärkt aktiv gewordenen Projektentwickler zu sehen, die einen erhöhten Gewinn erwirtschaften möchten und das Marktangebot steigt. Hierdurch dreht sich in dem modellhaften Beispiel die Wirkung um und die Preise sinken (ebd.).

Immobilienarten				
Wohnimmobilien	Nicht-Wohnimmobilien			
	Kommerzielle Immobilien			Sonderimmobilien
	Büro-/Verwaltungsimmobilien	Gewerbe-/Industrieimmobilien	Handelsimmobilien	
Ein-/Zweifamilienhäuser	Bürogebäude	Produktionsgebäude	Einzelhandel	Infrastruktur
Mehrfamilienhäuser	...	Lagerhallen	Einkaufszentren	Sozialimmobilien
Eigentumswohnungen (Stockwerkeigentum)		Werkstätten	...	Beherbergungs- und Gaststättenimmobilien
		Distributionszentren		Freizeitimmobilien
		...		...

Abbildung 2.1: Immobilienarten nach Nutzung (Quelle: HAASE (2011))

Aufgrund der Vielzahl verschiedenartiger Immobilien, wird eine systematische Klassifizierung nach Immobilienarten benötigt. Neben einer Einteilung nach Nutzergruppen, wie beispielsweise privaten Haushalten oder Dienstleistungsunternehmen, können Immobilien nach ihrer Funktion gruppiert werden, was den üblichen Standard darstellt (ARENS, 2016). Die Anzahl der Arten variieren jedoch in der Literatur, können aber grob in zwei Hauptklassen gruppiert werden: *Wohnimmobilien*, die überwiegend zu Wohnzwecken verwendet werden und *Nicht-Wohnimmobilien*, welche die unterschiedlichsten Nutzungen aufweisen können. Aus diesem Grund kann hier eine weitere Aufteilung in *kommerzielle Immobilien* und *Sonderimmobilien* vorgenommen werden. Die Sonderimmobilien umfassen beispielsweise Hotels, Krankenhäuser oder Sporthallen. Die kommerziellen Immobilien gliedern sich in Büro-/Verwaltungsimmobilien, Gewerbe-/Industrieimmobilien und Handelsimmobilien (Einzelhandel). Hieraus ergeben sich nach ARENS (ebd.) vier Hauptkategorien von Immobilienarten. Außerdem können sich gemischt genutzte Immobilien ergeben, wenn beispielsweise die Funktion von Arztpraxen und Wohnungen in einer Immobilie vereint sind und relevante Flächenanteile besitzen (ARENS, 2016; HENCKEL et al., 2010).

Da sich für unterschiedliche Immobilienarten divergierende wertbeeinflussende Kriterien alleine durch die Nutzungsabsicht ergeben (PLATZ, 2013), setzt die vorliegende Arbeit den Fokus auf Wohnimmobilien. So hat für eine Lager- oder Produktionshalle beispielsweise eine gut erreichbare Autobahnanbindung einen wertsteigernden Einfluss, während die unmittelbare Nähe bei Wohnimmobilien aufgrund der Lärmbelastung wertmindernde Auswirkungen haben kann (FELDMANN et al., 2016; PLATZ, 2013). Wie bereits erwähnt, sind **Wohnimmobilien** „Gebäude oder Gebäudeteile, bei denen die reine Wohnfunktion im Vordergrund steht“ (ARENS, 2016, S.86). Dabei kann auch bei Wohnimmobilien nach ARENS (ebd.) zwischen verschiedenen Arten differenziert werden:

- **Ein- und Zweifamilienhäuser:** Wohngrundstücke mit ein bzw. zwei Wohnungen in der Form freistehend (kein angrenzendes Nachbargebäude an der Hauswand), Doppelhaus (ein anschließendes Gebäude mit separatem Grundstück) oder Reihenhäuser (mehrere aneinander angrenzende Gebäude auf getrennten Grundstücken)
- **Mehrfamilienhäuser:** Gebäude mit mindestens drei Wohneinheiten (ab 20 Wohneinheiten wird auch von einer **Wohnanlage** gesprochen)

- **Eigentumswohnung:** Wohnung in einem Gebäude mit mehreren Wohneinheiten

Im Jahr 2017 gab es laut dem deutschen Statistischen Bundesamt fast 40,6 Mio. Wohnungen in 19 Mio. Wohngebäuden, wovon etwa 1,9 Mio. Wohnungen auf die Stadt Berlin entfallen. Ca. 93 % der Wohnungen in Berlin besitzen drei oder mehr Räume<sup>1</sup> (STATISTISCHES BUNDESAMT (DESTATIS), 2018). Die Nachfrage nach Wohnimmobilien kann durch unterschiedliche Aspekte geprägt sein. Während die ökonomischen Dimension – das Einkommen – bestimmt, welche Art nachgefragt werden *kann*, beeinflussen sozialstrukturelle Eigenschaften, wie das Alter oder der Haushaltstyp, welcher Wohnraum nachgefragt werden *möchte*. Hinzukommen psychografische Faktoren, wie bevorzugte Wohnformen, Wertorientierungen oder generelle Lebensstile (GONDRING, 2012). Eine Wohnimmobilie kann nach ARENS (2016) somit nicht nur als Wirtschaftsgut, sondern ebenso als Sozialgut verstanden werden.

Es stellt sich die Frage, welche immateriellen lagerelevanten Eigenschaften, zusätzlich zu den strukturellen Eigenschaften sowie der Ausstattung einer Immobilie, wertbeeinflussend sind<sup>2</sup>. Da die inneren Ausstattungsmerkmale einer Wohnimmobilie aktuell (noch) nicht automatisiert per Open Data Services abgefragt werden können, fällt die Konzentration hierbei auf äußerliche Einflussfaktoren. Die Lage wird häufig als wesentlicher wertbestimmender Faktor einer Immobilie genannt, bei derer Betrachtung eine Vielzahl an Kennzahlen mit unterschiedlicher Bedeutung berücksichtigt werden kann (VOSS und BAKKER, 2017). Nach BLÜML (2014), DIN, HOESLI und BENDER (2001), FELDMANN et al. (2016), GLAUSER et al. (2009), MÜRI LEUPP et al. (2011), PLATZ (2013) und VOSS und BAKKER (2017) haben beispielsweise die folgenden Standortindikatoren Einfluss auf den Wert einer Immobilie:

- Zentralität
- Distanz zum nächsten öffentlichen Personennahverkehr (ÖPNV)
- Distanz zu Bildungs- und Gesundheitseinrichtungen sowie Kindergärten
- Distanz zu Wald- bzw. Grünflächen
- Lärm- & Feinstaubbelastung
- Aussicht
- Wassernähe
- Bevölkerungsdichte
- Einkaufsmöglichkeiten sowie viele weitere

DIN, HOESLI und BENDER (2001) benennen besonders die Nähe zu Grünflächen, dem Stadtzentrum sowie das Ruhe-Level als den Immobilienpreis beeinflussende Merkmale. Nach FELDMANN et al. (2016) spielt die infrastrukturelle Zentralität eine entscheidende Rolle. Sie ist speziell für

<sup>1</sup>„Die Zahl der Räume umfasst alle Wohn-, Ess- und Schlafzimmer und andere separate Räume (z. B. bewohnbare Keller- und Bodenräume) von mindestens 6 m<sup>2</sup> Größe sowie abgeschlossene Küchen unabhängig von deren Größe. Bad, Toilette, Flur und Wirtschaftsräume werden grundsätzlich nicht mitgezählt“ (STATISTISCHES BUNDESAMT (DESTATIS), 2018, S.3).

<sup>2</sup>Die Kriterien, welche bei der Wertermittlung durch das Vergleichswertverfahren als wertbeeinflussend übereinstimmen sollten, werden im folgenden Kapitel 2.3 genannt.

Gewerbeimmobilien ein relevantes Kriterium, wengleich auch Wohnimmobilien maßgeblich von der Anbindung an den Straßen- oder öffentlichen Nahverkehr profitieren können. Ferner können Bevölkerungsstrukturen, wie Wanderungstrends in die Peripherie oder das Stadtzentrum, sowie Arbeitslosen- und Ausländeranteile (und deren Entwicklung) bei einer Standortanalyse hinzugezogen werden. Auch die Erwerbsquote als Indikator für die Kaufkraft ist eine Kennzahl von Interesse (ebd.).

Auch das Image einer Nachbarschaft kann als weicher Standortfaktor Einflüsse auf den Wert einer Immobilie haben. Jedoch sind Imagebilder oftmals sehr kleinräumig, orientieren sich an Straßenzügen und sind schwer in Zahlen zu fassen, da sie einen subjektiven Faktor darstellen (ebd.). Zusätzlich gibt es Faktoren wie die Exposition, die in anderen Untersuchungsgebieten eine weitaus größere Rolle spielen als in der flachen Topografie Berlins. So führen Süd- bis Westlagen in vielen Fällen zu einem Mietpreisaufschlag, da sie speziell in Kombination mit großen Hangneigungen – welche erhöhte Baukosten verursachen – für eine gute Besonnung und Aussicht sorgen (MÜRI LEUPP et al., 2011). Die Aussicht einer Immobilie auf Berge, Wasser oder Weite wird neben der Exposition auch von Barrieren in der Sichtachse beeinflusst. Hier wird auf die Master Thesis von HÜRZELER (2012) verwiesen, die auf der Basis von 3D-Stadtmodellen mit unterschiedlichen Detailgraden (LoD 0, LoD 1 und LoD 2) durch 3D-Sichtanalyse interessante Ergebnisse aufzeigte. So konnte am Beispiel Zürichs statistisch nachgewiesen werden, dass die Aussicht auf Gewässer, Berge oder historische Gebäude preissteigernde Auswirkungen hat, während die Sicht auf Industriegebäude zu einer Minderung des Preises führt. Somit kann die Aussicht bei potenziellem Interesse durchaus ein Entscheidungskriterium darstellen (LEOPOLDSBERGER, THOMAS und NAUBEREIT, 2016). Aufgrund des hohen Berechnungsaufwandes und der bedingten Praxistauglichkeit werden in dieser Arbeit jedoch keine Sichtanalysen kalkuliert und berücksichtigt (HÜRZELER, 2012). Zur Untermauerung der großen Bedeutung des Standorts sei



Abbildung 2.2: Der Preis der Aussicht (Quelle: HÜRZELER (2012))

eine empirischen Untersuchung von BLÜML (2014) genannt, die aufzeigte, dass etwa zwei Drittel der in der Studie befragten Investoren sogenannte *Lageorientierte Core-Investoren* darstellen. Der Makro- und noch stärker der Mikrostandort sind mit zusammen fast 30 % relativer Wichtigkeit die bedeutendsten Merkmale für diese Investoren und liegen vor anderen Objekteigenschaften, wie dem Lebenszyklus, der Gebäudeausstattung, der Mieterbonität oder dem Vermietungsstand (ebd.). Eine schlechte Standortentscheidung eines Investors kann nur selten mit wirtschaftlich vertretbarem Aufwand behoben werden (BONE-WINKEL, FOCKE und SCHULTE, 2016).

## 2.3 Das Vergleichswertverfahren

„Bewertungsanalysen dienen der objektiven Ermittlung des Verkehrswertes [einer Immobilie]“ (FELDMANN et al., 2016, S. 396). Dabei können nach eine Vielzahl von Anlässen eine Immobilienbewertung notwendig werden lassen – beispielhaft seien An- bzw. Verkauf, Beleihung, Versteigerung, Versicherungsabschluss oder steuerliche Gründe genannt (LEOPOLDSBERGER, THOMAS und NAUBEREIT, 2016). In Deutschland werden vorwiegend zwei Verordnungen für die Wertermittlung von Immobilien genutzt: die Immobilienwertermittlungsverordnung (ImmoWertV)<sup>3</sup> und die Beleihungswertermittlungsverordnung (BelWertV). Die ImmoWertV enthält basierend auf den Vorgaben des Baugesetzbuches (BauGB) die Grundlagen der Grundstückswertermittlung. Auch wenn die Anwendung der Vorschriften nicht formal verpflichtend ist, ist sie bei den Sachverständigen weit verbreitet, da eine Bewertung ohne ihre Berücksichtigung in der Regel nicht anerkannt wird (SCHMOLL et al., 2015). Im deutschen Recht ist der Verkehrswert nur im § 194 BauGB definiert:

«Der **Verkehrswert** (Marktwert) wird durch den Preis bestimmt, der in dem Zeitpunkt, auf den sich die Ermittlung bezieht, im gewöhnlichen Geschäftsverkehr nach den rechtlichen Gegebenheiten und tatsächlichen Eigenschaften, der sonstigen Beschaffenheit und der Lage des Grundstücks oder des sonstigen Gegenstands der Wertermittlung ohne Rücksicht auf ungewöhnliche oder persönliche Verhältnisse zu erzielen wäre.»

Hieraus ergibt sich, dass für dessen Ermittlung nur die dazugehörige ImmoWertV genutzt werden kann. Als Durchführungsverordnung der §§ 192-194 BauGB muss sie explizit bei den Gutachterausschüssen in Deutschland Verwendung finden. Jedoch sollen auch andere Adressaten aus der Immobilienwirtschaft und dem Kapitalmarkt angesprochen werden. Auch in vielen anderen Bereichen, wie der steuerlichen Bewertung oder im Sachverständigenwesen für Zertifizierungsregelungen, gilt die ImmoWertV für Gerichte als „allgemeiner Stand der Wertermittlungstechnik“<sup>4</sup> (HENDRIX, 2011).

Der Verkehrs- bzw. Marktwert ist zusammenfassend nach FELDMANN et al. (2016) der am wahrscheinlichsten zu erzielende Wert im allgemeinen Grundstücksverkehr. Er gibt dabei aber keinen Mittelwert an, wie es teilweise fälschlicherweise angenommen wird. Nichtsdestotrotz wird der Verkehrswert bei der Wertermittlung nicht als eine mathematisch genau ermittelbare Größe berechnet und stellt alleine auf Grund der in Kapitel 2.2 erwähnten Heterogenität der Immobilien eine Schätzung dar. Unterschiedliche Gutachter können somit zum selben Stichtag zu einem abweichenden Ergebnis kommen. Grundsätzlich kann ein Genauigkeitsgrad von  $\pm 10\%$  angenommen werden, wobei in der Rechtsprechung sogar von  $\pm 20$  bis  $\pm 30\%$  ausgegangen wird (ebd.).

Gemäß § 1 BelWertV finden die Vorschriften dieser Verordnung Anwendung auf die Ermittlung der Beleihungswerte nach § 16 (1) und (2) des Pfandbriefgesetzes (PfandGB) und bei der Erhebung der für die Wertermittlung erforderlichen Daten. Der Beleihungswert, der nach V.-H. MÜLLER (2009) vor allem dem Kreditsicherungsinteresse der Banken dient, wird in § 3 BelWertV wie folgt definiert:

<sup>3</sup>Vorgänger der ImmoWertV war die Wertermittlungsverordnung (WertV). Diese trat 1961 erstmals in Kraft und wurde zuletzt 1997 geändert. Am 01. Juli 2010 löste die ImmoWertV die WertV ab (SCHMOLL et al., 2015).

<sup>4</sup>siehe Urteil BGH vom 12.01.2001 - V ZR 420/99 (BUNDESGERICHTSHOF, o.D.)

«(1) Der Wert, der der Beleihung zugrunde gelegt wird (**Beleihungswert**), ist der Wert der Immobilie, der erfahrungsgemäß unabhängig von vorübergehenden, etwa konjunkturell bedingten Wertschwankungen am maßgeblichen Grundstücksmarkt und unter Ausschaltung von spekulativen Elementen während der gesamten Dauer der Beleihung bei einer Veräußerung voraussichtlich erzielt werden kann.

(2) Zur Ermittlung des Beleihungswerts ist die zukünftige Verkäuflichkeit der Immobilie unter Berücksichtigung der langfristigen, nachhaltigen Merkmale des Objekts, der normalen regionalen Marktgegebenheiten sowie der derzeitigen und möglichen anderweitigen Nutzungen im Rahmen einer vorsichtigen Bewertung zugrunde zu legen.»

Dabei darf nach § 16 (2) PfandBG der Beleihungswert den durch ein anerkanntes Bewertungsverfahren und auf transparente Weise ermittelten Marktwert nicht überschreiten. Die Ermittlung des Marktwerts kann somit beispielsweise durch die Bewertung nach ImmoWertV vollzogen werden (HENDRIX, 2011). Im Vergleich zur Marktwertdefinition der ImmoWertV/des BauGB, bei der ein stichtagbezogener Wert gefordert wird, wird in der BelWertV die zukünftige Entwicklung der Immobilie durch Begriffe wie *nachhaltig* und *langfristig* stark herausgestellt und zielt somit auf einen vorsichtigen Wert für langfristige Darlehen ab (ebd.).

Beide Verordnungen sollen bei der Wertbestimmung in Deutschland als Regelwerk eingesetzt und von den Gutachtern befolgt werden. Wenngleich es leichte Unterschiede in Formulierungen gibt, ähneln sich beide Verordnungen in ihrem Grundsatz stark und sehen die gleichen Methoden der Wertermittlung vor. Geringe Unterschiede gibt es bei der Methode des Vergleichswertverfahrens. Das Vergleichswertverfahren hat bei der Wertermittlung von Immobilien eine große Bedeutung. Laut § 4 (2) der BelWertV ist neben der Ermittlung des Ertrags- und Sachwerts bei Wohnungs- und Teileigentum auch das Vergleichswertverfahren durchzuführen, um einen Kontrollwert für den Beleihungswert zu berücksichtigen. Durch dieses Verfahren kann bei einzelnen Einheiten, die in sich selbständig gewerblich genutzt werden, oder bei Eigentumswohnungen die Sachwert-Ermittlung entfallen (BUNDESMINISTERIUMS DER JUSTIZ UND FÜR VERBRAUCHERSCHUTZ, 2009). Bevor genauer auf das Vergleichswertverfahren eingegangen wird, sollen das Sachwert- und das Ertragswertverfahren gemäß der §§ 17, 18, 21 ImmoWertV zumindest kurz vorgestellt werden:

«§ 21(1) Im **Sachwertverfahren** wird der Sachwert des Grundstücks aus dem Sachwert der nutzbaren baulichen und sonstigen Anlagen sowie dem Bodenwert [...] ermittelt; [...]

(2) Der Sachwert der baulichen Anlagen (ohne Außenanlagen) ist ausgehend von den Herstellungskosten [...] unter Berücksichtigung der Alterswertminderung [...] zu ermitteln.

(3) Der Sachwert der baulichen Außenanlagen und der sonstigen Anlagen wird, soweit sie nicht vom Bodenwert miterfasst werden, nach Erfahrungssätzen oder nach den gewöhnlichen Herstellungskosten ermittelt. »

«§ 17 (1) Im **Ertragswertverfahren** wird der Ertragswert auf der Grundlage marktüblich erzielbarer Erträge ermittelt. Soweit die Ertragsverhältnisse absehbar wesentlichen Veränderungen unterliegen oder wesentlich von den marktüblich erzielbaren Erträgen abweichen, kann der Ertragswert auch auf der Grundlage periodisch unterschiedlicher Erträge ermittelt werden.

(2) Im Ertragswertverfahren auf der Grundlage marktüblich erzielbarer Erträge wird der Ertragswert ermittelt

1. aus dem [...] ermittelten Bodenwert und dem [...] Reinertrag. [...]

§ 18 (1) Der Reinertrag ergibt sich aus dem jährlichen Rohertrag abzüglich der Bewirtschaftungskosten [...].

(2) Der Rohertrag ergibt sich aus den bei ordnungsgemäßer Bewirtschaftung und zulässiger Nutzung marktüblich erzielbaren Erträgen»

Auch die ImmoWertV nennt in § 8 (1) als erstes Verfahren das Vergleichswertverfahren, welches neben dem Ertragswert- oder dem Sachwertverfahren, für die Wertermittlung heranzuziehen ist. Hierbei soll eins der erwähnten Verfahren oder mehrere in Kombination genutzt werden. Die Wahl ist zu begründen und soll sich nach dem Wertermittlungsobjekt sowie den verfügbaren Daten richten. Dabei sollen die für den Geschäftsverkehr üblichen Vorgehen berücksichtigt werden, wie die Verwendung des Ertragswertverfahrens für Bürogebäude oder das Vergleichswertverfahren für Eigentumswohnungen (V.-H. MÜLLER, 2009). Doch was genau ist das Vergleichswertverfahren? Nach § 19 (1) BelWertV sind zur Ermittlung des Vergleichswerts

«[...] nachhaltig erzielbare Vergleichspreise von Objekten heranzuziehen, die hinsichtlich der maßgeblich ihren Wert beeinflussenden Merkmale, **insbesondere Lage**, Ausstattung und Nutzungsmöglichkeiten, mit dem zu bewertenden Objekt hinreichend übereinstimmen; die Vergleichspreise können aus Kaufpreis- oder anderen Marktdatensammlungen entnommen werden. Von dem so ermittelten Ausgangswert ist ein Sicherheitsabschlag in Höhe von mindestens 10 Prozent in Abzug zu bringen.»

Eine ähnliche Formulierung ist in § 15 (1) der ImmoWertV zu finden. Dort heißt es, dass im Vergleichswertverfahren andere vergleichbare Gebiete herangezogen werden sollen, sofern keine ausreichende Anzahl an Vergleichspreisen in dem Untersuchungsgebiet vorliegend ist. Eine Legaldefinition der genauen Anzahl von „genügend Vergleichspreisen“ oder den Eigenschaften der hinzuzuziehenden „vergleichbaren Gebiete“ erfolgt in der ImmoWertV nicht. Bezüglich der „hinreichend übereinstimmenden Grundstücksmerkmale“ wurde in der Rechtsprechung eine Konkretisierung vorgenommen. Nach LEOPOLDSBERGER, THOMAS und NAUBEREIT (2016) sollen demnach die Kriterien Lage, Größe, Grundstücksgestalt, Erschließungszustand, Art und Maß der baulichen Nutzung sowie die Bodenbeschaffenheit übereinstimmen. In § 4 (4) BelWertV werden für eine Orientierung am Vergleichswert für Ein- und Zweifamilienhäuser mindestens fünf Objekte gefordert, die hinsichtlich der Wohnfläche mit dem Referenzobjekt übereinstimmen. Der Aspekt *Lage* wird in beiden Verordnungen mehrfach erwähnt, wodurch angenommen werden kann, dass die geographische Lage als wertbeeinflussenden Indikator die Vergleichbarkeit maßgeblich beeinflusst und Standortanalysedaten eine Grundlage für die Bestimmung bieten (FELDMANN et al., 2016).

Auf Grund der individuellen Charakteristika von Immobilien müssen Abweichungen bei der Verwendungen des Vergleichswertverfahrens durch Zu- oder Abschläge berücksichtigt werden. Zusätzlich können Marktanpassungen vorgenommen werden. Die Faktoren der einzelnen Aspekte lassen sich zum aktuellen Stand automatisiert schwer bestimmen, sodass hier die Kompetenz der Gutachter gefragt ist. Die durch Anpassungen vorgenommenen Abweichungen der Vergleichspreise

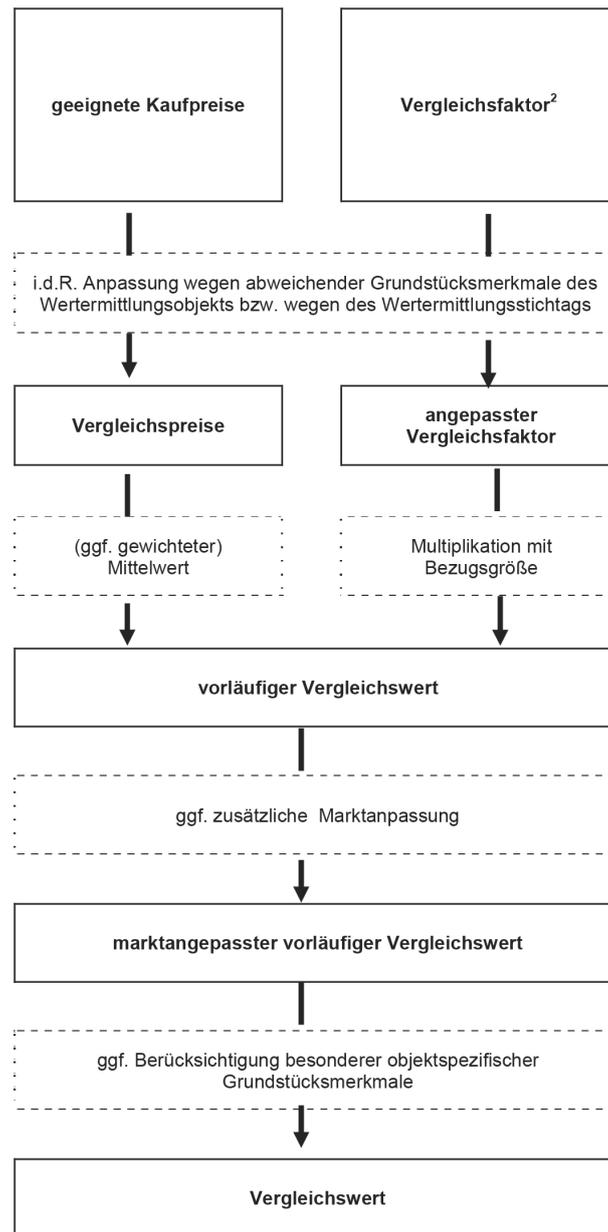


Abbildung 2.3: Ablaufschema Vergleichswertverfahren (Quelle: BUNDESMINISTERIUM FÜR UMWELT, NATURSCHUTZ, BAU UND REAKTORSICHERHEIT (2014))

sollten 30 bis 35 % nicht übersteigen (LEOPOLDSBERGER, THOMAS und NAUBEREIT, 2016). Neben der standortbedingten Heterogenität der Immobilien sind bei der Vorausbestimmung eines exakten Kaufpreis die Angebots- und Nachfragesituation sowie individuelle Wertvorstellungen der Involvierten ein Unsicherheitsfaktor. Hier spielen Punkte, wie der geplante Nutzen oder Spekulationen genauso eine Rolle wie das individuelle Verhandlungs- und Verkaufsgeschick von Käufer und Verkäufer (V.-H. MÜLLER, 2009). Das in Abbildung 2.3 gezeigte Schema visualisiert den Ablauf des Vergleichswertverfahrens.

Die Ermittlung von geeigneten Vergleichsimmobilien spielt somit für die Wertermittlung eine große Rolle. Die ermittelten Werte können nach § 9 (1) ImmoWertV neben offiziellen Kaufpreissammlungen der zuständigen Gutachterausschüsse, auch aus anderen Marktdatenquellen entnommen werden. Daher sind eigens angereicherte Immobilienbestände mit ermittelten Belei-

hungswerten/Kaufpreisen von entscheidender Relevanz, da nicht immer eine ausreichende Anzahl an Kauffällen aus den offiziellen Kaufpreissammlungen vorliegen. Auch in anderen Bereichen der Immobilienbranche abseits der Wertermittlung besitzt die Thematik der Vergleichsobjekte eine Relevanz – beispielsweise um in der Akquisephase potenzieller Projektentwicklungen geeignete Referenzprojekte zu identifizieren, um in einem Immobilienportfolio ähnliche Objekte bezüglich verschiedenster Kennwerte zu analysieren oder für Marklerzwecke.

Nach V.-H. MÜLLER (2009) ist die Vergleichbarkeit von Bewertungs- und Vergleichsobjekten sowie die Aktualität der Vergleichsdaten beim Vergleichswertverfahren als problematisch anzusehen. Der erste Punkt soll im Zuge dieser Arbeit durch die Verwendung unterschiedlicher Datensätze an den geographischen Positionen der Vergleichsobjekte erhöht werden. Die Vergleichbarkeit der Objekte selbst, beispielsweise aufgrund derer Ausstattung, bleibt hierdurch jedoch unberührt. Der zweite Kritikpunkt soll durch die dynamische Verwendung von aktuellen Diensten adressiert werden, wenngleich auch dort zum jetzigen Zeitpunkt ein zeitlicher Versatz unvermeidbar ist (siehe Kapitel 3).

## 2.4 Open Geo Data und deren Bedeutung

Aufgrund der wachsenden Bedeutung von Geoinformationssystem (GIS) in den letzten 10-20 Jahren hat auch die Nachfrage an räumlichen Daten zugenommen (KALASEK und WENINGER, 2015). Das Aufkommen unterschiedlicher Initiativen, die das freie Zurverfügungstellen von Geoinformationen für jedermann fördern, bietet eine Vielzahl von Möglichkeiten und Chancen. Während zu Beginn der Einsatz von GIS vorwiegend Spezialisten vorbehalten war, hat sich durch deren Weiterentwicklung sowie günstigere, leistungsfähigere und einfachere Computer-Systeme der Anwenderkreis stark erweitert. Durch diese Veränderung stieg die Nachfrage nach systemübergreifend austauschbaren Daten, welches wiederum die Entwicklung relevanter Standards zur Folge hatte. Veröffentlichungen frei verfügbarer Werkzeuge – wie GIS-Systeme oder räumliche Datenbanken – sowie die Weiterentwicklungen des Web 2.0, welche die WebGIS-Systeme in die Browser der Nutzer brachten, förderten diesen Impuls nachhaltig (ebd.).

### 2.4.1 Open (Geo) Data

Der Begriff «open» (englisch für offen bzw. frei) wird derzeit viel genutzt und kommt in den verschiedensten Bereichen der Open-Bewegung zum Ausdruck: Open Source, Open Government, Open Learning, OpenStreetMap sowie Open Data und **Open Geo Data**. Mit *offen* ist der freie Zugang zu öffentlich verfügbaren Inhalten gemeint, von dem keine Personen oder Gruppen ausgeschlossen sind (SEUSS, 2015). Sieht man digitalen Daten als Ressource an, werden sie nicht knapp, sobald sie einmal öffentlich verfügbar gemacht wurden, und können nahezu kostenfrei kopiert, verteilt und wiederverwendet werden (KUZEV, 2016). OPEN KNOWLEDGE INTERNATIONAL (2017) nennt in der aktuellen Open Definition Version 2.0 folgende Kriterien, damit ein Werk als frei bezeichnet werden kann:

1. **Zugang:** Das Werk soll vorzugsweise als kostenloser Download als Ganzes bereitgestellt werden und muss zusätzlich sämtliche Informationen zur Einhaltung der Lizenzbestimmungen bereitstellen.
2. **Offenes Format:** Das bereitgestellte Format des Werks muss modifizierbar und zweckdienlich sein. Es dürfen keine unnötigen technischen Hindernisse geschaffen werden, die die Ausübung der genehmigten Rechte einschränken würden. Außerdem muss das Format als solches offen sein – in Form von veröffentlichten und frei verfügbaren Spezifikationen ohne irgendwelche Einschränkungen bei der Nutzung – oder mindestens mit freier/open-source-Software bearbeitbar sein. Ein wichtiger Aspekt ist außerdem die Maschinenlesbarkeit (siehe auch KUZEV (2016)). Weitere Details zu Formaten können Kapitel 2.6.5 entnommen werden.
3. **Offene Lizenz:** Damit eine Lizenz nach OPEN KNOWLEDGE INTERNATIONAL (2017) als offen deklariert werden kann, müssen unter anderem die folgenden Berechtigungen unwiderrufflich erlaubt werden:
  - a) Kostenfreie Nutzung muss ermöglicht sein.
  - b) Weiterverbreitung (alleine oder als Teil einer Sammlung verschiedener Quellen) muss erlaubt sein. Hierunter fällt auch die Möglichkeit des Verkaufs.
  - c) Schaffung von Derivaten der ursprünglichen Arbeit und deren Verbreitung unter gleichen Bedingungen muss erlaubt sein.
  - d) Durch die Verwendung mit anderen eigenständigen Werken entstehen keine Einschränkungen für diese.
  - e) Keine Personen oder Gruppen dürfen diskriminiert werden.
  - f) Der Zweck darf für niemanden beschränkt sein.
  - g) Es dürfen keine Gebühren durch die Lizenz anfallen.

Ferner werden durch die Definitionen der Open Knowledge International einige Bedingungen akzeptiert, die die unter Punkt 3 aufgelisteten Genehmigungen in einem bestimmten Umfang einschränken können. Hierunter fällt z. B. das *Verbot technischer Einschränkung*, bei dem die Lizenz verbieten kann, «[...] dass das Werk in einer Weise verbreitet wird, bei der technische Maßnahmen das Ausüben der sonst erlaubten Rechte einschränken» (ebd.). Ferner kann durch die Lizenz verlangt werden, dass Modifizierungen der lizenzierten Arbeit durch einen anderen Namen oder eine Versionsnummer gekennzeichnet werden oder die Änderungen anderweitig hervorgehoben werden – hierdurch soll *Integrität* geschaffen werden (ebd.).

Die Ansätze der Open-Bewegung beziehen sich generell auf offene Werke, als ein Element oder Stück von Wissen unabhängig des Inhalts. In dem Begriff Open Geo Data ist jedoch explizit der Bereich der Geoinformationen enthalten. Was genau umfasst dieser Teilbereich der offenen Daten? In dem in Abbildung 2.4 dargestellten Venn-Diagramm ergeben sich unterschiedliche Schnittmengen der Teilbereiche Open, Geo und Data. Die erste Schnittmenge *Open Data* umfasst frei zugängliche Datenbestände, die ohne Einschränkungen von der Allgemeinheit weiterverarbeitet und weiterverwendet werden dürfen. Der Überlappungsbereich *Open Geo* umfasst freie

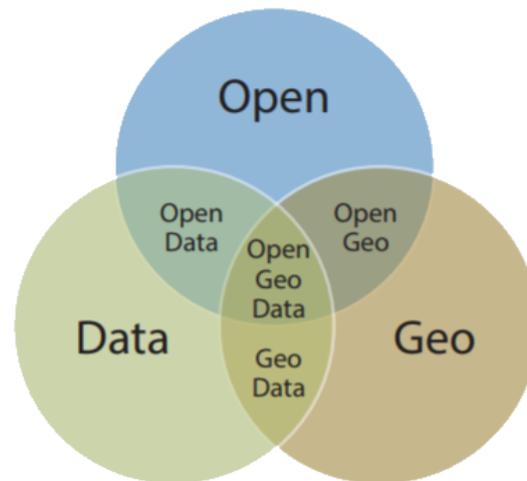


Abbildung 2.4: Open Geo Data als Venn-Diagramm (eigene Darstellung nach SEUSS (2015))

Softwarelösungen, mit deren Hilfe Geoinformationen gespeichert, bearbeitet, analysiert und auch präsentiert werden können. Als Beispiele seien hier die folgenden genannt:

- **PostGIS**: eine räumliche Datenbankeerweiterung für die Datenbank PostgreSQL
- **QuantumGIS (QGIS)**: ein Open Source GIS
- **OpenLayers**: eine Javascript-Bibliothek für Kartendarstellungen aus verschiedenen Quellen

Im dritten Bereich überschneiden sich die Begriffe Geo und Data. Hierunter fallen sämtliche Daten mit direktem oder indirektem Bezug zu einem geographischen Gebiet oder einem bestimmten Standort, beispielsweise aus der Gesellschaft, Wirtschaft oder Verwaltung. Durch die Kombination der einzelnen Schnittmengen ergibt sich für den Begriff *Open Geo Data* die Definition der frei zugänglichen raumbezogenen Daten, die auch mit freier Software verarbeitet werden können (SEUSS, 2015). Da für viele Daten der Raumbezug eine entscheidende Rolle spielt, wird eine Open Data Politik ohne Geodaten als nicht sinnvoll erachtet, vielmehr wächst deren Bedeutung stetig. Die heutige Wissensgesellschaft benötigt für eine Vielzahl von Aufgaben und Entscheidungen Geoinformationen, die einfach nutzbar und wirtschaftlich wirkungsvoll verwendbar sind (BUNDESMINISTERIUM DES INNEREN, 2017). Aus diesem Grund hat die Europäische Union (EU) die INSPIRE-Richtlinie entwickelt, die sich auf europäischer Ebene den Entwicklungen dieses Themenfeldes annimmt (siehe Kapitel 2.4.2). Bei der Betrachtung von Open (Geo) Data sollten diese jedoch nicht nur auf die Bereitstellung von Daten mit einer offenen Lizenz in einem offenen Format reduziert werden, da auch die Auffindbarkeit eine entscheidende Rolle für die Nutzbarkeit spielt (PADERTA, 2012).

Das Politikfeld „Open Data“ ist noch ein sehr junges mit einer hohen Dynamik. Wenn im Zuge der Digitalisierung auf Daten und hierbei speziell auf Open Data Bezug genommen wird, fallen gegenwärtig Begriffe wie „Rohstoff des 21. Jahrhunderts“, „neues Öl“ oder „digitales Gold“, um das wirtschaftliche Potenzial aufzuzeigen. Neben zahlreichen anderen Studien, die einen positiven Zusammenhang zwischen offenen Daten und wirtschaftlichem Wachstum darlegen, konnte eine Studie im Auftrag der Konrad-Adenauer-Stiftung konnte den volkswirtschaftlichen Mehrwert

von offenen Verwaltungsdaten aufzeigen (KUZEV, 2016; MCKINSEY GLOBAL INSTITUTE, 2011). Nach GURIN (2014) kann beispielsweise durch die folgenden Aspekte ein wirtschaftlicher Nutzen entstehen:

- Aufbau von neuen, zuvor nicht möglichen, Geschäftsmodellen durch Firmengründer
- Wirtschaftliche Verwertbarkeit von zentralen Datenbeständen der Verwaltung
- Gesteigertes Vertrauen von Bürgern und Kunden durch erhöhte Transparenz in der Verwaltung und in Unternehmen
- Beschleunigung von wissenschaftlicher Arbeit
- Verbesserte Risikoanalyse und -minimierung durch Investoren
- Verbesserte Entscheidungsfindung durch aufbereitete Daten

Basierend auf einer Vielzahl von Potenzial-Studien zu Open Data und Public Sector Information (PSI) wurden nach KUZEV (2016) drei mögliche Zielbereiche ausgearbeitet, die in den Jahren bis 2026 realistisch erscheinen, sofern die politischen Rahmenbedingungen gegeben sind. Bei dem *konservativen* Fall ergibt sich durch den Ausbau bereits begonnener Aktivitäten im Bereich Open Government Data ein Potenzial von 12,1 Mrd. € jährlich. Der *ambitionierte* Zielbereich umfasst ein Potenzial von 43,1 Mrd. € unter der Annahme, dass Deutschland eine proaktive Strategie („open by default“) verfolgt<sup>5</sup>. In diesem Szenario werden bis zu 20.000 neue Arbeitsplätze durch Open Data als realistisch eingeschätzt (ebd.). Abbildung 2.5 visualisiert anhand des ambitionierten Falls der Studie, dass es der Digitalisierung und Vernetzung aller Organisationen und Gesellschaftsbereiche bedarf, um durch eine digital transformierte Gesellschaft den positiven Nutzen für Wirtschaft, öffentliche Hand und somit der Zivilgesellschaft zu ermöglichen.

Der *optimistische* Fall kann bis zu 131,1 Mrd. € jährlich einbringen. Hierbei wird angenommen, dass die Regierung in der nationalen Entwicklungsstrategie für das 21. Jahrhundert Open Data als Kernkomponente ansieht und somit nicht nur eine digitale Transformation anstrebt, sondern den herausfordernden Anspruch hat, zur weltweit führenden Open Data Nation zu werden (ebd.).

Trotz der bereits begonnenen Maßnahmen befindet sich Deutschland im internationalen Vergleich in Bezug auf den Auf- und Ausbau eines Open Data Ecosystems nur im Mittelfeld. War Deutschland im Jahr 2013 auf dem Open Data Index noch auf Platz 11, reichte es 2015 lediglich für Platz 27, da 16 andere Länder in diesen zwei Jahren deutlich bessere Fortschritte erzielten. Im aktuell vorliegenden Global Open Data Index 2016 konnte Deutschland drei Plätze gut machen und liegt damit aktuell auf Platz 24 von 94 (OPEN KNOWLEDGE INTERNATIONAL, 2019). Aufgrund der aufgezeigten Potenziale von Open Data besteht bei mangelnden Fortschritten hier das Risiko, den internationalen Anschluss an die Entwicklungen zu verlieren (KUZEV, 2016).

---

<sup>5</sup>Durch das Open Data Gesetz (§ 12a E-Government Gesetz (EGovG)) sollen unbearbeitete Daten der Verwaltung standardmäßig offen zur Verfügung gestellt werden. Hierdurch wird ein Grundsatz aufgebaut, der die Richtung „open by default“ einschlägt – ein Paradigmenwechsel. Am 14.10.2016 haben sich auch die Regierungschefinnen und Regierungschefs von Bund und Ländern dazu entschieden, wie die Bundesregierung Open Data Gesetze zu erlassen. Hierdurch sollen bundesweite Standards für öffentliche Zugänge aufgebaut werden. Außerdem nimmt Deutschland an der internationalen Initiative Open Government Partnership (OGP) teil, bei der über 80 Staaten mittels nationaler Aktionspläne im Zweijahresrhythmus ihre politischen Vorhaben austauschen und sich zu den Grundsätzen einer offenen Regierung bekennen (BUNDESMINISTERIUM DES INNERN, 2017).

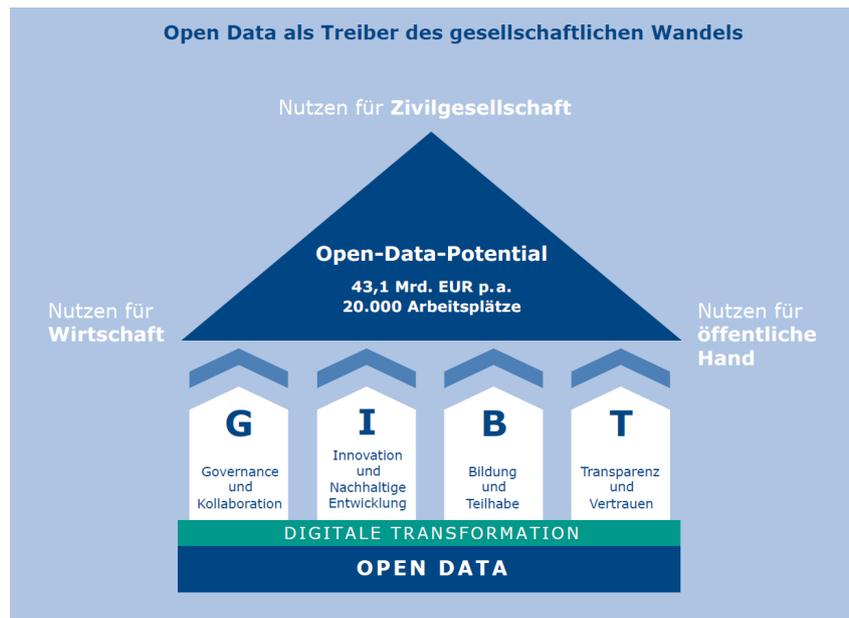


Abbildung 2.5: Open Data Potenzial (verändert nach KUZEV (2016))

Im Zuge ihrer Studie zu erwarteten wirtschaftlichen Auswirkungen von „Open Data“ in Großbritannien, stellten HAMMEL et al. (2012) drei Akteursebenen in einem System offener Daten vor: die Regierung, Wirtschaft und Bürger (siehe Abbildung 2.6). Dabei sei jede einzelne Einheit nicht nur in der Rolle des Nutzers, sondern auch als Datenproduzent zu sehen. Die aktive Partizipation der verschiedenen Anwenderkreise spielt eine entscheidende Rolle bei der Konzeptumsetzung. Verschiedene Bereiche erfahren unterschiedlichen Druck durch Anwender, wodurch sich die Bereitschaft an einer Beteiligung ebenso unterscheidet (ebd.).

Wenngleich sich die aufgezeigten Potenziale nicht ausschließlich auf offene Geodaten beziehen, wird deutlich, welche Chancen sich durch freien Zugang zu offenen Daten ergeben. Laut einer Studie der Europäischen Kommission wurden fünf Haupt-Datendomänen identifiziert, die in der EU als am stärksten nachgefragt gelten, wenn es um die Wiederverwendung von Open Data geht (EUROPÄISCHE KOMMISSION, 2014):

- **Räumliche Daten:** Postleitzahlen, Verwaltungsgrenzen, nationale und lokale Karten etc.
- **Erdüberwachungs- und Umweltdaten:** Überwachungsdaten von Wetter, Land und Wasser, Emissionswerte usw.
- **Statistische Daten:** Demografie- und Wirtschaftsindikatoren, wie Alter, Gesundheit, Arbeitslosigkeit, Einkommen usw.
- **Unternehmensdaten:** Gesellschafts- und Unternehmensregister, Bilanzen etc.
- **Transport- und Infrastrukturdaten:** Fahrpläne des öffentlichen Verkehrs, Straßenarbeiten, Verkehrsinformationen usw.

Bei einer Befragung des *European Data Portal* der Europäischen Kommission wurden 76 Organisationen bezüglich der Wiederverwendung von offenen Daten befragt. Das Ergebnis zeigt, dass der Sektor der statistischen Daten mit 27,3% am häufigsten weiter genutzt wird, gefolgt von den Geodaten mit 25,8%. Fasst man die Erdbeobachtungs- und Umweltdaten mit der Kategorie

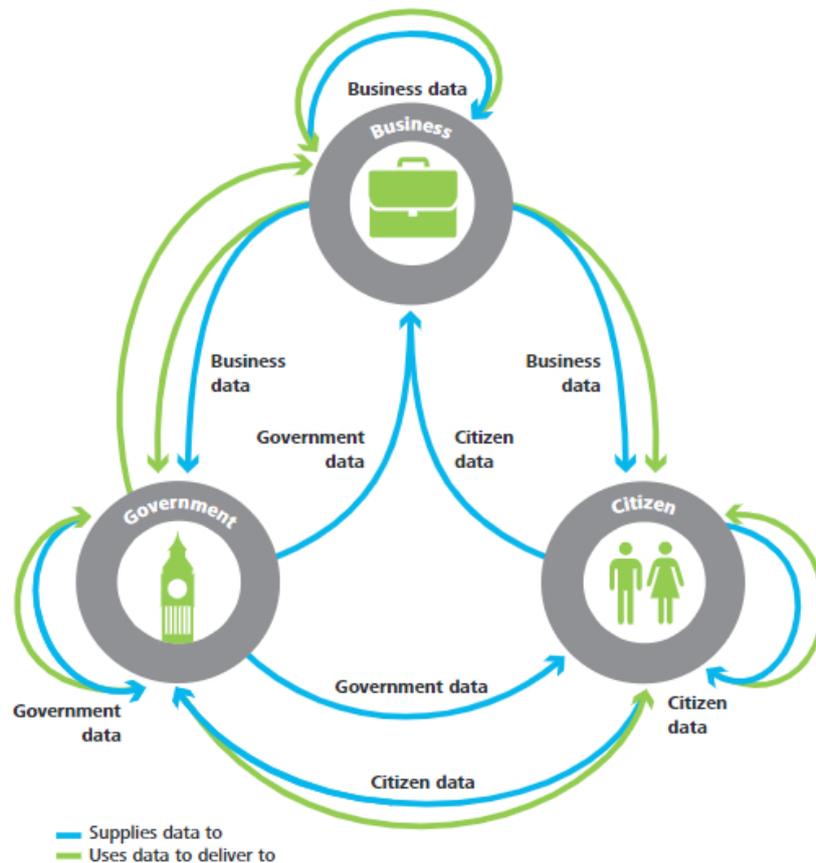


Abbildung 2.6: Akteure eines Open Data Systems (Quelle: HAMMEL et al. (2012))

der räumlichen Daten zusammen, da sie in den meisten Fällen eine exakte oder interpolierte Geolokation enthalten, ergeben sich 34,4% (EUROPÄISCHE KOMMISSION, 2017). Während die Unternehmensdaten für die Betrachtung von Wohnimmobilien und deren Lage vernachlässigbar sind, ist der Sektor „Transport- und Infrastrukturdaten“ ebenfalls von Relevanz. Statistische Daten sowie solche aus dem Bereich Transport und Infrastruktur stehen häufig auch mit einem Geobezug zur Verfügung, sodass auch hier die klare Abgrenzung zu den Geodaten in Frage zu stellen ist. Nichtsdestotrotz verdeutlichen die Ergebnisse in Abbildung 2.7, dass „Open Geo Data“ einen großen Anteil der auf europäischer Ebene verfügbaren Daten, die weiterverarbeitet werden, ausmachen.

Die Entwicklung von Open Data und Open Geo Data kann in Entwicklungsphasen gegliedert werden. Während in der ersten Phase 1.0 die Daten im Rohdatenformat bereitgestellt werden, was häufig unstrukturierte Daten und eine Vielzahl von Formaten zur Folge hat, wird diese Problematik in der darauffolgenden Phase Open Data 2.0 überwunden (SEUSS, 2015). Nach SEUSS (ebd.) befinden sich Deutschland aktuell genau in dieser Phase, in der standardisierte Schnittstellen und Schnittstellenformate zur Datenbereitstellung genutzt werden. Hierdurch erhöht sich die Verlässlichkeit und Geschäftsmodelle können sich entwickeln. In der nächsten Entwicklungsstufe Open Data 3.0 werden auch die Datenmodelle besser standardisiert, wobei sich hierbei die Frage stellt wer sich in der Pflicht sieht diese Entwicklung voran zu treiben (ebd.). Auf die Aspekte der Standardisierungen und Formate in Kapitel 2.6 eingegangen.

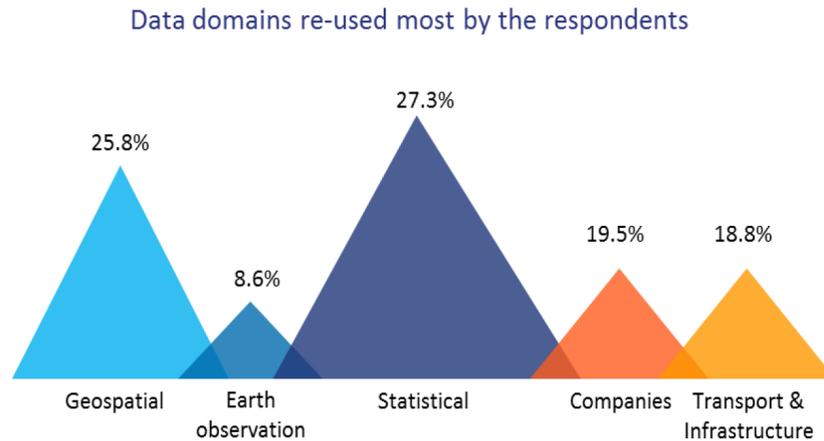


Abbildung 2.7: Domänen von wiederverwendeten Open Data (Quelle: EUROPÄISCHE KOMMISSION (2017))

### 2.4.2 INSPIRE

Mit der Infrastructure for Spatial Information in Europe (INSPIRE)-Richtlinie 2007/2/EG<sup>6</sup> haben sich im Jahr 2007 die damals 27 Mitgliedsstaaten<sup>7</sup> der EU dazu verpflichtet, eine gesamteuropäische und jeweils eine nationale Geodateninfrastruktur aufzubauen, um fach- und grenzübergreifend Geoinformationen als elektronische Daten und Dienste bereitstellen zu können. Konkreter sollen aus allen teilnehmende Staaten bis 2021 zehntausende harmonisierte Datensätze zu 34 Fachthemen verfügbar, leicht ermittelbar und auf ihre Eignung prüfbar gemacht werden. Ferner sollen sie für unterschiedliche Anwendungsfälle kohärent verknüpfbar sein (ILLERT, 2012). Für diese Arbeit relevante Themenbereiche der INSPIRE-Richtlinie sind beispielsweise:

1. Statistische Einheiten
2. Gebäude
3. Gesundheit und Sicherheit (z. B. geographische Verteilung von Erkrankungen sowie Informationen zum Wohlbefinden oder Umweltqualitäten, wie Luftverschmutzung & Lärm)
4. Versorgungswirtschaft und staatliche Dienste (z. B. Schulen und Krankenhäuser)
5. Umweltüberwachung (z. B. Messungen von Schadstoffen oder Informationen zur Vegetation)
6. Demografie

Für andere Untersuchungsgebiete könnte zusätzlich das Themenfeld „Gebiete mit naturbedingten Risiken“, welches z. B. Überschwemmung, Erdbeben und Erdbeben umfasst, von großer Relevanz sein, während es in Berlin auf Grund des geringen Risikos in dieser Arbeit zu vernachlässigen ist.

Eine Anforderung für die Bereitstellung von Geodaten ist nach der Richtlinie 2007/2/EG (6), dass die Nutzungsbedingungen (siehe Kapitel 2.5) leicht feststellbar sind und diese einer vielseitigen

<sup>6</sup>Siehe EUROPÄISCHES PARLAMENT UND RAT (2007)

<sup>7</sup>2007 waren 27 Mitgliedsstaaten in der EU. Am 01.07.2013 ist zusätzlich Kroatien beigetreten, sodass aktuell 28 Länder der EU angehören.

Nutzung nicht unangemessen im Wege stehen. Um die Hürden einer umfassenden Nutzung der Geodaten zu minimieren, sollen die Mitgliedstaaten nach 2007/2/EG (15) Metadaten zu den verfügbaren Daten und Diensten veröffentlichen. Ferner wird erwähnt, dass Netzdienste für die gemeinsame Nutzung notwendig sind und Mindestleistungskriterien erfüllen müssen, um Interoperabilität zu ermöglichen (siehe 2007/2/EG (17)). Explizit werden in Artikel 11 für Services die Typen Such-, Darstellungs-, Download- und Transformationsdienste genannt. Grundsätzlich soll es auch Dritten möglich sein, unter Einhaltung der Kriterien leichter Nutzung ihren Beitrag zu der Infrastruktur beizusteuern.

Bei der Umsetzung von INSPIRE liegt die Federführung in Deutschland beim Bundesministerium für Umwelt, Naturschutz und Reaktorsicherheit (BMU), welches sich mit den Ländern und kommunalen Spitzenverbänden abstimmt und die Bundesrepublik bezüglich ihrer Interessen beim Regelungsausschuss vertritt (ebd.). Von Bund, Ländern und Kommunen wurde die Geodateninfrastruktur Deutschland (GDI-DE) aufgebaut – durch sie werden Geodaten über das Internet frei zugänglich zur Verfügung gestellt. Über das sogenannte Geoportal lassen sich diese Daten finden, darstellen und deren Einbindungsmöglichkeiten über Dienste feststellen. Dieses Portal wurde am 6. März 2012 freigeschaltet und bildete als Schnittstelle zwischen den menschlichen Anwendern und den Netzdiensten einen wichtigen Meilenstein für die GDI-DE (REIMER, 2012; ILLERT, 2012). Laut DE LANGE (2013) wird die Mehrfachnutzung von Geodaten durch eine mangelnde Qualität der Metadaten sowie sich unterscheidende Erfassungsmethoden und -ziele erschwert. Durch Mehrfacherhebungen in isolierten Datenbeständen wird die Datenhaltung ineffektiv und es kommt zu Redundanzen. Diese können durch den Aufbau nationaler und internationaler GDI vermieden werden. Außerdem können nach BERNARD, FITZKE und R. M. WAGNER (2004) durch eine GDI Effizienzsteigerungen erzielt werden, da der Aufwand des Datenzugriffs geringer ausfällt<sup>8</sup>. Ferner können die Kosten für die Datenproduktion und -bereitstellung gesenkt und durch den verbesserten Austausch eine effizientere Datennutzung erreicht werden. Essentiell ist dabei die Verwendung standardisierter und interoperabler Webdienste (ebd.).

Die Politik ist sich der gesellschaftlichen Bedeutung des Themas INSPIRE bewusst und veröffentlicht im Zuge des Reportings seit 2005 alle drei Jahre einen Geo-Fortschrittsbericht (BILL, 2014). Im vierten Geo-Fortschrittsbericht des Bundesministeriums des Innern heißt es, dass für das Jahr 2015 „mehr als 12.000 Daten und 16.000 Dienste als INSPIRE-betroffen gemeldet und innerhalb des INSPIRE-Zeitplans über das Internet zugänglich gemacht [worden]“ (BUNDESMINISTERIUM DES INNEREN, 2017, S. 7). Mittlerweile ist diese Zahl für das Beobachtungsjahr 2017 laut der Koordinierungsstelle der GDI-DE auf ca. 22.000 Geodatensätze und 21.000 Geodatendienste angewachsen. Hiervon entfallen jedoch lediglich 72 Datensätze und 340 Dienste auf das Land Berlin. Verglichen mit anderen Bundesländern liegt Berlin in Bezug auf das INSPIRE-relevante Angebot an Geodaten damit aber im Mittelfeld (siehe Tabelle 2.1).

---

<sup>8</sup>Durch Metadaten-Auskunftssysteme für Geodaten und Geodienste sowie Katalogdienste soll die Auffindbarkeit der gesuchten Datensätze verbessert werden. Zum Stand Februar 2019 sind im sogenannten Geodatenkatalog 35 Katalogdienste eingebunden, die ca. 135.000 Metadatensätze umfassen (BUNDESAMT FÜR KARTOGRAPHIE UND GEODÄSIE, 2017)

Tabelle 2.1: Anzahl INSPIRE-betroffene Geodatensätze und -dienste (Stand 2017) (KOORDINIERUNGSSTELLE GDI-DE 2018)

Bundesland	Geodatensätze	Geodatendienste	Summe
Sachsen-Anhalt	69	67	136
Bremen	65	83	148
Brandenburg	122	135	257
Hamburg	114	159	273
Bayern	211	82	293
Thüringen	244	51	295
Baden-Württemberg	324	67	391
Schleswig-Holstein	334	64	398
Nordrhein-Westfalen	218	184	402
Berlin	72	340	412
Hessen	183	368	551
Mecklenburg-Vorpommern	515	312	827
Sachsen	929	288	1217
Saarland	1759	3518	5277
Niedersachsen	9566	332	9898
Rheinland-Pfalz	7400	14916	22316

Im dritten Geo-Fortschrittsbericht wurde die Aufstellung einer Nationalen Geoinformations-Strategie (NGIS) als zentrale Vorgabe definiert. Als Orientierung für zukünftige Aktivitäten aller Akteure wurden in der NGIS drei wesentliche Grundsätze aufgeführt, die in Abbildung 2.8 dargestellt sind und ferner in 15 Ziele mit 42 Detailzielen aufgeteilt werden können. Hierunter fallen Punkte wie Nutzerorientierung, Wirtschaftlichkeit, Transparenz oder Nachhaltigkeit. Ein weiterer sehr wichtiger Punkt ist die leistungsfähige Informationstechnologie (IT), denn nur mit ihr kann eine produktive Nutzbarkeit gewährleistet werden (BUNDESMINISTERIUM DES INNERN, 2017). Speziell die Punkte Nutzerorientierung, Wirtschaftlichkeit und leistungsfähige IT werden mit dieser Arbeit adressiert. Fraglich ist, ob diese Kriterien bereits soweit erfüllt werden, dass eine wirtschaftliche Nutzung der Open Geo Data Dienste möglich ist oder ob die Leistungsfähigkeit der IT noch nicht ausreichend ist, um diese nutzerfreundlich einzubinden.

Im Zuge der INSPIRE Entwicklungen wurde 2009 das Geodatenzugangsgesetz (GeoZG) angepasst, wodurch die Geodaten des Bundes bereits weitgehend kostenfrei zur kommerziellen und nicht kommerziellen Nutzung zur Verfügung gestellt worden. Ferner wurde das Dienstleistungszentrum für Geodaten im Bundesamt für Kartographie und Geodäsie (BKG) eingerichtet. Mit dem OpenSearch GeoTemporal Service (OSGTS) konformen Geokodierungsdienst, welcher gemeinsam mit den Ländern aufgebaut wurde, ist ein weiterer Meilenstein erreicht worden. Weitere positive Entwicklungen lassen sich im einfachen Zugriff auf Fernerkundungsprodukte über das Zentrale Kriseninstitut (ZKI) oder die freie und offene Bereitstellung von Copernicus<sup>9</sup> Sentinel-Daten feststellen (ebd.).

<sup>9</sup> *Copernicus* ist ein Erdbeobachtungsprogramm, welches von der Europäischen Kommission und der Europäischen Weltraumorganisation (ESA) gegründet wurde und Dienste aus sechs Hauptthemenbereichen umfasst: Atmosphäre, Meeresumwelt, Land, Klima, Katastrophen- und Krisenmanagement sowie Sicherheit (COPERNICUS.EU, 2018).



Abbildung 2.8: Grundsätze der Nationalen Geoinformations-Strategie (eigene Darstellung nach BUNDESMINISTERIUM DES INNERN (2017))

Originär ist die INSPIRE-Richtlinie für den Austausch von Daten auf Ebene des öffentlichen Sektors entwickelt worden. Es ist kein freier Zugang für die Bürger im Sinne der Open Data vereinbart und Regelungen zu Kosten und Lizenzen wurden weitgehend ausgespart (ILLERT, 2012). Grundsätzlich sollen die Mitgliedstaaten dafür Sorge tragen, dass die Inhalte der Öffentlichkeit kostenfrei zur Verfügung gestellt werden. Sie können jedoch gemäß Artikel 14 (2) der INSPIRE-Richtlinie Gebühren für beispielsweise Darstellungs- oder Downloaddienste verlangen, wenn durch die Einnahmen die Wartung der Datensätze und Dienste gesichert wird. In diesem Fall müssen Dienstleistungen des elektronischen Geschäftsverkehrs gewährleistet sein (ebd.). Ferner können nach Artikel 13 (1) der Richtlinie bestimmte Geodatensätze in ihrer öffentlichen Zugänglichkeit eingeschränkt werden, sofern der Zugang nachteilige Auswirkungen auf die öffentliche Sicherheit, nationale Verteidigung oder internationale Beziehungen hätte.

Wenngleich INSPIRE die Grundlagen für staatliche Open Data Initiativen mithilfe der Verbesserung der Daten-Nutzbarkeit sowie der Förderung des (freien) Austauschs schafft, ist die ursprüngliche Zielsetzung der Richtlinie nicht rein Open Data getrieben. Trotz allem etabliert sie auf Basis der *International Organization for Standardization* (ISO) und des *Open Geospatial Consortiums* (OGC) (siehe Kapitel 2.6.1) Standards, sorgt für eine vereinfachte Verbreitung von Geodatensätzen und schärft das Bewusstsein der beteiligten Akteure (ebd.). Die Umsetzung von INSPIRE ist jedoch noch nicht abgeschlossen und steht noch vor einigen Herausforderungen. So müssen beispielsweise inkompatible Lizenzmodelle bereits existierender Portale und deren Datenbestände weiterhin miteinander in Einklang gebracht werden (KALASEK und WENINGER, 2015). Wachsende kostenlose Angebote an freien Daten im Geoinformationswesen liegen nach ILLERT (2012) weniger an INSPIRE selbst, als vielmehr an dem allgemeinen „Open“-Trend und Anbietern wie OpenStreetMap (siehe Kapitel 2.4.4) oder Google, welche innovative Geschäftsmodelle anbieten. Auch die öffentliche Verwaltung könne sich dieser Entwicklung nicht entziehen, sodass die Bundesverwaltung bereits Überlegungen äußerte, INSPIRE-konforme Datensätze zukünftig kostenlos und für sämtliche Benutzer anzubieten (ebd.).

### 2.4.3 PSI und IWG, IFG und UIG

Neben der europäischen Richtlinie INSPIRE und ihrer Umsetzung im GeoZG gibt es auf EU-, Bundes- und Länderebene weitere gesetzliche Rahmenbedingungen zu GDIs und der Nutzung, Weiterverwendung und Veröffentlichung von Geodaten, die im folgenden kompakt Erwähnung finden sollen. Grundsätzlich ist die Bereitstellung von Geodaten Ländersache – hiervon ausgenommen sind Landkreise und kreisfreie Städte (PADERTA, 2012). Die Kataster- und Vermessungsverwaltungen der Bundesländer haben neben der Erhebung der Geobasisdaten die Aufgabe, diese zu halten und zur weiteren Nutzung bereitzustellen. Außerdem sind sie je nach Vermessungsgesetz des jeweiligen Landes dazu verpflichtet, das Liegenschaftskataster sowie topographische Karten zu führen. Die Länder müssen auf Grund von verfassungsrechtlichen Vorgaben jeweils eigene Landesgesetze erlassen, wodurch eine Vielzahl von Vorschriften vorliegen, die beispielsweise die Nutzung von Geodaten des öffentlichen Sektors regeln. Diese sind sehr unterschiedlich und weisen eine jeweils andere Präzision auf (ebd.).

Im Jahr 2003 wurde auf europäischer Ebene die „Richtlinie über die Weiterverwendung von Informationen des öffentlichen Sektors“ (Richtlinie 2003/98/EG), kurz „Public Sector Information“-Richtlinie (PSI), verabschiedet. Seit 2008 haben alle Mitgliedsstaaten die Richtlinie umgesetzt, die eine verbesserte Zugänglichkeit von Informationen aus dem öffentlichen Sektor, die Vereinheitlichung von Grundprinzipien zur Weiterverwendung dieser Informationen und daraus folgend eine Stärkung der Wirtschaft zum Ziel hat. Durch diese Angleichungen innerhalb der EU soll auch die Wettbewerbsfähigkeit gesteigert werden, da beispielsweise US-Unternehmen auf eine Vielzahl öffentlicher Informationen in einem gut funktionierenden System zugreifen können (PADERTA, 2012; FALLENBÖCK, GALLA und STOCKINGER, 2005). Auf Bundesebene wird die PSI-Richtlinie durch das 2006 verabschiedete Informationsweiterverwendungsgesetz (IWG) umgesetzt. Zwar werden im ursprünglichen IWG Rahmenbedingungen zur Weiterverwendung geregelt, aber keine Aussagen zum Anspruch auf Zugang getroffen. Eine verpflichtende Bereitstellung von Daten wird für die öffentlichen Verwaltungen somit nicht begründet, obwohl Transparenz und fairer Wettbewerb in der PSI-Richtlinie die Säulen bilden (PADERTA, 2012). Durch diese Hindernisse kann nach FALLENBÖCK, GALLA und STOCKINGER (2005) das ökonomische Potenzial von Informationen des öffentlichen Sektors nicht genutzt werden. Im Jahr 2013 wurde die PSI-Richtlinie durch die Richtlinie 2013/37/EU aktualisiert. Durch diese Änderung wurde beispielsweise eine Verpflichtung eingeführt, durch die die Weiterverwendung der allgemein zugänglichen öffentlichen Daten zugelassen wird. Außerdem wurden die Grenzkosten für Gebühren (beispielsweise zur Deckung der Reproduktion und Bereitstellung) reduziert und diese Kosten müssen transparenter dargelegt werden (EUROPÄISCHE KOMMISSION, 2003). In allen 28 Mitgliedstaaten wurde die Änderungsrichtlinie umgesetzt – in Deutschland wurde eine Anpassung des IWG im Jahr 2015 vorgenommen.

Zum Zeitpunkt dieser Arbeit ist eine Neufassung der PSI-Richtlinie in Ausarbeitung, die im April 2018 von der EU-Kommission vorgeschlagen wurde. Voraussichtlich wird die Neufassung der Richtlinie – die im Englischen den Namen „Open Data and Public Sector Information Directive“ tragen wird – noch vor der Europawahl im Mai 2019 verabschiedet werden. Die Neufassung fördert die Verwendung von Open Data, was durch den Zusatz im Titel verdeutlicht wird. So wird konkret auf offene Formate und standardisierte Lizenzen eingegangen. Ferner

sollen die Daten grundsätzlich kostenfrei oder zu marginalen Kosten veröffentlicht werden, die lediglich die Aufwände zur Verbreitung decken. Hierfür wurden die möglichen Ausnahmen für die Erhebung von Gebühren minimiert, wodurch klein- und mittelständische Unternehmen eine geringere Markteintrittsbarriere erhalten. Des Weiteren sind nun mehr Sektoren von der Richtlinie betroffen, wodurch beispielsweise auch Forschungsergebnisse besser verfügbar gemacht werden sollen. Ein weiteres Ziel ist die Bereitstellung von dynamischen Daten durch Schnittstellen. Es ist also ein Ziel mit angemessenen technischen Mitteln nicht nur die Bereitstellung von Daten, sondern eine Entwicklung hin zu verbundenen Diensten zu fördern (EUROPÄISCHE KOMMISSION, 2019).

Ferner seien das Informationsfreiheitsgesetz (IFG) sowie das Umweltinformationsgesetz (UIG) genannt. Das IFG gewährleistet für jede Person – auch ohne Begründung des Interesses – den Rechtsanspruch auf Zugang zu amtlichen Informationen (siehe § 1 (1) IFG). Im Jahr 2005 erfolgte das Inkrafttreten auf Bundesebene, während die Umsetzung auf Landesebene unter Berücksichtigung des zeitlichen Aspekts sehr unterschiedlich ausfällt. Bayern, Niedersachsen und Sachsen haben noch immer kein Landes-IFG erlassen, während diese in Brandenburg bereits 1998 und Berlin 1999 in Kraft getreten sind. Das UIG setzte im Jahr 2004 auf Bundesebene die EU-Richtlinie 2003/4/EG über den „Zugang der Öffentlichkeit zu Umweltinformationen“ um. Die jeweiligen Gesetze auf Landesebene erfolgten zwischen 2005 und 2007. Durch die Richtlinie sollen speziell Umweltinformationen für Bürger frei zugänglich gemacht werden, um Beteiligungen in der Umweltpolitik zu stärken und die Informationspflicht zu intensivieren (PADERTA, 2012).

#### **2.4.4 OpenStreetMap**

Ein beeindruckendes Beispiel eines „Open Geo Data“-Projektes ist OpenStreetMap (OSM). Das Community-Projekt wurde 2004 in Großbritannien gestartet und befasst sich mit der Erfassung einer weltweiten Straßenkarte. Hierbei können von Jedem Informationen in einer großen Datenbank hinzugefügt werden, die anschließend in Karten umgewandelt und zur Verfügung gestellt werden. Üblicherweise geschieht die Aufnahme der Daten durch das Abzeichnen von Satellitenbildern oder mithilfe von GPS-Geräten, deren GPS-Tracks auf die OSM-Server hochgeladen werden. Außerdem werden Datenimporte von offizieller Seite durchgeführt, die jedoch speziell bezüglich rechtlicher Aspekte abgesprochen werden müssen. Ein abgeschlossenes Beispiel hierfür wären Küstenlinien der US-Behörde National Geospatial-Intelligence Agency (RAMM und TOPF, 2010). Neben Straßen, Wegen, Flächen und deren Eigenschaften werden auch Points Of Interest (POIs) aufgenommen, z. B. Restaurants, Briefkästen oder öffentliche Einrichtungen. Abgebildete POIs spielen bei der geographischen Einordnung eine entscheidende Rolle, da Entfernungen zu Schulen, Supermärkten oder Krankenhäusern bei der Betrachtung auf einer Karte schnell erfassbar sind. Im Zuge dieser Arbeit werden die Entfernungen zu ausgewählten POIs auf Grundlage von OSM betrachtet (siehe Kapitel 3.2). Im Untersuchungsgebiet Berlin eignet sich die Verwendung von OSM, während es in dünner besiedelten Gebieten derzeit noch einige Erfassungslücken gibt (BURGDORF, KRISCHAUSKY und MÜLLER-KLEISSLER, 2015).

Mittlerweile hat OSM bei 4,3 Mio. angemeldeten Nutzern die 1 Million-Marke in Bezug auf mitwirkende Nutzer überschritten (siehe Abbildung 2.9) und stellt eine echte Alternative zu

proprietären Anbietern wie Google Maps oder Bing Maps dar. Die Datenbank enthält dabei etwa 4,1 Milliarden Knotenpunkte und erfährt 3 Millionen Änderungen pro Tag (Stand 08.11.2017) – eine immense Datenmenge (OPENSTREETMAP, 2018b). Die Liste an Projekten auf Grundlage von OSM ist riesig, von speziellen Karten für Radfahrer, Wanderer oder Gehbehinderte über Kunstprojekte bis hin zu 3D-Darstellungen und Routenplanern. Eine bei weitem nicht vollständige Liste kann unter OPENSTREETMAP (2018a) eingesehen werden.

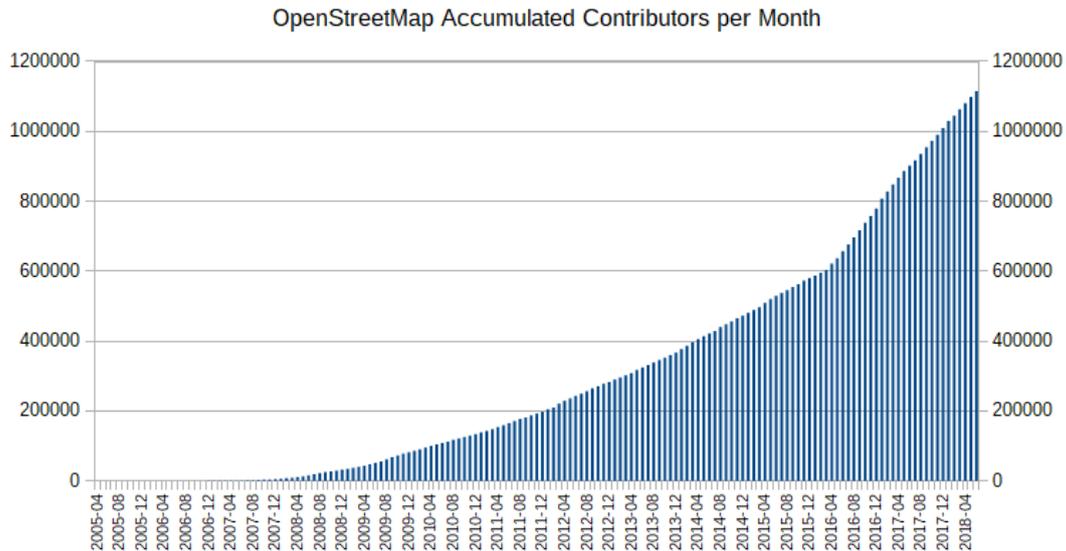


Abbildung 2.9: Entwicklung der OpenStreetMap Mitwirkenden (Quelle: OPENSTREETMAP (2018b))

Auf Grund von institutionellen Initiativen wie INSPIRE sowie Engagements auf Basis privater Beteiligungen wie OSM, werden die Entwicklungen zum Open Data Gedanken gefördert. Rein quantitativ ist somit sowohl aus öffentlicher als auch privater Hand mit einem Zuwachs an freien Geodaten zu rechnen. Jedoch spielt auch die Qualität der Daten sowie die Interoperabilität eine wichtige Rolle für die Anwender, sodass diese Aspekte bei dem Wettkampf der Institutionen um einen guten Ranking-Platz nicht auf der Strecke bleiben sollten. Nur so können vermarktbarere Dienste und korrekte Forschungsprojekte realisiert werden (KALASEK und WENINGER, 2015).

## 2.5 Nutzungsregelungen und Lizenzen

Für die Nutzung von offenen Geodaten gibt es verschiedene Regelungen/Lizenzen, die ihren Zweck und Nutzerkreis sowie deren Rechte bestimmen. Neben vielen individuellen Nutzungsbedingungen haben sich einige Lizenzen durchgesetzt, die zu großen Anteilen Verwendung finden. Mithilfe dieser Bestimmungen können die Bereitsteller sicherstellen, wie die Daten und Dienste verwendet werden, denn „eine offene Bereitstellung als ‚OpenData‘ bedeutet nicht, dass die Nutzung der Daten ohne Regelungen zur Verfügung gestellt werden“ (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017, S. 4). INSPIRE bewirbt die technische Interoperabilität von geographischen Daten, die neue Produkte und Dienste durch einfach kombinierbare Datensätze verschiedener Länder schaffen soll. Doch ohne standardisierte Lizenzen und deren Verwendung birgt dies schwer überwindbare Hindernisse, die zu Verzögerungen oder dem kompletten Verwerfen von Plänen sorgen können (K. JANSSEN und CROMPVOETS, 2013). Das hiesige Kapitel soll einen Überblick über die in Deutschland vorwiegend verwendeten Nutzungsregelungen (speziell für die GDI-DE) geben. Detailspekte, wie der Zeitpunkt des Inkrafttretens oder Entscheidungshilfen für die Wahl einer Lizenz, werden nicht beleuchtet.

Grundsätzlich muss zwischen zwei Formen der Nutzungsregelungen unterschieden werden: den öffentlich-rechtlichen und den privatrechtlichen. Die Rechtsgrundlage bildet in Deutschland das Urheberrechtsgesetz (UrhG), welches durch persönliche geistige Schöpfung entstandene Werke der Literatur, Wissenschaft oder Kunst schützt. Hierunter fallen nach § 2 UrhG beispielsweise Karten, Skizzen, Tabellen oder Darstellungen wissenschaftlicher oder technischer Art. Weitere verwandte Schutzrechte sind ebenfalls von großer Relevanz, da diese z. B. Luftbilder als Lichtbild schützen (§ 72 UrhG) oder Datenbestände aus digitalen Landschaftsmodellen als Datenbanken (§ 87a UrhG) (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017). Während Kartendarstellungen die Voraussetzungen für den Urheberrechtsschutz erfüllen, tun dies reine Koordinatensammlungen oder Messdaten nicht, da diese ohne geistig schöpferische Leistung nur die Wirklichkeit wiedergeben (ebd.). Nach § 87a UrhG können diese jedoch als Datenbank geschützt sein. Folgende Geodaten sind nach BEHRENS, IBENDORF und WÜRRIEHAUSEN (ebd.) grundsätzlich vom Datenbankrecht oder Urheberrecht erfasst:

- «Topographische Karten (Datenbankrecht)
- Liegenschaftskarte, Liegenschaftsbuch, Vermessungszahlenwerk (Datenbankrecht)
- Geotopographische Landesaufnahme, Topographische Landeskartenwerke (Urheberrecht, Datenbankrecht)
- Landesluftbildsammlung (Urheberrecht)
- Quasigeoid (Datenbankrecht)
- Kaufpreissammlung, Bodenrichtwerte, Übersichten über die Bodenrichtwerte typischer Orte, sonstige zur Wertermittlung erforderliche Daten (Datenbankrecht)
- Grundstücksmarktbericht (Urheberrecht, Datenbankrecht)»

Eine weitere öffentlich-rechtliche Regelungsform stellt dank der Open Government und Open Data Entwicklungen der letzten Jahre die Verordnung zur Festlegung der Nutzungsbestimmungen

für die Bereitstellung von Geodaten des Bundes (GeoNutzV) dar, die im Jahr 2012 mit der Änderung des Geodatenzugangsgesetzes (GeoZG) geschaffen wurde (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017). Nach der bundesweit verbindlich geltenden GeoNutzV müssen laut § 2 Geodaten und Geodatendienste des Bundes sowie deren Metadaten sowohl für nicht-kommerzielle als auch kommerzielle Zwecke kostenfrei zur Verfügung gestellt werden. Sofern die Nutzung nach GeoNutzV nicht weiter eingeschränkt wird, können privatrechtliche Lizenzen abgeschlossen und Entgelte verlangt werden. In Hinblick auf das konkrete Untersuchungsgebiet dieser Thesis sei angemerkt, dass zum 01.10.2013 eine Adaption der GeoNutzV im Land Berlin eingeführt wurde. Die sogenannte GeoNutz-BE (auch GeoNutzV-Berlin) gilt für amtlichen Geodaten der Berliner Vermessungsverwaltung (ebd.). Speziell die folgenden Punkte aus § 1 GeoNutz-BE sind von Relevanz:

- «(1) Geodaten und Geodatendienste, einschließlich zugehöriger Metadaten, werden für alle derzeit bekannten sowie für alle zukünftig bekannten Zwecke kommerzieller und nicht kommerzieller Nutzung geldleistungsfrei zur Verfügung gestellt. [...]
- (3) Die bereitgestellten Geodaten und Metadaten dürfen insbesondere [...] mit eigenen Daten und Daten Anderer zusammengeführt und zu selbständigen neuen Datensätzen verbunden werden; [...] in interne und externe Geschäftsprozesse, Produkte und Anwendungen in öffentlichen und nicht öffentlichen elektronischen Netzwerken eingebunden werden.»

Voraussetzung hierfür ist ein Quellenvermerk und gegebenenfalls ein Veränderungshinweis (SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN, 2013). Gleiches gilt laut § 1 (4) GeoNutz-BE auch für angebotene Geodatendienste.

Folgend werden privatrechtliche Regelungsformen vorgestellt, die zu Teilen auch internationale Bedeutung haben.

## Creative Commons

Als eine der bekanntesten seien die *Creative Commons* (CC) Lizenzen genannt. Die gleichnamige Non-Profit-Organisation hilft Urhebern dabei, kreative und rechtlich geschützte Inhalte durch vorgefertigte Lizenzverträge freizugeben. Dabei wird mithilfe der Lizenz klargestellt, was mit den Inhalten geschehen darf und welche Restriktionen es gibt. Durch die CC-Lizenzverträge sollen den Urhebern mehr Optionen zur Freigabe bereitgestellt werden. Durch den immensen Anstieg digitaler Medien möchte die Organisation speziell für Kreative die rechtlichen Unsicherheiten reduzieren, um beispielsweise auf Werken anderer aufzubauen, ohne selbst große Expertise im Urheberrecht zu besitzen (CREATIVE COMMONS DEUTSCHLAND, 2018b).

In Deutschland hat sich in Bezug auf Creative Commons speziell die *CC BY 3.0 DE* (Namensnennung 3.0 Deutschland) durchgesetzt. Hierbei darf das Material in jedem Format vervielfältigt und weiterverarbeitet werden. Außerdem darf es entgeltfrei verändert, mit anderen Daten gemischt oder weiter ausgebaut werden – auch für kommerzielle Zwecke. Bedingung hierfür ist zum einen, dass keine zusätzlichen Verfahren oder Klauseln eingesetzt werden dürfen, die laut

Tabelle 2.2: Creative Commons Lizenzen nach CREATIVE COMMONS DEUTSCHLAND (2018b)

Symbole	Lizenzinhalte
	Namensnennung
	Namensnennung + Keine Bearbeitung
	Namensnennung + Nicht kommerziell
	Namensnennung + Weitergabe unter gleichen Bedingungen
	Namensnennung + Nicht kommerziell + Keine Bearbeitung
	Namensnennung + Nicht kommerziell + Weitergabe unter gleichen Bedingungen

Lizenz zulässige Handlungen untersagen. Zum anderen müssen angemessene Urheber- und Rechteangaben gemacht sowie ein Link zur Lizenz hinzugefügt werden. Sofern Änderungen an dem Datensatz vorgenommen wurden, muss dies gekennzeichnet werden.

Neben dem *BY* Lizenztyp (zur Kennzeichnung der Namensnennung) gibt es drei weitere Grundelemente, die jeweils durch ein Symbol dargestellt sind. Durch deren Kombination ergeben sich somit sechs CC Lizenzen, die in der Version 3.0 angepasst an deutsches Recht zur Verfügung stehen. Tabelle 2.2 listet die Symbole zusammen mit den Lizenzinhalten auf. Während eine Namensnennung immer gefordert ist, wird „eine Weitergabe unter gleichen Bedingungen [...] nur für Bearbeitungen als sinnvoll erachtet“ (STAATSBETRIEB SÄCHSISCHE INFORMATIK DIENSTE, 2018). Ferner gibt es die *CC0 1.0 Universell (CC0 1.0) Public Domain Dedication* (auch *CC Zero* genannt), bei der der Bereitsteller das Werk in die Gemeinfreiheit (Public Domain) entlassen hat und somit weltweit, soweit gesetzlich möglich, auf alle Urheberrechte und verwandte Schutzrechte verzichtet (CREATIVE COMMONS DEUTSCHLAND, 2018a).

Die Rechteinhaber geben bei der Verwendung einer der genannten Lizenzen klar definierte Freigaben für ihre Werke und ein Statement für Open Access für Kulturgüter, ohne Verlust des Schutzes durch das Urheberrecht. Da die gezielte Nachfrage an frei lizenzierten Inhalten steigt, können Werke oftmals schneller verbreitet werden. Datenerzeuger und Künstler tragen somit zur Vermehrung eines frei nutzbaren Materialpools bei, der zum Teil Einschränkungen in der Freigabe aufzeigen kann, aber aufgrund des geringeren Aufwands überhaupt zu einer Nutzung und Weiterverwendung der Inhalte führt. Daten, die schnell und unkomplizierte ohne Kontaktaufnahme zum Urheber verwendet werden können, werden wahrscheinlicher genutzt als solche, bei denen diese bei jeder Verwendung nötig wird oder komplizierte Lizenzen studiert werden müssen. Einige Anbieter sind davon beeindruckt, wie ihre Werke auf Grund einer freien Lizenz in den Weiten des Internets teils international weiterverwendet werden (CREATIVE COMMONS DEUTSCHLAND, 2018b).

Aufgrund des Ursprungs der CC-Lizenz (künstlerische urheberrechtliche Werke wie Musik, Bilder, Filme etc.) werden Formulierungen verwendet, die für die Nutzung von Geodaten und

Geodatendiensten nicht einfach nachvollziehbar sind. Bis einschließlich der Version 3.0 wird nicht klar, ob beispielsweise das einzelne Objekt einer Karte oder die gesamte Karte als Produkt geschützt wird, da vorrangig das sogenannte „Werk“ geschützt wird. Auch eine Modifizierung ist dementsprechend schwer zu deklarieren (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017; K. JANSSEN und CROMPVOETS, 2013). Demgegenüber wurden nach BEHRENS, IBENDORF und WÜRRIEHAUSEN (2017) in der im Jahr 2013 eingeführten Version 4.0 – die bisher nicht in einer an das deutsche Urheberrechtsgesetz angepassten Fassung vorliegt – die folgenden Neuerungen hinzugefügt:

- «die ausdrückliche Nennung und Lizenzierung von Datenbankrechten und verwandten Schutzrechten,
- die Klarstellung, dass sogenanntes Data-Mining keine Bearbeitung darstellt,
- dass im Rahmen von Share-Alike-Lizenzen (Weitergabe unter gleichen Bedingungen) lediglich die zeitlich zuletzt vergebene Lizenz maßgeblich ist,
- dass nunmehr die Möglichkeit besteht, Lizenzverstöße durch Abstellen des Verstoßes innerhalb einer 30-Tage-Frist zu heilen,
- dass die Möglichkeit erwähnt wird, bezüglich der Rechtangaben (z. B. Namensnennung) nunmehr immer via Link auf eine andere Seite zu verweisen.»

### Open Data Commons

Ein Projekt der Stiftung Open Knowledge International (OKFN) ist *Open Data Commons*<sup>10</sup>, bei dem Lizenzen für freie Datenbanken gepflegt und somit rechtliche Lösungen für Open Data bereitgestellt werden. So erlaubt beispielsweise die *Open Database License* (ODC-ODbL) das Teilen und Bearbeiten von Datenbanken unter der Voraussetzung der Namensnennung und Bereitstellung des neuen Werkes unter der gleichen Lizenz (wie CC BY-SA). Das in Kapitel 2.4.4 vorgestellte OSM-Projekt ist das wohl bekannteste Beispiel, welches die ODbL für seine Datenbank nutzt. Seit dem 12.09.2012 sind die Daten nicht mehr nach der CC BY-SA, sondern nach der ODbL lizenziert (OPENSTREETMAP, 2012). Die Kartografie der Kartenkacheln sowie die Dokumentation ist weiterhin unter der CC BY-SA verfügbar (OPENSTREETMAP, 2018c).

Die *Attribution License* (ODC-BY) ist genauso wie die ODC-ODbL speziell für Datenbanken, genauer Datenbankrechte und -strukturen, entwickelt worden. Die Inhalte der Datenbank werden von der Lizenz nicht erfasst, anders als bei der CC BY (CREATIVE COMMONS, 2018). Neben Hinweisen auf die Originaldatenbank müssen gegenüber Dritten die Nutzungsbedingungen für jede öffentliche Nutzung und Weiterverarbeitung erklärt werden – dies gilt auch für abgeleitete Werke (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017).

Die *Public Domain Dedication and Licence* (PDDL) sieht im Gegensatz zur ODC-BY und ODC-ODbL in Hinblick auf die vorgesehene Nutzung keinerlei Beschränkungen vor und stellt nach BEHRENS, IBENDORF und WÜRRIEHAUSEN (ebd.) eher ein Verzicht aller Rechte des Rechteinhabers dar als eine Lizenz. Hierdurch wird „eine uneingeschränkte öffentliche Nutzung

<sup>10</sup> *commons* engl. für Gemeingut/öffentliches Gut

ermöglicht“ (ebd., S. 13). Es erfolgt nur dann eine Lizenzierung, wenn aufgrund der Rechtsordnung eine gänzliche Aufgabe der Rechte unzulässig ist (ebd.).

### **AdV-Musterlizenzvereinbarung**

Basierend auf der Musterlizenzvereinbarung einer Kooperation der Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder und der Bundesrepublik Deutschland (AdV), der Kommission für Geoinformationswirtschaft (GIW) sowie des Deutschen Industrie und Handelskammertages (DIHK) aus dem Jahre 2008, wurde die *AdV Musterlizenzvereinbarung* veröffentlicht. Sie gilt für Geobasis- und Geofachdaten des amtlichen Vermessungs- und Geoinformationswesens und ist konkret für das Vermessungswesen erweitert worden. Für den Online-Abschluss gibt es drei verschiedene Varianten zur Nutzung:

1. freie Nutzung
2. private Nutzung und Nutzung für Bildung & Forschung
3. geschäftliche Nutzung

### **Datenlizenz Deutschland**

Speziell für die offene Datenbereitstellung auf dem Portal des Bundes und der Länder *GOVDATA* wurde die *Datenlizenz Deutschland – Namensnennung* entwickelt. In der Standardvariante dürfen die Daten unter Nennung der Datenquelle sowohl nicht-kommerziell als auch kommerziell entgeltfrei genutzt werden. Ferner gibt es die Variante *Datenlizenz Deutschland – Zero*, die eine Weiterverwendung frei von Einschränkungen ermöglicht. Diese in der Zusammenarbeit von Bund, Ländern und kommunalen Spitzenverbänden entstandene Lizenz ist in der Regel nicht für Geo-Daten vorgesehen und hat nur eine geringe Bekanntheit und Akzeptanz in der Öffentlichkeit. Sie ist speziell für die Verwaltung geeignet, da sie einen starken Bezug zum deutschen Rechtsraum besitzt und somit als einfache und sichere Variante gilt (SENATSVERWALTUNG FÜR WIRTSCHAFT, 2015). Da die Nutzung von Datendiensten, wie z. B. WFS-Diensten, nicht explizit genannt wird, ist die Lizenzierung für diesen Fall uneindeutig. Ferner werden von einigen Datenanbietern individuelle Haftungsregelungen eingeführt, da die Datenlizenz Deutschland – entgegen der Prinzipien der Vereinheitlichung – diesbezüglich keine Regelungen nennt (DIETRICH, 2013). In der aktuellen Version 2.0 ist die Datenlizenz Deutschland „offiziell durch den Sachverständigenrat der *Open Definition* als offene Lizenz anerkannt“ (SENATSKANZLEI, GESCHÄFTS- UND KOORDINIERUNGSSTELLE GOVDATA, 2016), nachdem es von der CC sowie der OKFN und Wikimedia starke Kritik bezüglich der Version 1.0 gab (DIETRICH, 2013). Grund hierfür waren unklare Formulierungen, die Inkompatibilität gegenüber „Open Licence“-konformer Lizenzen sowie unpraktikable Einschränkungen, wie eine „vorgesehene Verpflichtung zur nachträglichen Entfernung der Attribution“ (ebd.).

### **Geolizenz**

Die *Geolizenz* (aktuell in Version 1.4) kann deutschlandweit für sämtliche Verwaltungsebenen und Gewerbe angewandt werden. Dabei stand bis Ende 2016 ein Baukasten zur Verfügung, der

die standardisierten Lizenzen individuell konfigurierbar machte, um verschiedene Nutzungsrechte oder Nutzergruppen zu definieren. Speziell für Open Data Angebote gibt es eine Lizenzvariante, die räumlich und zeitlich uneingeschränkt zur Verfügung steht und mit Beginn der Datennutzung in Kraft tritt (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017).

### **INSPIRE Basic/Specific Licence**

Speziell für die Veröffentlichung von Daten & Diensten der EU-Mitgliedstaaten unter INSPIRE seien die *INSPIRE Basic/Specific Licences* genannt. Sie unterscheiden sich darin, ob bestimmte Beschränkungen geregelt werden sollen (Specific) oder es sich um eine einfache Abgabe handelt (Basic). Hierdurch soll den Institutionen und Einrichtungen der europäischen Gemeinschaft im Rahmen von INSPIRE eine Lizenzmöglichkeit geboten werden (ebd.).

Obgleich der Existenz mehrerer Standardlizenzen bieten sich für Datenbereitsteller einige Möglichkeiten der Lizenzierung. Deren Kenntnis ist auch für die Nutzer zur korrekten Verwendung der jeweiligen Daten notwendig. Nach BEHRENS, IBENDORF und WÜRRIEHAUSEN (ebd.) werden für Veröffentlichungen von geodatenhaltenden Stellen das Zurückgreifen auf existierende und verbreitete Lizenzmodelle sowie der Verzicht auf individuelle Anpassungen und Insellösungen empfohlen. Nur so kann in Bezug auf Lizenzen das Ziel von fachübergreifenden und internationalen Anwendungen von Geodaten gefördert werden (BEHRENS, IBENDORF und WÜRRIEHAUSEN, 2017; K. JANSSEN und CROMPVOETS, 2013). Bei der Verwendung mehrerer Datenquellen muss für jede einzelne geprüft werden, ob und in welchem Umfang und unter welchen Restriktionen diese eingesetzt werden kann.

## **2.6 Standardisierung von Geoinformationen**

Die Verwendung von technischen Standards ist ein wichtiger Aspekt in Hinblick auf Interoperabilität und die Nutzbarkeit von Daten und Diensten. Ohne Standards entwickeln sich häufig Insellösungen, die schwer miteinander verknüpft werden können. Aus diesem Grund bilden sich verstärkt Zusammenschlüsse einzelner Akteure, die gemeinsam Standards definieren und umsetzen und somit verbesserte Möglichkeiten für den Datenaustausch und die -verarbeitung bereitstellen.

### **2.6.1 Open Geospatial Consortium**

Das 1994 gegründete Open Geospatial Consortium (OGC) ist eine Vereinigung mehrerer rechtlich und wirtschaftlich selbstständiger Unternehmen, Behörden und Forschungsinstitutionen aus dem Geschäftszweck der Geoinformationen. Ziel der über 500 Mitglieder aus Wissenschaft, Wirtschaft und Regierung ist es, Technologien der Geodatenverarbeitung über das Internet zu definieren und zu standardisieren (VOGT, 2011). Ein Hauptaugenmerk des Konsortiums liegt auf dem bereits beschriebenen Begriff *offen* (siehe Kapitel 2.4), wodurch die Interoperabilität sowie die

Portabilität und die Wartbarkeit von Programmen besonders gefördert werden soll (GROOT und McLAUGHLIN, 2000). Seit vielen Jahren finden die Standards des OGC Verwendung in unterschiedlichsten Projekten. Vorschläge der Vereinigung werden oft schnell übernommen, da sich das OGC durch Mitglieder großer Bekanntheit, wie NASA, Google oder Esri, zu einer renommierten Organisation entwickelt hat (KRALIDIS, 2008).

Die folgenden Standards sind Komponenten der OGC Web Services Architektur (OWS) und bezeichnen Webdienste. Diese service-orientierte Architektur ist nicht nur im Bereich der Geoinformatik weit verbreitet. Das in Abbildung 2.10 dargestellte Schema verdeutlicht den Aufbau einer solchen Architektur. Der *Service* oder *Service Provider* bietet externen Benutzern Operationen zur Verwendung an. Durch Anfragen an den Service können die Konsumenten oder *Service Consumer* auf diesen Dienst zugreifen und die jeweilige Antwort erhalten. Diese Kommunikation zwischen Nutzer und Dienst ist in Kapitel 4.1 noch einmal ausführlicher beschrieben. Um den Dienst auffindbar zu machen, gibt es sogenannte *Service Directories*, in denen sich die Anbieter eintragen können. In diesen Verzeichnissen sind alle notwendigen Informationen des Dienstes enthalten, um den Nutzern den Zugriff zu ermöglichen. Ein solches Verzeichnis kann mithilfe eines *Catalog Service for Web* (CSW) verwirklicht werden. Hierbei handelt es sich um ein OGC Standard für das Veröffentlichen von Geodaten über das Internet (PERCIVAL, 2017).

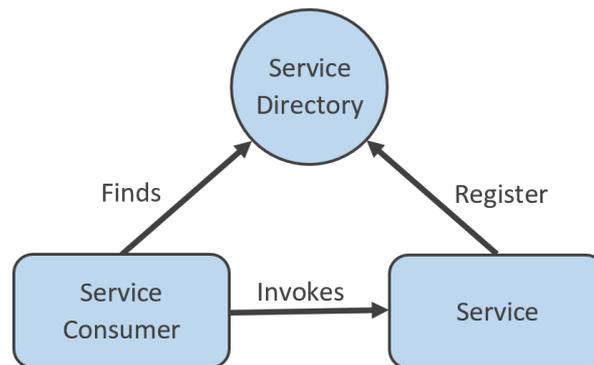


Abbildung 2.10: Service orientierte Architektur (SOA)

## 2.6.2 Web Map Service (WMS)

Ein *Web Map Service* (WMS) ist ein HTTP-Webdienst „zur bildhaften Darstellung von raumbezogenen Daten“ (VOGT, 2011, S. 205). Basierend auf den übermittelten Informationen, wie der geographischen Lage in Form einer Bounding Box<sup>11</sup>, definierten Stilvorgaben oder dem gewünschten Dateiformat, werden Bildkacheln (auch *tiles* genannt) über das Internet bereitgestellt. WMS Dienste sind grundsätzlich „read-only“, wodurch die Kacheln nur abgerufen, aber nicht modifiziert werden können. Ein *Basic WMS* verfügt nach der aktuellen Spezifikation in der Version 1.3 nach dem OPEN GEOSPATIAL CONSORTIUM (2006) immer über zwei Operation: **GetCapabilities** und **GetMap**. Während *GetMap* die eigentlichen Bilddaten in dem gewünschten Format übermittelt, liefert die Anfrage *GetCapabilities* die beschreibenden Metainformationen zu dem jeweiligen

<sup>11</sup>Eine Bounding Box ist ein Rechteck, welches die Ausdehnung eines Gebiets beschreibt und mithilfe zweier Koordinatenpaare definiert wird. Hierbei wird üblicherweise die linke obere und die rechte untere Position angegeben. Eine *minimum bounding box* ist das kleinste umschließende Rechteck einer Geometrie oder Geometriesammlung (WORBOYS, 1995).

Dienst. Diese beinhalten beispielsweise eine Beschreibung, Stichwörter, aber auch verfügbare Darstellungsebenen, Ausgabeformate, Kartenstile oder URLs, die für die einzelnen Operationen sowie optional für Legenden zur Verfügung stehen. Außerdem kann der *GetCapabilities*-Aufruf Informationen zu der Kontaktorganisation und den lizenzrechtlichen Rahmenbedingungen, wie einer erforderlichen namentlichen Copyright-Nennung oder die Einschränkung auf eine rein private Nutzung, nennen (OPEN GEOSPATIAL CONSORTIUM, 2006). Abbildung 2.11 zeigt beispielhaft die Antwort eines WMS *GetMap*-Requests in Form eines Bildes – dargestellt sind die Wohnlagen des Berliner Mietspiegels 2017.

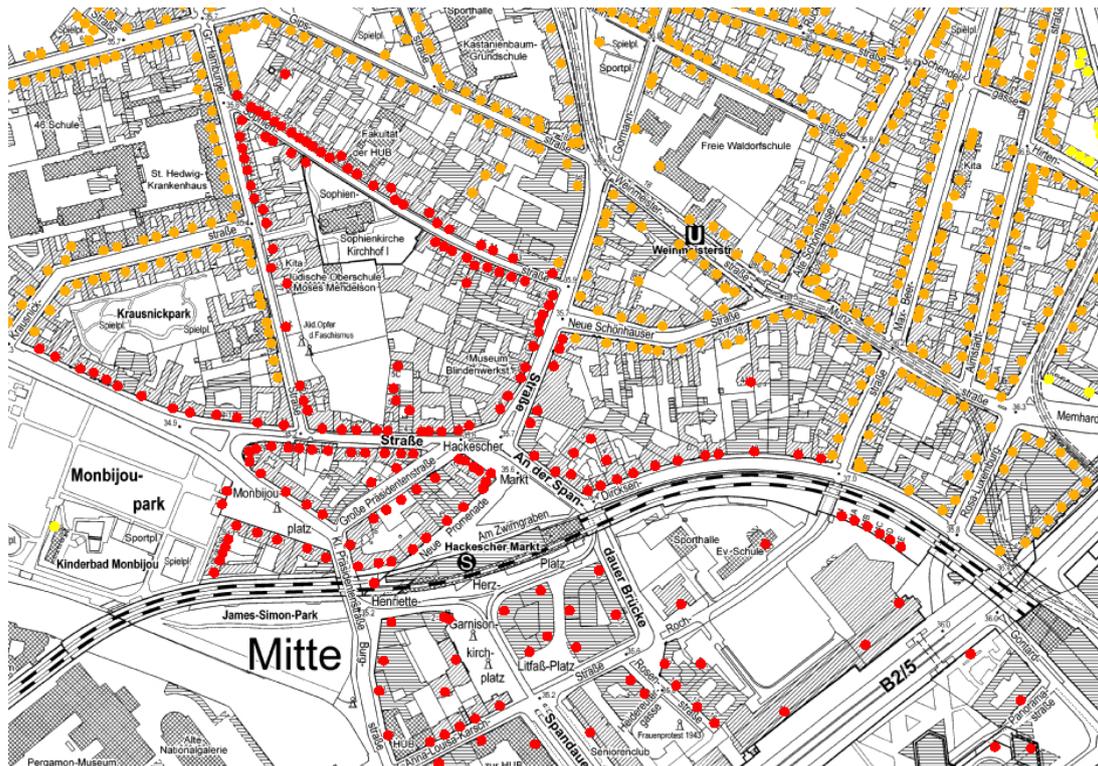


Abbildung 2.11: Beispiel eines WMS Dienstes - Wohnlagenkarte Berliner Mietspiegel 2017

Zusätzlich zu den zwei bereits erwähnten Operationen steht optional die **GetFeatureInfo**-Funktionalität zur Verfügung. Als Ergänzung zu den abrufbaren Bilddaten können so für einen angegebenen Punkt Sachdaten in einem Textformat abgerufen werden. Ein Dienst, der diese Methode unterstützt, wird *Queryable WMS* genannt (ebd.). Zur automatisierten vergleichenden Analyse von Immobilien spielt diese Operation eine wichtige Rolle, da die grafischen Bildformate wie JPEG oder PNG nicht maschinenlesbar (siehe Kapitel 2.6.5) und schwer automatisierbar zu analysieren sind. Für den weiteren Verlauf dieser Arbeit werden solche Services, die kein *Queryable WMS* darstellen, für die automatisierte Verarbeitung ausgeschlossen und nur zu Darstellungszwecken verwendet.

### 2.6.3 Web Feature Service (WFS)

Ein *Web Feature Service (WFS)* Dienst liefert Geoinformationen nicht in Form von Bilddaten, sondern als sogenannte *Features* im Vektorformat. Somit stehen dem Nutzer die Geometrien in Form von Punkten, Linien oder Polygonen zur Verfügung, aber auch die damit verknüpften

Sachdaten. Meist handelt es sich bei der Ausgabe um das Geography Markup Language (GML)-Format (siehe Kapitel 2.6.5).

Als *Basic WFS* wird nach OPEN GEOSPATIAL CONSORTIUM (2010) ein Read-Only-Dienst verstanden, der die folgenden Operationen anbieten muss, um die Konformitätsklasse zu erfüllen:

1. GetCapabilities
2. DescribeFeatureType
3. GetFeature
4. ListStoredQueries
5. DescribeStoredQueries

Dabei ist die *GetCapabilities*-Operation nahezu identisch zu verstehen, wie die des WMS Dienstes – sie gibt die Metadaten des Dienstes aus und beschreibt dessen Funktionalitäten. Aufgrund der unterschiedlichen Zielsetzung der Services werden im Vergleich zum WMS Dienst jedoch beispielsweise eine Liste der *FeatureTypes* sowie die Filterfähigkeiten ausgegeben (ebd.). Die Methode *DescribeFeatureType* stellt die einzelnen Featuretypen bezüglich ihrer Eigenschaften dar. Sie gibt an welche Felder das Feature beschreiben und welche Eigenschaften diese besitzen. Somit kann beispielsweise auch die Art der Geometrie überprüft werden. Das folgende Codebeispiel 2.1 zeigt die gekürzte Beschreibung eines Featuretypen.

Code 2.1: Beispiel DescribeFeatureType Operation

```

1  [...]
2  <xs:complexType name="user_Position">
3    <xs:complexContent>
4      <xs:extension base="~gml:AbstractFeatureType">
5        <xs:sequence>
6          <xs:element name="gid" minOccurs="0" nillable="true" type="xs:
              int"/>
7          <xs:element name="firstname" minOccurs="0" nillable="true" type=
              "xs:string"/>
8          <xs:element name="lastname" minOccurs="0" nillable="true" type="
              xs:string"/>
9          <xs:element name="organisation" minOccurs="0" nillable="true"
              type="xs:string"/>
10         <xs:element name="the_geom" minOccurs="0" nillable="true" type="
              ~gml:PointPropertyType"/>
11        </xs:sequence>
12      </xs:extension>
13    </xs:complexContent>
14  </xs:complexType>
15  <xs:element name="example" type="example:user_Position"
              substitutionGroup="gml:_Feature"/>
16  [...]
```

Wichtig zu erwähnen ist hierbei, dass ab der WFS-Version 1.1.0 über den Tag `<documentation>` für jedes Element eine Beschreibung hinzugefügt werden kann. Davor war es teils mit einem großen Aufwand verbunden, die oftmals abstrakten Elementnamen, wie „INDI16“ oder „STATUS1“, der gesuchten Kennzahl zuzuordnen.

Über einen *GetFeature*-Request können die Features mit ihren Eigenschaften abgerufen werden. Die Filtermöglichkeiten, die bei der Anfrage zusätzlich verwendet werden können, reichen je nach Unterstützung des Services von geometrischen Filtern, wie *Contains*, *DWithin* oder *Touches*<sup>12</sup>, bis hin zu Wertüberprüfungen einzelner Attribute. Häufig werden die ersten drei Operationen für einen Basic WFS als notwendig angegeben. Laut der WFS-Spezifikation in Version 2.0 nach OPEN GEOSPATIAL CONSORTIUM (2010) werden jedoch auch die Methoden zu *Stored Queries* in die Konformitätsklasse aufgenommen. Mithilfe von sogenannten *Stored Queries* kann der WFS-Server Filterdefinitionen speichern, die bei einer Anfrage referenziert werden können. Die konkreten Attributwerte können in einem Stored Query meist je Anfrage individuell definiert werden. So kann beispielsweise ein Query dahingehend definiert und gespeichert werden, dass er die Bounding Box eines Features ausgibt, welches durch eine ID referenziert wird. Der Filter muss somit nur einmal formuliert werden und kann anschließend mehrfach mit der Angabe einer ID verwendet werden. Das folgende Beispiel zeigt, wie ein solcher GET-Request (siehe Kapitel 4.1.2) mit der Angabe einer ID aussehen könnte:

```
http://my.example.de/wfs?service=WFS&request=GetFeature&version=2.0.0&
StoredQuery_ID=bboxQuery&gid=23.
```

Die Methoden *ListStoredQueries* und *DescribeStoredQueries* dienen, wie die Namen vermuten lassen, zur Auflistung und Beschreibung der einzelnen verwendbaren *StoredQueries* (ebd.). Zusätzlich zu den Operationen des Basisdienstes, gibt es nach OPEN GEOSPATIAL CONSORTIUM (ebd.) die folgenden zwei Operationen, die von sogenannten *transactional WFS* (WFS-T) implementiert werden. Durch diese ist nicht nur der Lesezugriff auf Geoobjekte, sondern auch die Manipulation dieser möglich (VOGT, 2011):

1. Transaction
2. LockFeature (Optional)

Durch Transaktions-Operationen können einzelne Instanzen von Features via WFS erstellt, modifiziert, ersetzt oder gelöscht werden. Um Konsistenzen zu bewahren, kann beispielsweise während einer solchen Bearbeitung durch *LockFeature* gewährleistet werden, dass kein anderer Client die Instanz modifizieren kann (OPEN GEOSPATIAL CONSORTIUM, 2010). Für die vorliegende Arbeit werden die Funktionalitäten eines WFS-T jedoch nicht benötigt, sodass diese nicht detaillierter behandelt werden. Relevanter sind im vorliegenden Anwendungsfall die lesenden Operationen der WFS-Dienste sowie deren Filtermöglichkeiten nach geographischen Eigenschaften.

Die Komplexität der Handhabung ist nach GEOINFORMATION UND LANDENTWICKLUNG NIEDERSACHSEN (2013) bei WFS deutlich höher als bei WMS. Die Implementierungen in den

<sup>12</sup>Die hier genannten Filter sind gängige geometrische Filter, wie sie in GIS-Systemen, räumlichen Datenbanken usw. zu finden sind. Sie prüfen, ob eine Geometrie eine andere beinhaltet (*Contains*), ob eine Geometrie in einem Umkreis liegt (*DWithin*) oder die eine Geometrie die andere berührt (*Touches*).

Softwareprodukten sind zu Teilen noch immer nicht ganz ausgereift und auf Grund der komplexen Daten, die durch WFS übertragen werden, benötigt der Nutzer Wissen zum jeweiligen Datenmodell und zur Geodatenverarbeitung (ebd.).

#### 2.6.4 Web Processing Service (WPS)

Ein Web Processing Service (WPS) definiert einen Mechanismus, mit dessen Hilfe eine vom Client beauftragte Verarbeitungsaufgabe durch den Server bearbeitet wird. Dabei können mehrere Einzelschritte zu einem Prozess zusammengefasst werden und mehrere, nicht zwingend in Relation stehende Prozesse pro Serverinstanz verarbeitet werden (MICHAEL und AMES, 2007). Dabei findet die Kommunikation laut Spezifikation basierend auf XML statt. Wie auch bei den anderen Diensten ist die *GetCapabilities*-Operation verpflichtend implementiert. Die Beschreibung eines bestimmten Prozesses kann durch die Methode **DescribeProcess** abgerufen werden, während der jeweilige Prozess unter Angabe von Eingabeparametern über **Execute** gestartet werden kann und das Resultat ausgegeben wird (OPEN GEOSPATIAL CONSORTIUM, 2018). Für die Nutzbarkeit eines WPS spielt die zu bearbeitende Datenmenge sowie die Geschwindigkeit der Internetanbindung eine entscheidende Rolle. Je nach Aufgabe kann es sinnvoller sein diese lokal durchzuführen, anstatt sie von einem externen Server bearbeiten zu lassen. Doch speziell bei einem komplexen Prozess, der eine lange Bearbeitungszeit braucht – auch wenn die Datenmenge gering ist – bietet sich ein WPS mit höherer Leistungsfähigkeit an (MICHAEL und AMES, 2007). Der Vorteil eines WPS ist, dass der Client bei einer Anpassung oder Verbesserung an den Algorithmen bei sich keine Änderungen vornehmen muss. Bei lokalen Datenverarbeitungen muss der Nutzer sich selber um die jeweilige Implementierung kümmern (ebd.).

Ein möglicher Prozess eines WPS könnte beispielsweise eine Bufferberechnung sein. Der Nutzer würde die Koordinaten einer Geometrie an den Dienst übertragen und weitere Parameter, wie die gewünschte Buffergröße oder das Ausgabeformat, angeben. Die Berechnung erfolgt anschließend auf der Serverseite und das Ergebnis wird im gewünschten Format, beispielsweise GML, an den Client geschickt. Durch die Nutzung externer WPS wird somit lediglich eine Internetverbindung aber keine komplexe GIS-Software für Geodatenbearbeitungen benötigt.

#### 2.6.5 Formate

Um digitale Geodaten zu speichern, zu verarbeiten und zu teilen gibt es unterschiedliche Formate. In Kapitel 2.4 wurde bereits erwähnt, dass laut der *Open Definition* auch das Format von „Open (Geo) Data“ offen sein muss, um eine Interoperabilität zwischen heterogenen Systemen zu gewährleisten. Daten, die in einem spezifischen Format codiert wurden, können nur von solcher Software gelesen werden, die die Unterstützung des Formats implementiert hat. Ohne offene Formate wäre auch die heutige Form des Internets schwer vorstellbar, da es sowohl auf offenen Protokollstandards wie TCP/IP als auch auf Datenformaten wie *Hypertext Markup Language* (HTML) basiert. Nur so ist eine Kommunikation über das Internet von Nutzern unterschiedlicher Betriebssysteme möglich (PADERTA, 2012). Doch auch proprietäre Formate erlangen teilweise eine große Marktdurchdringung. So ist beispielsweise das *ESRI Shapefile Format*, ein Vektorformat welches geographische Informationen und dazugehörige Attribute speichert, weit verbreitet.

Obwohl es sich hierbei um ein proprietäres Format der Firma ESRI handelt, ist die Spezifikation offengelegt und viele freie Softwareprodukte wie QGIS oder PostGIS unterstützen das Format. Bei der Betrachtung der Geoportale ist zu beobachten, dass viele Datensätze in diesem Format zum Download bereit gestellt werden. Die folgende Tabelle zeigt eine Auswahl an gängigen Geoformaten sowie solche, die für Text, Tabellen oder Bilder genutzt werden, da sie häufig in Kombination Verwendung finden.

Tabelle 2.3: Formate nach PADERTA (2012), aktualisiert und ergänzt

Format	maschinen- lesbar	Spezifikation verfügbar	offenes Format
<b>Geoformate</b>			
Geography Markup Language (.gml)	+	+	+
GPS Exchange Format (.gpx)	+	+	+
Keyhole Markup Language (.kml)	+	+	+
GeoJSON (.geojson)	+	+	+
GeoPackage (.gpkg)	+	+	+
Scalable Vector Graphics (.svg)	+	+	+
ESRI Shapefile Format (.shp, .shx, .dbf)	+	+	-
<b>Text-, Tabellen und Bildformate</b>			
Extensible Markup Language (.xml)	+	+	+
JavaScript Object Notation (.json)	+	+	+
Comma Separated Value (.csv)	+	+	+
JPEG Format (.jpg / .jpeg)	-	+	+
Geo-TIFF Format (.geotiff)	-	+	-

Mithilfe dieser Formate ist es möglich, Geodaten als Download bereitzustellen ohne beispielsweise einen Vollzugriff auf Datenbanken zu gewähren (PADERTA, 2012). Nachfolgend werden beispielhaft einzelne offene Formate kurz und komprimiert vorgestellt. Ein besonderes Augenmerk wird hier auf GML und JSON gelegt, da GML bei der Verwendung der WFS-Dienste oft als Standard genutzt wird, so auch in dieser Arbeit. Ferner wird das JSON-Dateiformat für den Dateiaustausch der eigens entwickelten Schnittstelle dieser Arbeit genutzt.

## XML

Das Format *eXtensible Markup Language (XML)* wurde vom World Wide Web Consortium (W3C) im Jahr 1998 veröffentlicht und ist ein Textformat für den Austausch von Daten (BRAY, PAOLI und SPERBERG-MCQUEEN, 1998). Es hat einen logischen Aufbau, der durch sich öffnende und schließende *Tags* von Elementen geprägt ist und vor allem für hierarchisch strukturierte Daten geeignet ist. Hieraus ergibt sich eine klare, leicht lesbare und einfach zu verarbeitende Struktur, die jedoch durch einen großen Overhead der Tags gekennzeichnet ist. Da XML ein generisches Format ist, welches kein Vokabular vorgibt, kann es vielseitig für verschiedene Anwendungsfälle eingesetzt werden, weswegen es sich großer Beliebtheit erfreut.

Code 2.2: Beispiel eines XML-Dokuments

```

1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <verzeichnis>
3   <titel>Geo-Schnittstellen</titel>
4   <eintraege>
5     <eintrag>
6       <stichwort>Web Map Service (WMS)</stichwort>
7       <beschreibung>Ein Web Map Service...</beschreibung>
8     </eintrag>
9     <eintrag>
10      <stichwort>Web Feature Service (WFS)</stichwort>
11      <beschreibung>Ein Web Feature Service...</beschreibung>
12    </eintrag>
13  </eintraege>
14 </verzeichnis>

```

Hilfreich sind dabei *XML Schema Definitionen* (XSD), die die Verwendung bestimmter Attribute und Elemente vorgeben (SPICHALE, 2017). Solch ein Schema könnte für das vorangegangene Beispiel wie folgt aussehen:

Code 2.3: Beispiel einer XML Schema Definition

```

1 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
2   <xs:element name="stichwort" type="xs:string"/>
3   <xs:element name="beschreibung" type="xs:string"/>
4   <xs:element name="titel" type="xs:string"/>
5   <xs:element name="eintrag">
6     <xs:complexType>
7       <xs:sequence>
8         <xs:element ref="stichwort"/>
9         <xs:element ref="beschreibung"/>
10      </xs:sequence>
11    </xs:complexType>
12  </xs:element>
13  <xs:element name="eintraege">
14    <xs:complexType>
15      <xs:sequence>
16        <xs:element ref="eintrag" maxOccurs="unbounded" minOccurs="0"/>
17      </xs:sequence>
18    </xs:complexType>
19  </xs:element>
20  <xs:element name="verzeichnis">
21    <xs:complexType>
22      <xs:sequence>
23        <xs:element ref="titel"/>
24        <xs:element ref="eintraege"/>
25      </xs:sequence>
26    </xs:complexType>
27  </xs:element>
28 </xs:schema>

```

Zuerst werden die Elemente `stichwort`, `beschreibung` und `titel` als Typ *String* definiert (Zeilen 2-4), also als Felder deren Inhalt Text ist. Anschließend wird in den Zeilen 5-12 das Element `eintrag` deklariert, welches aus einem Stichwort und einer Beschreibung besteht, gefolgt von dem Element `eintraege`, welches wiederum eine Liste aus Elementen mit dem Namen `eintrag` ist (Zeilen 13-19). Diese wird durch weitere Attribute in Zeile 16 genauer spezifiziert: `minOccurs` gibt an wie viele Einträge minimal in der Liste vorkommen müssen, während `maxOccurs` die Obergrenze festlegt. Hier darf die Liste also sowohl leer sein, als auch unbegrenzt viele Einträge enthalten. Das Element `verzeichnis` enthält lediglich das Element `titel` sowie die zuvor beschriebene Liste an Einträgen (Zeile 20-27). In Abbildung 2.12 ist das Schema noch einmal als UML-Diagramm<sup>13</sup> dargestellt.

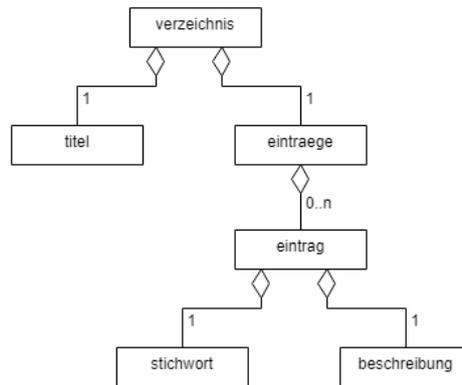


Abbildung 2.12: UML-Diagramm des Beispiel XML-Schemas

Das gezeigte Schema könnte nun über einen sogenannten *Namespace* am Anfang eines Dokuments, beispielsweise durch `xmlns:vz="http://www.mein-schema.de/verzeichnis/1.0"`, eingebunden werden. Anhand dieser Definition kann jeder Nutzer nachvollziehen, welche Elemente verwendet werden können oder müssen, wenn eine Instanz des Elements `<vz:verzeichnis>` im XML Dokument genutzt wird. Somit können klare Regeln geschaffen und neue Standards bei der Beschreibung von Inhalten entwickelt werden.

## GML

Mithilfe des vom OGC (siehe Kapitel 2.6.1) eingeführten Formats *Geography Markup Language (GML)* lassen sich räumliche und nicht-räumliche Daten sowie deren Beziehungen zueinander formal beschreiben (VOGT, 2011). Es erweitert XML um eine Vielzahl an Möglichkeiten zur Beschreibung von räumlichen Information. So können beispielsweise Geometrien, Koordinatenreferenzsysteme, Zeit oder Maßeinheiten standardisiert abgebildet werden. Das folgende Codebeispiel 2.4 zeigt die simple Beschreibung eines Punktes unter Angabe des Koordinatensystems durch das Attribut `srsName` nach dem Standard des GML-Formats.

GML zeichnet sich als der Standard für den internationalen Austausch von Geodaten ab (NEUMANN und ECKSTEIN, 2003). Mittlerweile existieren nach OPEN GEOSPATIAL CONSORTIUM

<sup>13</sup>UML steht für *Unified Modeling Language* (einheitliche Modellierungssprache) und ist die Standard Modellierungssprache für Software und Systementwicklungen (MILES und HAMILTON, 2006). Weitere Details hierzu können MILES und HAMILTON (ebd.) entnommen werden.

Code 2.4: Beispiel einer Verwendung von GML Point

```

1 <faculty>
2   <name>Z_GIS - Interfakultaerer Fachbereich Geoinformatik</name>
3   <university>Universitaet Salzburg</university>
4   <position>
5     <gml:Point gml:id="p1" srsName="http://www.opengis.net/def/crs/EPSC
6       /0/4326">
7       <gml:coordinates>13.039202,47.823613</gml:coordinates>
8     </gml:Point>
9   </position>
</faculty>

```

(2019b) sogenannte *application schemas*, die basierend auf GML für bestimmte Anwendungen weitere Standards beschreiben. Ein recht bekanntes Beispiel ist *CityGML*, welches zur Beschreibung von (3D) Objekten in Städten, wie Gebäuden, Brücken oder Straßen, verwendet werden kann (OPEN GEOSPATIAL CONSORTIUM, 2019a).

Das folgende Beispiel zeigt das Ergebnis einer WFS-Anfrage an den Berliner Bodenrichtwert-Dienst, bei welchem GML als Ausgabeformat gewählt wurde. Räumliche Informationen, wie die Bounding Box oder das Polygon der jeweiligen Bodenrichtwertzone sind in GML angegeben. Die weiteren Sachinformationen, die in der Ausgabe für den jeweiligen FeatureType definiert sind, können so mit der geographischen Repräsentation verknüpft werden:

Code 2.5: Beispiel eines GML Dokuments

```

1 [...]
2 <wfs:boundedBy>
3   <gml:Envelope>
4     <gml:lowerCorner>371191.95 5799675.01</gml:lowerCorner>
5     <gml:upperCorner>415786.29 5837254.86</gml:upperCorner>
6   </gml:Envelope>
7 </wfs:boundedBy>
8 <wfs:member>
9   <fis:s_brw_2018 gml:id="s_brw_2018.1002">
10    <fis:BRW>450</fis:BRW>
11    <fis:NUTZUNG>W - Wohngebiet</fis:NUTZUNG>
12    [...]
13    <fis:SHAPE>
14      <gml:Polygon gml:id="P1">
15        <gml:exterior>
16          <gml:LinearRing>
17            <gml:posList count="57">378075.89 5825093.59 [...]</gml:
18              posList>
19          </gml:LinearRing>
20        </gml:exterior>
21      </gml:Polygon>
22    </fis:SHAPE>
23  </fis:s_brw_2018>
24 </wfs:member>
25 [...]

```

## KML

Wenngleich in dieser Master Thesis keinerlei Daten im *Keyhole Markup Language* (KML)-Format verarbeitet werden, soll das Format kurz als Ergänzung zu XML und GML vorgestellt werden. KML wurde ursprünglich von der Keyhole, Inc. Google entwickelt und durch die Anwendung in dem digitalen Globus *Google Earth* bekannt. 2008 wurde das Format ein internationaler Standard des Open Geospatial Consortiums (siehe Kapitel 2.6.1). Da KML ebenfalls auf XML basiert ist die Ähnlichkeit der Formate offensichtlich. Das im GML-Abschnitt gezeigte Beispiel eines Punktes könnte im KML-Format durch ein sogenanntes *Placemark* dargestellt werden (siehe Code 2.6).

Code 2.6: Beispiel eines KML Dokuments

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <kml xmlns="http://www.opengis.net/kml/2.2">
3   <Document>
4     <Placemark>
5       <name>Z_GIS - Interfakultaerer Fachbereich Geoinformatik</name>
6       <description>Fachbereich der Universitaet Salzburg</description>
7       <Point>
8         <coordinates>13.039202,47.823613,0</coordinates>
9       </Point>
10    </Placemark>
11  </Document>
12 </kml>

```

Wird das dargestellte Beispiel in Google Earth geöffnet, wird ein Positionsmarker an der definierten Stelle mit dem Namen des Fachbereichs dargestellt. Auf Grund der Ähnlichkeit zu GML einigten sich OGC und Google darauf, dass es in Zukunft zu einer weiteren Harmonisierung der beiden Formate kommen kann, beispielsweise durch gleiche Geometrie-Repräsentationen (OPEN GEOSPATIAL CONSORTIUM, 2019c).

## JSON

*JavaScript Object Notation* (JSON) ist das von vielen Entwicklern bevorzugte Datenaustauschformat für Webanwendungen/-schnittstellen. Die Gründe hierfür sind, dass es schlank, einfach zu lesen und zu schreiben sowie maschinell unkompliziert zu parsen<sup>14</sup> und zu erstellen ist. Dabei besitzt es trotz des schlankeren Aufbaus alle Vorteile, die XML auch besitzt und ist außerdem deutlich performanter zu verarbeiten. NURSEITOV et al. (2009) zeigten in einer Fallstudie, dass JSON gegenüber XML signifikant schneller geparkt werden kann und dabei weniger Ressourcen verbraucht. Bei JSON handelt es sich um ein von Programmiersprachen unabhängiges Textformat, welches jedoch den Konventionen von C-basierten Sprachen (darunter C, C++, C#, Java, JavaScript) folgt, wodurch es sich großer Beliebtheit erfreut. Ein entscheidender Grund für die wachsende Beliebtheit ist, dass JavaScript – die Programmiersprache, auf der viele moderne Webinhalte basieren – nativen JSON-Support bietet und die Implementierung für Webentwickler

<sup>14</sup>Unter parsen versteht man das Analysieren von Datenstrukturen.

einen deutlich geringeren Aufwand im Vergleich zum XML-Format bietet (BARBAGLIA, MURZILLI und CUDINI, 2016). JSON basiert auf zwei Strukturen:

- **Name/Wert Paare**
- **geordnete Listen von Werten**

Diese Strukturen sind universell und in nahezu allen modern Programmiersprachen in unterschiedlichen Formen unterstützt, wodurch sich ein solches Austauschformat zwischen diesen Sprachen anbietet. Folgend werden die einzelnen Bestandteile von JSON vorgestellt, die für die in der Arbeit verwendeten Implementierungen von Relevanz sind:

**Objekte:** Ein Objekt ist eine ungeordnete Menge von Name/Wert Paaren. Ein Objekt beginnt mit einer offenen geschwungenen Klammer { und endet mit einer geschlossenen }. Jedem Namen folgt ein Doppelpunkt : gefolgt von einem Wert. Die einzelnen Name/Wert Paare werden durch Kommata , voneinander getrennt.

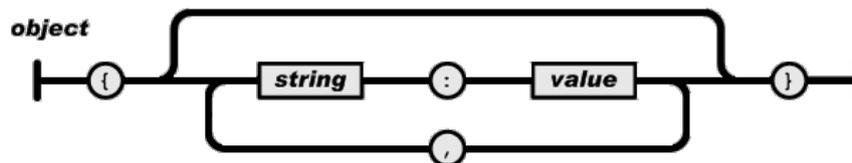


Abbildung 2.13: JSON Objekt (Quelle: ECMA INTERNATIONAL (2017))

**Array:** Ein Array ist eine geordnete Liste von Werten, welche im Gegensatz zu den Objekten mit eckigen statt geschweiften Klammern umschlossen wird [ ].

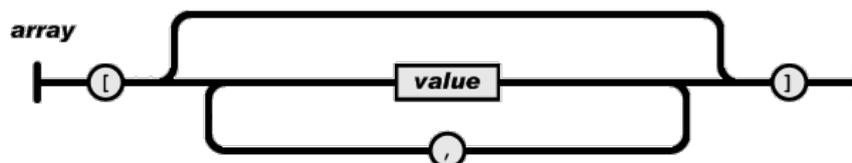


Abbildung 2.14: JSON Array (Quelle: ECMA INTERNATIONAL (2017))

**Wert:** Ein Wert kann die folgenden Ausprägungen annehmen (ebd.).

- Objekt
- Array
- Zeichenkette (*string*)
- Dezimalzahl (*number*)
- Ganzzahl (*integer*)
- Wahrheitswert (*boolean* - true, false)
- Null-Wert (*null*)

Hieraus ergibt sich, dass in einem Array auch verschiedene Objekte aufgelistet sein können, die wiederum andere Objekte oder Arrays bestimmter Werttypen enthalten (siehe Code 2.7).

Code 2.7: Beispiel eines JSON Datensatzes

```
1 {
2   "users": [
3     {
4       "id":1,
5       "firstName":"Max",
6       "lastName":"Mustermann",
7       "adress":{
8         "street":"Musterstrasse",
9         "number":1,
10        "addition":"a"
11      },
12      "verified":true,
13      "contracts":[
14        "18-0001",
15        "18-0002"
16      ]
17    },
18    {
19      "id":2,
20      "firstName":"Maria",
21      "lastName":"Musterfrau",
22      "adress":{
23        "street":"Musterstrasse",
24        "number":1,
25        "addition":"a"
26      },
27      "verified":true,
28      "contracts":[
29        "18-0003"
30      ]
31    }
32  ]
33 }
```

Für die Festlegung von Regeln, welche Inhalte in einem Dokument enthalten sein können oder müssen, wird JSON Schema verwendet, welches im nachfolgenden Kapitel vorgestellt wird.

## JSON Schema

*JSON Schema* ist ein JSON basiertes Format, welches ähnlich der XML Schema Definition bei XML Dokumenten eine Vereinbarung dafür schafft, welche Daten für eine spezielle Applikation benötigt werden, um mit dieser zu interagieren. Dabei ist das Format höchst modular und erlaubt

es, wiederkehrende Objekte zu definieren und wiederzuverwenden. Wenngleich sich das JSON Schema noch immer in einer sogenannten *Entwurfsversion* (engl. *draft*) befindet, wird es bereits produktiv von Firmen wie Google, Swagger oder Heroku eingesetzt (BARBAGLIA, MURZILLI und CUDINI, 2016). Im Header-Bereich des JSON Schemas sollte vorerst das verwendete Metaschema, selbst ein JSON Schema, angegeben werden. Dieses beschreibt die unterstützten Inhalte des zu erstellenden Schemas, wie die *Sektionen* und *Attribute*. Es wird somit beschrieben, in welcher Art und Weise der Nutzer das Schema formulieren kann und welche Möglichkeiten ihm dabei zur Verfügung stehen. Nach der Definition des verwendeten Schemas folgt eine eindeutige ID sowie ein Titel und eine Beschreibung des jeweiligen Schemas:

Code 2.8: Header Bereich eines JSON Schemas

```
1 {
2   "$schema": "http://json-schema.org/04/schema#",
3   "$id": "http://www.example.com/rate-request",
4   "title": "ComparablesService Rate Request",
5   "description": "A rate request for the ComparablesService including a reference
6     object as well as a list of objects which should be compared."
7   [...]
8 }
```

Die ID ist üblicherweise eine URL von der aus das Schema im Internet abgerufen werden kann, kann aber auch eine URI darstellen<sup>15</sup>.

In der *definitions*-Sektion können Objekte definiert werden, die an anderer Stelle referenziert werden können, um wiederkehrende Beschreibungen zu vermeiden. Eine Referenzierung kann nicht nur auf Objekte in der *definitions*-Sektion verweisen, sondern ebenfalls auf externe Schemata beispielsweise per URL. Hierdurch findet ein Aufbau statt, wie man ihn aus der objektorientierter Programmierung kennt - für sich stehende Objekte können in eigenen „Klassen“ implementiert und anschließend instantiiert werden.

Wie im Codebeispiel 2.10 gezeigt, können Referenzen auch geschachtelt werden. Das Objekt `position` wird innerhalb des `comparableObject` in Zeile 21 referenziert, welches wiederum bei der Item-Beschreibung des `objects`-Arrays verwendet wird (Zeile 33). Anpassungen an den im Schema mehrfach verwendeten „Klassen“ müssen somit nur einmalig vorgenommen werden und wirken sich auf sämtliche referenzierten Stellen aus.

Die Auswahl an Schlüsselwörtern aus Tabelle 2.4 können an den einzelnen Knotenpunkten des Schemas verwendet werden, um die Objekte für die Validierung weiter zu definieren. So können beispielsweise mithilfe des Attributs `required` die Pflichtfelder eines Objekts festgelegt werden. Hierfür werden die Feldnamen der Pflichtfelder in einem Array angegeben.

---

<sup>15</sup>Ein *Uniform Resource Identifier* (URI) besteht aus einer Zeichenfolge zur Identifizierung einer Ressource. Im WWW kann eine URI beispielsweise wie folgt aussehen: `http://example.com/users/myuser`. Groß- und Kleinschreibung spielen bei einer URI keine Rolle (SPICHALE, 2017). Jede Uniform Resource Locator (URL) ist auch eine URI, währenddessen eine URI nicht zwingend eine Repräsentation besitzen muss und damit keine URL ist. In den meisten Fällen können URI und URL jedoch synonym verwendet werden.

Tabelle 2.4: Validierungs-Schlüsselwörter im JSON Schema

Schlüsselwort	Referenz	Typ	Beschreibung
required	Objekte	Array[Strings]	Gibt an, welche Felder oder Objekte Pflichtfelder sind und zwingend benötigt sind. Alle nicht aufgeführten Felder sind optional.
additionalProperties	Objekte	boolean	Gibt an, ob zusätzliche Felder zu den im Schema definierten verwendet werden dürfen oder nicht. <b>false</b> , wenn keine Zusatzfelder erlaubt sind
minItems	Arrays	int	Mindestanzahl an Objekten in einem Array
uniqueItems	Arrays	boolean	<b>true</b> , wenn das Array nur eindeutige Objekte beinhalten darf. Doppelte Elemente führen zu einer negativen Verifizierung gegen das Schema.
pattern	Strings	RegEx	Über das pattern kann mithilfe einer RegEx angegeben werden, welches Muster der String aufweisen muss. Ein Beispiel wäre beispielsweise ein Datum im deutschen Format: 01.01.1990.

Während der String `epsg` für das `position`-Objekt nicht zwingend benötigt wird und alternativ der `default`-Wert verwendet wird, sind die Felder `lat` und `lon` Pflichtfelder (Codebeispiel 2.11 Zeile 21). Die Abwesenheit eines der beiden Felder bei einer Instanz würde zu einer negativen Validierung führen:

```
$.reference.position.lat: is missing but it is required
```

Ferner kann im Schema festgelegt werden, dass mindestens ein Vergleichsobjekt übermittelt werden muss und dass bei einer Liste mehrerer Objekte jedes einzelne einmalig sein muss (Zeile 6-7):

Code 2.9: Die Attribute "minItems" und "uniqueItems"

```
1 "objects":{
2   "type":"array",
3   "items":{
4     "$ref":"#/definitions/comparableObject"
5   },
6   "minItems":1,
7   "uniqueItems":true
8 }
```

Code 2.10: Definitionen und Referenzen im JSON Schema

```
1 [...]
2 "definitions":{
3   "position":{
4     "type":"object",
5     "properties":{
6       "lat":{ "type":"number" },
7       "lon":{ "type":"number" }
8     }
9   }
10 },
11 "properties":{
12   "positions":{
13     "type":"array",
14     "items":{
15       "$ref":"#/definitions/position"
16     }
17   }
18 }
```

Code 2.11: Das "required"-Attribut

```
1 "position":{
2   "type":"object",
3   "properties":{
4     "lat":{
5       "type":"number",
6       "description":"The lat coordinate.",
7       "examples":[52.46782684326172]
8     },
9     "lon":{
10      "type":"number",
11      "description":"The lon coordinate.",
12      "examples":[13.340462684631348]
13    },
14    "epsg":{
15      "type":"number",
16      "description":"The EPSG code the coordinates are referenced to.",
17      "default":4326,
18      "examples":[4326,25833]
19    }
20  },
21  "required":["lat", "lon"]
22 }
```

Ebenfalls sehr hilfreich ist die Verwendung sogenannter *Regular Expressions* (RegEx) über das `pattern`-Attribut. Mithilfe dieser regulären Ausdrücke kann ein Textwert gegen ein bestimmtes Muster geprüft werden. Die Möglichkeiten reichen von einer Festlegung auf eine reine Großschrei-

bung bis hin zu komplexeren Zeichenketten, wie Datumsformate, URLs oder E-Mailadressen<sup>16</sup>. Außerdem können mithilfe von RegEx Koordinaten auf Gültigkeit geprüft werden, sofern diese als String übertragen werden. Der folgende Ausdruck akzeptiert beispielsweise nur Latitude-Werte, die zwischen -90 und 90 liegen. Sofern ein Dezimaltrenner in Form eines Punktes angegeben wird, müssen weitere Zahlen folgen, um den String als valide zu prüfen:

$$\text{^\{\\+|-\\}?(?:90(?:\{?:\\.01,\\}?)|(?:[0-9]| [1-8] [0-9])(?:\{?:\\. [0-9] 1,\\}?)\}$}$$

Gültige Werte sind demnach 45.23, 90.0 oder 89.98 während 90.1, 47. oder -180 als ungültig geprüft werden.

Durch die genannten Definitionen von Regeln im Schema werden manuelle Validierungen überflüssig und es können Standards geschaffen und verwendet werden.

## GeoJSON

Das GIS-Datenformat *GeoJSON* ergänzt das soeben ausführlich vorgestellte JSON Dateiformat um die Möglichkeit, geographische Informationen standardisiert abzubilden – äquivalent zu GML, welches das Format XML erweitert. Zu den unterstützten Geometrien gehören Punkte, Linie, Polygone und mehrteilige Typen der Geometrien (MultiPoint, MultiLine und MultiPolygon), die wiederum in einer *GeometryCollection* kombiniert werden können (BUTLER et al., 2016). Um das bereits bei GML und KML gezeigte Beispiel erneut aufzugreifen, wird in dem folgenden Codebeispiel ein Punkt in GeoJSON abgebildet. Die Geometrie wird parallel zu der Typenangabe und den zusätzlichen Attributen des Features definiert.

Code 2.12: Beispiel eines Punktes in GeoJSON

```

1 {
2   "type": "Feature",
3   "geometry": {
4     "type": "Point",
5     "coordinates": [13.039202, 47.823613]
6   },
7   "properties": {
8     "type": "faculty",
9     "name": "Z_GIS - Interfakultaerer Fachbereich Geoinformatik",
10    "parent": "Universitaet Salzburg"
11  }
12 }
13 }
```

<sup>16</sup>Für weitere Informationen zu *Regular Expressions* und deren Bestandteilen siehe STUBBLEBINE (2012) und FRIEDL (2007)

---

## 3 Dienste und Datensätze

### 3.1 Dienste und Datensätze

«Offene Daten lesbar für Mensch und Maschine. Das ist das Ziel». Dieser Slogan begrüßt die Besucher auf dem Berlin Open Data Portal, welches aktuell etwa 1773 Datensätze in 22 Kategorien zur Verfügung stellt (Stand: März 2019) (SENATSVERWALTUNG FÜR WIRTSCHAFT, ENERGIE UND BETRIEBE, 2019). Die Kategorien sind umfassend aufgestellt und reichen von *Arbeitsmarkt*, *Demographie*, *Geographie und Stadtplanung* über *Gesundheit* bis hin zu Themen aus *Umwelt und Klima* oder *Tourismus*. Dabei wächst der Datenbestand fortlaufend, wodurch den Bürgerinnen und Bürgern immer mehr Daten kostenfrei zur Verfügung stehen. Aus dieser Vielzahl von verfügbaren Datensätzen sollen für den Anwendungszweck relevante Inhalte identifiziert werden. Unterschieden werden können hierbei Dienste und downloadbare Datensätze in Form von Geoformaten oder z. B. Excel-Tabellen. Während die Dienste, im Zusammenhang mit dieser Arbeit speziell WFS und WMS mit der Funktion *GetFeatureInfo*, live über das Internet abgefragt werden, müssen die heruntergeladenen Datensätze lokal vorgehalten und bevorzugt in einer Datenbank gespeichert werden, um einzelne Werte abrufen zu können. Der Vorteil bei der Verwendung von Diensten ist einerseits, dass keine eigene Infrastruktur auf der Nutzerseite aufgebaut werden muss, um die Datensätze zu verwenden. Auf der anderen Seite sind die Daten, je nach Handhabung der temporalen Fortschreibung von Seiten der Anbieter (ein Dienst mit stetiger Fortschreibung oder ein Dienst pro Jahr), stets aktuell (GEOINFORMATION UND LANDENTWICKLUNG NIEDERSACHSEN, 2013). Außerdem können lediglich Teilbereiche abgefragt werden und die Leistung von komplexen Anfragen an einen externen Dienst kann performanter sein, sofern die externe Hardware auch bei vielen Zugriffen leistungsfähiger ist als die vom Nutzer verwendete. Ein weiterer Vorteil ist, dass bei einem OGC-Dienst oftmals eine große Auswahl an Ausgabeformaten und angebotenen Projektionssystemen zur Verfügung stehen (ebd.). Dem gegenüber steht, dass stets eine aktive, möglichst schnelle und stabile Internetverbindung benötigt wird und Ausfälle auf der Serverseite nicht vom Anwender selber behoben werden können. Außerdem kann die Performanz der Schnittstellen auch sehr schwach sein, wobei der Endnutzer keinerlei Verbesserungen vornehmen kann, außer die eigenen Anfragen bestmöglich zu optimieren. Bei der Nutzung von Datensätzen stehen diese auch offline zur Verfügung, wenngleich je nach Nutzerkreis und Anforderungen eine eigene leistungsstarke Infrastruktur für den Zugriff aufgebaut werden muss. Es bedarf außerdem einem administrativen Aufwand die Datensätze in eine Datenbank zu importieren, diese ggf. aufzubereiten und stets aktuell zu halten. Je nach Anzahl der verwendeten Quellen können diese Aufwände sehr hoch sein.

Eine Auswahl wertbeeinflussender Merkmale wurde bereits in Kapitel 2.2 behandelt. Für diese Faktoren wurden offene Geodatendienste im Untersuchungsgebiet gesucht und deren Nutzbarkeit analysiert. Dabei wurde verstärkt der Katalog nach VOSS und BAKKER (2017) berücksichtigt, der basierend auf einer Expertenbefragung, die im Rahmen einer Prioritätenanalyse durchgeführt wurde, die wichtigsten lagerelevanten Indikatoren beschreibt. Bei der Befragung wurden Experten

des niedersächsischen Gutachterausschusses befragt, deren Aussagen durch die bundesweit durchgeführte Befragung von Wertermittlungssachverständigen verifiziert wurden (VOSS und BAKKER, 2017). In der vorliegenden Arbeit wurden Dienste berücksichtigt, die einen der beiden OGC-Standards WFS oder WMS mit der Unterstützung der *GetFeatureInfo*-Operation implementieren und eine Nutzung gemäß der lizenzrechtlichen Bestimmungen erlauben. Zusätzlich wurden die Daten von OSM mithilfe der Overpass API (siehe Kapitel 3.2) auf Eignung geprüft, um auf die aktuellen POIs von OSM zugreifen zu können. Die nachstehende Tabelle 3.1 zeigt eine Auswahl der für relevant befundenen Dienste. Manche Datensätze, die thematisch von Bedeutung wären, wie beispielsweise der Gesamtlärmindex, welcher strategische Lärmkarten von unter anderem Straßen-, Schienen und Fluglärm zusammenfasst, stehen zum Zeitpunkt der Untersuchung nur als Datensatzdownload aber nicht als geeigneter Geoservice zur Verfügung. Somit werden sie im Zuge dieser Arbeit nicht berücksichtigt. Es bleibt abzuwarten, ob in Zukunft weitere Daten über entsprechende Dienste zur Verfügung gestellt werden. Zusätzlich zu den in Tabelle 3.1 gezeigten Indikatoren und Datensätzen konnten weitere bedeutsame Inhalte, wie die Altersarmut, das Wanderungssaldo und -volumen oder die Anzahl der sozialversicherungspflichtigen Beschäftigten, ausfindig gemacht werden. Wie bereits in der Abgrenzung der Arbeit (siehe Kapitel 1.4) erwähnt, werden im Zuge dieser Arbeit nicht alle verfügbaren und relevanten Datensätze berücksichtigt. Der Fokus liegt somit auf einer Auswahl verfügbarer Daten, die zukünftig durch weitere Inhalte ergänzt werden kann. Es fällt auf, dass die Bezugsjahre der einzelnen Datensätze teils weit auseinander liegen und von 2009 bis 2019 reichen. Hieraus ergeben sich Herausforderungen bei der Vergleichbarkeit, auf die in Kapitel 5 noch einmal eingegangen wird.

Tabelle 3.1: Auswahl relevanter WFS Dienste und OSM POIs

Indikator	Dienst	Datensatz	Bezugsjahr
Arbeitslosigkeit	WFS*	Arbeitslosenanteil	2016
		Arbeitslose: Veränderung	2014-2016
Bodenrichtwert	WFS*	Bodenrichtwert	2014/2019
Bevölkerungsstruktur	WFS*	Ausländeranteil	2016
		Einwohnerdichte	2017
Lärmbelastung	WFS*	Umweltgerechtigkeit: Kernindikator Lärmbelastung	2012
Luftbelastung	WFS*	Umweltgerechtigkeit: Kernindikator Luftqualität	2009
Nähe zur Natur	WFS*	Umweltgerechtigkeit: Kernindikator Grünversorgung	2012
Nähe zu Bildungseinrichtungen	Overpass**	POIs Schulen und Universitäten	2019
Nähe zu Gesundheitseinrichtungen	Overpass**	POIs Krankenhäuser	2019
Nähe zu Kindergärten	Overpass**	POIs Kindergärten	2019
Nähe zum Stadtzentrum	Overpass**	POI Stadtzentrum	2019
Einkaufsmöglichkeiten	Overpass**	POIs Supermärkte	2019

\* GeoNutz-BE Lizenz

\*\* ODC-ODbL Lizenz

Für die Nutzung der Geodatendienste der Senatsverwaltung für Stadtentwicklung und Wohnen Berlin müssen die Nutzungsbestimmungen der GeoNutz-BE eingehalten werden, während die OSM Daten unter der ODC-ODbL Lizenz veröffentlicht wurden. Beide Lizenzmodelle wurden bereits in Kapitel 2.5 vorgestellt. Demnach ist eine freie Nutzung für sämtliche Zwecke sowie eine Verschneidung und Weiterverarbeitung zugelassen. Gemäß § 2 GeoNutz-BE ist jeweils ein Quellenvermerk notwendig. Dieser wird bei der zu entwickelnden *Comparables-Schnittstelle* berücksichtigt, indem er über einen eigenen Endpunkt `/rate/sources` abrufbar gemacht wird und zusätzlich bei den relevanten Ergebnissen mit an den Client übertragen wird.

Zu den im Zuge des *Umweltatlas Berlin* erfassten Daten wurden detaillierte Informationen bezüglich der Methodik veröffentlicht, von denen im Folgenden exemplarisch vier kompakt vorgestellt werden. Bei den Untersuchungen zur **Grünversorgung** wurden erholungswirksame Grünflächen in Wohnungsnähe ( $< 500$  m) und Siedlungsnähe ( $\geq 500$  m bis  $< 1500$  m) untersucht. Diese haben speziell im städtischen Bereich positive Auswirkungen, da sie beispielsweise den Lärm mindern, die Luft filtern und das Temperaturniveau senken. Außerdem können die Grünflächen zur Erholung genutzt werden und tragen somit zur Gesundheit und dem Wohlbefinden der Bürger bei (SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN, 2015b). Für den Freiraumtyp *wohnungsnah* wurden bereits kleine Flächen mit einer Mindestgröße von 0,5 ha berücksichtigt, während die Flächen für den Freiraumtyp *siedlungsnah* eine Fläche über 10 ha aufweisen müssen. Als ausreichend versorgt gilt die Bevölkerung ab 6 m<sup>2</sup> wohnungsnaher Grünfläche bzw. 7 m<sup>2</sup> siedlungsnaher Grünfläche pro Einwohner. Anhand einer prozentualen Erfüllung dieses Versorgungsziels wurden drei Versorgungsgrade bestimmt: gut/sehr gut, mittel und schlecht/sehr schlecht. Bei den Untersuchungen zur Grünversorgung wurde herausgefunden, dass fast die Hälfte der Berliner gut bis sehr gut versorgt ist (47%), von denen wiederum nur 5% innerhalb des S-Bahn-Ringes wohnen. Jeweils rund ein Viertel entfällt auf die anderen Klassen (ebd.). Eine Visualisierung der Grünversorgung kann Abbildung 3.1 entnommen werden.

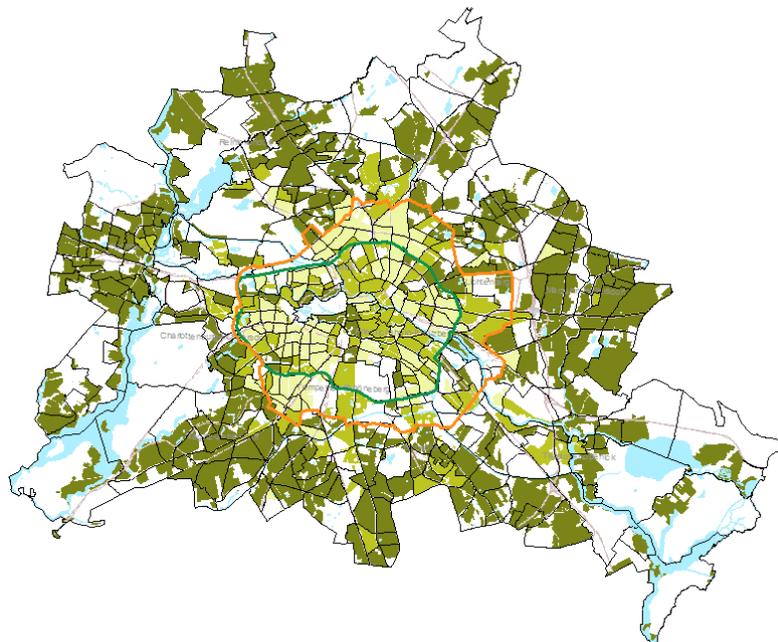


Abbildung 3.1: Grünversorgung (Quelle: Umweltatlas Berlin / Umweltgerechtigkeit: Kernindikator Grünversorgung)

Die **Lärmbelastung** umfasst solche Schallereignisse, „die durch ihre individuelle Ausprägung als störend und/oder belastend für Wohlbefinden und Gesundheit wahrgenommen werden“ (SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN, 2015a). Da Lärm ein gesundheitsbeeinträchtigender Faktor ist, der speziell in Städten als zentral angesehen werden kann, spielt er für die Wertbestimmung von Immobilien eine zentrale Rolle. Die Auswirkungen von Lärm sind vielseitig: Neben den bereits erwähnten ökonomischen Auswirkungen in den Miet- und Immobilienpreisen oder Lärmschutzkosten gibt es physiologische Auswirkungen wie Kopfschmerzen, Schlafstörungen oder Herz-Kreislaufprobleme. Auch psychologische Folgen, wie Stress, Leistungsbeeinträchtigungen oder Niedergeschlagenheit können auftreten. Außerdem seien die sozialen Lärmwirkungen genannt, die beispielsweise Aggressionen, Kommunikationserschwerung oder Nachlassen des Hilfeverhaltens zur Folge haben können (BAFU 2009). Lärm kann nach NIEMANN, MASCHKE und HECHT (2005) in verschiedene Kategorien eingeordnet werden: Industrie- und Gewerbe, Sport- und Freizeit, sowie der für Städte als Hauptursache für Lärm geltende Verkehr. Für die Bewertung des Lärms wurde nach SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN (2015a) ein monetärer Ansatz verwendet, bei dem der Nutzenverlust je Einwohner finanziell abgebildet wird. Im Durchschnitt liegen die externen Kosten durch Verkehrslärm in Berlin bei 45 € je Einwohner pro Jahr, bei geringer Belastung  $\leq 21$  € und bei der Belastungskategorie *hoch* bei 40-103 €. Es zeigt sich, dass die Lärmbelastung von den Stadtrandlagen zum Stadtzentrum steigt. Die lautesten Gebiete sind jene im Einzugsgebiet des Flughafens Berlin-Tegel im Nordwesten der Stadt (siehe Abbildung 3.2).

Auch die **Luftbelastung** ist speziell in urbanen Gebieten ein Problem, da die erhöhte Schadstoffbelastung zu Atemwegserkrankungen oder Herz-Kreislaufproblemen führen kann. Bewertet wurden die Belastungen von Feinstaub (PM<sub>2,5</sub>) und Stickstoffdioxid (NO<sub>2</sub>) in drei Klassen: *hoch*, *mittel* und *gering* belastet. Abbildung 3.3 zeigt, dass auch bei der Luftbelastung die Konzentration zum Stadtzentrum hin steigt. Erhöhte Werte zeigen sich verstärkt im Südwesten der Umweltzone (grünes Polygon) (KINDLER und FRANCK, 2015).

Zur Berechnung der **Einwohnerdichte** wurde die digitale Blockkarte 1:5000 (ISU 5) als räumliches Bezugssystem genutzt. Die Einwohnerdichte wurde für jeden einzelnen Block als Quotient aus Einwohnerzahl und Flächengröße errechnet. Die vorliegenden Adressdaten wurden mit den registrierten Einwohnern am Ort der Hauptwohnung in Berlin verknüpft und den Block- und Blockteilflächen zugeordnet. Die verwendeten Einwohnerdaten der Einwohnerregisterstatistik weisen leichte Abweichungen zu denen der amtlichen Bevölkerungsfortschreibung auf und sollten aufgrund der fehlenden Genauigkeit beispielsweise nicht als Verwaltungsregister verwendet werden. Außerdem können weitere Ungenauigkeiten auftreten, da aus Gründen der statistischen Geheimhaltung keine Einwohnerwerte unter 3 auftreten dürfen. Die Geheimhaltungsfälle der Werte 1 und 2 müssen daher verändert und auf andere Blöcke umgebucht werden, wodurch das Ergebnis nicht die realen Werte repräsentiert. Für den Anwendungsfall dieser Arbeit ist dieser Sachverhalt jedoch vernachlässigbar – ebenso wie die Tatsache, dass ein unmittelbarer Vergleich nur zwischen den Jahren 2016 und 2017 möglich ist, da zuvor eine andere Bezugsgrundlage verwendet wurde (SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN, 2019).

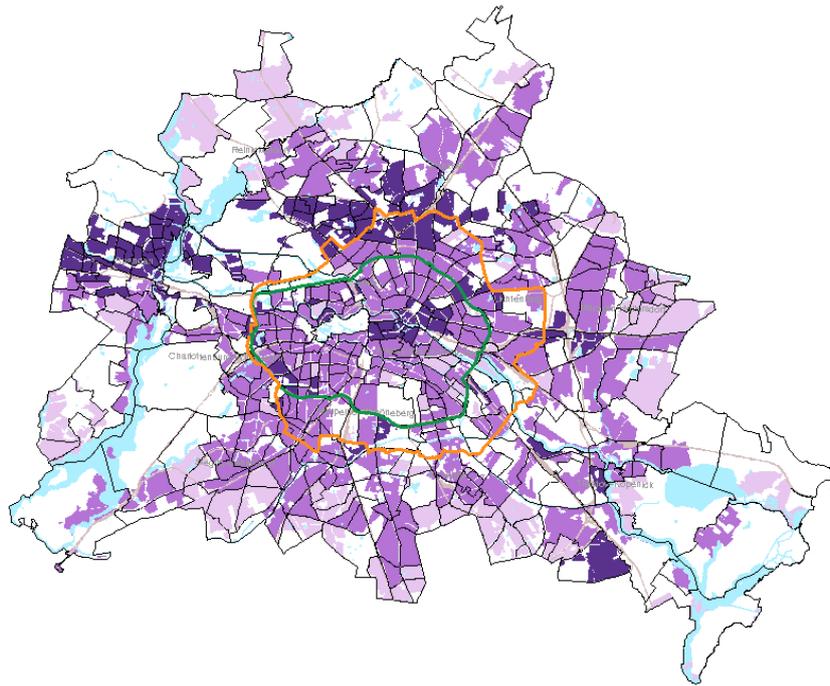


Abbildung 3.2: Lärmkarte (Quelle: Umweltatlas Berlin / Umweltgerechtigkeit: Kernindikator Lärmbelastung)

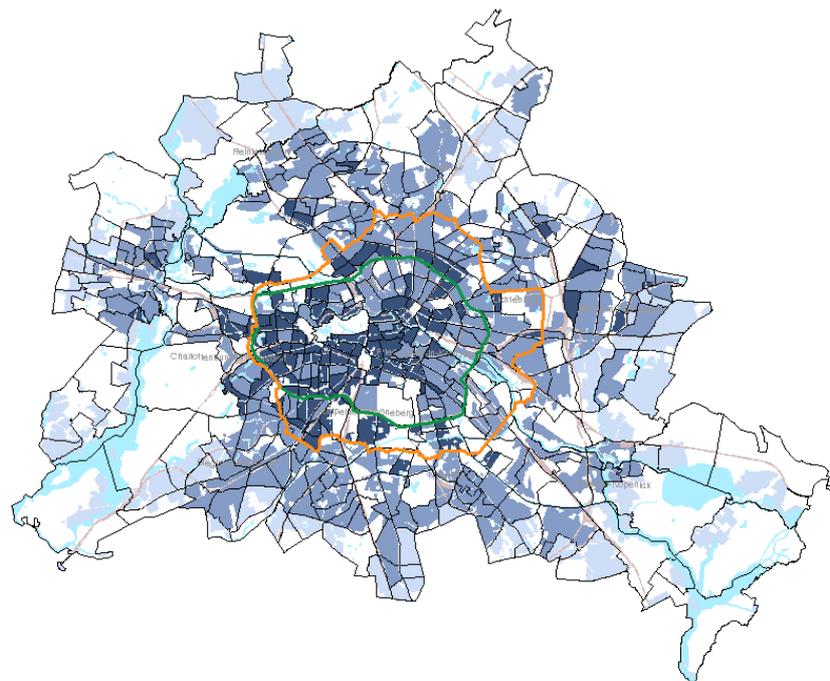


Abbildung 3.3: Luftbelastung (Quelle: Umweltatlas Berlin / Umweltgerechtigkeit: Kernindikator Luftbelastung)

## 3.2 Overpass API

Wie bereits in Kapitel 2.4.4 erwähnt, stellt OSM freie Kartendaten mit POIs für die Nutzer bereit. Eine Variante, um auf die Rohdaten zuzugreifen, ist der Download statischer Extraktionen. Einige Organisationen bieten für verschiedene Regionen wie Bundesländer, Staaten oder Kontinente aus den OSM-Daten extrahierte Datensätze an. Alternativ kann auch der Auftrag für eine Bereitstellung der Daten auf Basis einer definierten Bounding Box erfolgen (OPENSTREETMAP, 2019). Hieraus ergeben sich jedoch Nachteile wie die doppelte Datenhaltung, der Bedarf einer eigenen Infrastruktur zur Verarbeitung der Daten oder der Mangel an Aktualität. Wenn der Nutzer nur ein geringes Subset der Daten benötigt, wird trotz allem ein großer Overhead an Daten übermittelt. Über die sogenannte **Overpass API** können die Daten dynamisch über eine Schnittstelle abgerufen werden. Dabei werden Anfragen über eine eigene mächtige *Query Language* gestellt, die eine geographische und attributbasierte Filterung zulässt (siehe OPENSTREETMAP (ebd.)). So können beispielsweise nur bestimmte Informationen einer kleinen Region von den Servern abgerufen werden und die Datenmenge somit stark reduziert werden (ebd.).

Die in Codebeispiel 3.1 gezeigte Anfrage liefert die kartierten Schulen für die Bounding Box Berlins. Hierbei werden sowohl die **nodes** (Zeile 4) als auch die **ways** (Zeile 5) im JSON-Format (Zeile 1) übermittelt, da beide Typen von den Bearbeitern genutzt wurden, um die POIs zu erfassen. *Nodes* sind Punkte, während *ways* Linien oder Flächen darstellen können (in diesem Fall Flächen). Mithilfe der Option **center** in Zeile 6 wird für jedes Element des Typs *way* der Mittelpunkt berechnet und im Ergebnis übermittelt. Hierbei muss beachtet werden, dass der berechnete Mittelpunkt eines Polygons nicht zwingend in diesem liegen muss. Für eine genauere Betrachtung müsste basierend auf der Geometrie ein Punkt innerhalb des Polygons berechnet werden (in PostGIS heißt diese Funktion beispielsweise `ST_PointOnSurface`). Alternativ könnte auch der nächste Punkt der Polygongrenzen für eine Entfernungskalkulation verwendet werden – in dieser Arbeit wird dieser Aspekt jedoch bewusst nicht berücksichtigt und für eine vereinfachte Bearbeitung der ausgegebene Mittelpunkt verwendet.

Code 3.1: Beispiel eines Overpass API Requests

```

1 [out:json][timeout:30];
2 (
3   node["amenity"="school"](52.3418234221,13.0882097323,52.6697240587,13.7606105539);
4   way["amenity"="school"](52.3418234221,13.0882097323,52.6697240587,13.7606105539);
5 );
6 out body center;
```

Da manche Einrichtungen als Punkte erfasst sind und andere als Flächen, müssen beide Typen abgerufen werden, um ein möglichst umfassendes Ergebnis zu erhalten. Jedoch kann es hierdurch vorkommen, dass ein und die selbe Einrichtung in OSM doppelt enthalten ist – einmal als *node* und einmal als *way*. Dabei liegen unterschiedliche Koordinaten vor und zum Teil auch ein abweichender Name, sodass eine automatische Filterung der „Duplikate“ erschwert wird (speziell in Fällen bei denen es keine Überlappung von Fläche und Punkt gibt). Abbildung 3.4 zeigt ein solches Vorkommnis im Untersuchungsgebiet. Zwei nebeneinander liegende Kindergärten sind als *nodes* erfasst (gelbe Markierung), zusätzlich liegt jedoch der Mittelpunkt eines Polygons der



Abbildung 3.4: Doppelte Vorkommnisse in OSM

gleichen Kategorie in der Nähe (rote Markierung). Da im vorliegenden Anwendungsfall aber keine metergenaue Betrachtung der POIs oder eine namentliche Unterscheidung der benachbarten Instanzen notwendig ist, wird für diesen Fall keine Konfliktlösung implementiert. Für andere Anwendungsfälle könnte dieser Sachverhalt von größerem Interesse sein.

OSM gibt den Nutzern die Möglichkeit, die Features mit verschiedenen Tags zu versehen, um diese zu klassifizieren – beispielsweise `amenity=school` oder `shop=supermarket`. Eine Beschreibung der verschiedenen Kategorien ist zwar im Wiki von OSM festgehalten, dennoch sind die Unterschiede häufig sehr granular. So kann eine Einkaufsmöglichkeit über den Tag `store` beispielsweise als `department_store`, `general`, `convenience` oder `supermarket` deklariert werden (neben vielen weiteren Möglichkeiten). Die Unterschiede liegen hier z. B. bei den angebotenen Artikeln oder der Größe des Marktes. Auf Grund von sprachlichen, kulturellen und ontologischen Gründen sowie dem divergenten Verständnis von unzureichenden Definitionen, werden die Einträge in der OSM Dokumentation von den sehr heterogenen Beteiligten oft unterschiedlich interpretiert (BALLATORE und MOONEY, 2015). Dies hat zur Folge, dass die Daten nicht immer einheitlich verschlagwortet sind (ebd.). Je nach Anwendungsfall muss sich hierdurch intensiv mit den Möglichkeiten und Ausnahmen der Klassifizierungen auseinandergesetzt werden, um ein möglichst umfassendes Ergebnis der gewünschten POIs zu erhalten. Auch diesem Sachverhalt wurde zur Aufwandsminimierung und der eher nachgestellten Relevanz bei den Untersuchungen dieser Arbeit nicht bis ins kleinste Detail nachgegangen – er soll für andere Anwendungsfälle jedoch nicht unerwähnt bleiben.

Da die Daten von OSM unter der ODC-OdbL lizenziert sind (siehe Kapitel 2.5), ist die freie Nutzung unter Angabe der Datenherkunft möglich. Trotz allem sollen nach OPENSTREETMAP (2019) nicht zu viele Anfragen an die Overpass API gestellt oder zu große Datenmengen abgerufen werden (10.000 Anfragen oder 5 GB pro Tag), um die Nutzung für andere nicht einzuschränken. Dieser Aspekt wurde bei der Implementierung berücksichtigt – weitere Details können Kapitel 4.5 entnommen werden.

## 4 Schnittstellenimplementierung

In den Jahren 2010 bis 2016 hat sich die Anzahl der Web-Schnittstellen laut dem *The State of the API Integration*-Report um 758 % erhöht (CLOUD ELEMENTS, 2017). Das *World Wide Web* (WWW) hat sich von einer Ansammlung von Dokumenten hin zu einem großen Pool von verbundenen und miteinander interagierenden Diensten entwickelt (BARBAGLIA, MURZILLI und CUDINI, 2016). Aus diesem Grund sollen auch die in dieser Arbeit angestrebten Implementierungen in einer Webschnittstelle umgesetzt werden.

Die zu entwickelnde „Comparables<sup>1</sup>-Schnittstelle“ verfolgt das Ziel, in einem Eingangsdatensatz von Immobilienobjekten, die beispielsweise aus einem bestehenden Bestand stammen können, basierend auf einem Referenzobjekt geeignete Vergleichsobjekte zu identifizieren. Für jede Immobilie sollen hierbei automatisiert Kennzahlen aus den in Kapitel 3 genannten Diensten und Datensätzen ausgelesen und miteinander verglichen werden. Als Ausgabe erfolgt eine Liste der Immobilienobjekte mit einem ergänzten Rating der Eignung als Vergleichsobjekt und sofern vom Nutzer erwünscht den ermittelten Rohdaten. Vor der Programmierung der API wurde sich mit den Möglichkeiten der technischen Umsetzung auseinandergesetzt. Die zwei häufigsten Implementierungen von Webdiensten stellen Simple Object Access Protocol (SOAP) und REpresentational State Transfer (REST) dar. Beide Implementierungen haben Ihre Vor- und Nachteile und werden nachfolgend kurz vorgestellt, um die Wahl für die Entwicklung dieser Arbeit zu begründen.

### 4.1 RESTful HTTP

REpresentational State Transfer (REST) ist ein Architekturstil für netzwerkbasierte Systeme und wurde erstmals im Jahr 2000 von Roy Fieldings beschrieben. REST-Schnittstellen oder application programming interfaces (APIs) verwenden üblicherweise Methoden aus dem per RFC 2616 definierten *Hypertext Transfer Protocol* (HTTP), welches das Standardprotokoll im WWW darstellt. Dieses Protokoll ist dokumentenbasiert und hat klare Definitionen, wie der *envelope* (engl. für Umschlag) aussehen muss in dem Dokumente verschickt werden. Die Inhalte die vom Client zum Server oder vice versa transportiert werden, sind jedoch sehr offen gehalten (RICHARDSON und RUBY, 2007). Nachfolgend soll REST immer im Sinne von RESTful HTTP verstanden werden, da REST üblicherweise mit HTTP umgesetzt wird und andere Protokolle im Zuge dieser Thesis vernachlässigt werden können. Die Grundprinzipien von REST sind nach TILKOV et al. (2015):

- Eindeutige Identifikation sowie unterschiedliche Repräsentationen von Ressourcen
- Verwendung von Hypermedia
- Verwendung von HTTP-Standardmethoden
- Statuslose Kommunikation

---

<sup>1</sup> *Comparables* engl. für Vergleiche, Vergleichswerte, Vergleichsobjekte.

Ressourcen können bei REST durch URIs eindeutig identifiziert werden. Außerdem können sie in unterschiedlichen Formaten repräsentiert werden, die beispielsweise auf XML (siehe Kapitel 4.1.1) oder JSON (siehe Kapitel 4.2) basieren können. Mithilfe von Hypermedia – Links zu verschiedenen Medien wie Texten, Dokumenten, Bildern oder Videos – können per REST zusätzliche Informationen für den Client verknüpft werden, die per URI<sup>2</sup> auch von anderen Systemen aus als Ressource referenziert werden können. Durch die Verwendung von HTTP-Standardmethoden (siehe Kapitel 4.1.2) werden dem Client einheitliche Standards geboten, um die URIs korrekt aufrufen zu können und eine Kommunikation mit der API sicherzustellen. Für REST-konforme Anwendungen spielt die statuslose Kommunikation eine große Rolle. Auf der Serverseite gibt es aufgrund derer keinen Sitzungsstatus, der für mehrere Clientanfragen vorgehalten werden muss. Hierdurch kann der Dienst beispielsweise zwischen zwei Anfragen des Nutzers neugestartet werden oder der zweite Request von einer anderen Serverinstanz als der erste bearbeitet werden (beispielsweise durch Load Balancing<sup>3</sup>).

#### 4.1.1 Alternative SOAP

Die Alternative zu REST stellen klassische Simple Object Access Protocol (SOAP)-Schnittstellen dar. SOAP ist ein Kommunikationsprotokoll, welches auf XML zum Datenaustausch basiert. Schnittstellen, die auf diesem Standard basieren, werden mit der Web Service Description Language (WSDL) beschrieben, sodass zwischen Server und Client ein Vertrag existiert, welche Elemente für eine SOAP-Nachricht notwendig sind (RICHARDSON und RUBY, 2007). Um die Schnittstelle nutzen zu können, benötigt der Client diese detaillierte Beschreibungen zu dem Webservice, die in der WSDL-Definition zu finden sind. Es werden die verfügbaren Operationen in menschen- und maschinenlesbarer Form beschrieben. Hierfür wird wiederum die XML Schema Definition Language (XSDL) genutzt, um den Aufbau der Nachrichten und deren Datentypen zu definieren (RICHARDSON und RUBY, 2007; SPICHALE, 2017). Durch diese Standards bieten einige Entwicklertools die Möglichkeit SOAP-Clients oder -Services automatisiert zu erstellen (KUMARI und RATH, 2015). In den vergangenen Jahren gab es viele Schnittstellen, die auf dem SOAP-Standard basierten – mittlerweile macht dieser Anteil nur noch 15 % aus während REST bei dominierenden 83 % liegt (CLOUD ELEMENTS, 2017). Nichtsdestotrotz ist dieser Anteil noch immer beachtlich und ergibt sich mit Sicherheit aus den Vorteilen, die mit dieser Implementierung einhergehen: Protokollunabhängigkeit, die Verwendung eines eigenen Sicherheitsmodells, die einfache Wartbarkeit auf der Client-Seite und je nach Umsetzung die geringe Anzahl möglicher Fehler (TIHOMIROVS und GRABIS, 2016).

Für die Web-API-Implementierung dieser Thesis wurde sich trotz allem für den neueren Ansatz REST als Architekturstil entschieden, da SOAP eigene komplizierte Standards nutzt, während REST Standard-HTTP auf Basis von URIs verwendet. Nach TIHOMIROVS und GRABIS (ebd.) ist die serverseitige Implementierung hierdurch einfacher und wartungsärmer. Ferner können bei REST Daten in unterschiedlichen Datenformaten repräsentiert werden, während SOAP praktisch

---

<sup>2</sup>Bezüglich der URIs wurde sich bei dieser Arbeit auf Grundlage von MASSÉ (2011) für eine Worttrennung mit Bindestrichen entschieden, sodass `reference-object` anstelle von `referenceObject` oder `reference_object` verwendet wird.

<sup>3</sup>Beim Load Balancing wird die von Clients verursachte Last auf mehrere Serverinstanzen verteilt, um eine skalierbare und ausfallsichere Leistung gewährleisten zu können (SPICHALE, 2017).

auf XML beschränkt ist. Das alternative JSON-Format kann leichter und performanter geparkt werden und ist schlanker (siehe Kapitel 4.2). Außerdem werden SOAP-Nachrichten zusätzlicher um ein Header- und ein Body-Teil erweitert, was eine erhöhte Datenmenge und gegebenenfalls Leistungseinbußen zur Folge hat (TIHOMIROVS und GRABIS, 2016). REST-APIs unterstützen im Gegensatz zu SOAP Schnittstellen standardmäßig Caching und können bei Punkten, wie der geringeren Implementierungs-Komplexität, einer besseren Ausführungsgeschwindigkeit mit weniger Ressourcenbedarf und einer besseren Performanz, ihre Vorteile ausspielen (SPICHALE, 2017; TIHOMIROVS und GRABIS, 2016). Aus diesen Gründen erfreut sich REST mittlerweile auch bei den größten Webservice-Providern, wie Twitter, Yahoo und Flickr, großer Beliebtheit (TIHOMIROVS und GRABIS, 2016).

### 4.1.2 HTTP Methoden

Es gibt unterschiedliche Anfragemethoden (engl. *request methods*) für das HTTP, um Informationen zum Server zu senden oder von diesem abzurufen. Im Folgenden sollen diese Methoden nach FIELDING et al. (1999) und SPICHALE (2017) genauer beschrieben werden.

#### OPTIONS

Die Methode OPTIONS liefert die verfügbaren Kommunikationsoptionen einer Ressource. Üblicherweise liefert der Server den Statuscode 200 (siehe Kapitel 4.1.3) mit einem „Allow“-Header, der die möglichen HTTP Operationen auflistet.

```
HTTP/1.1 200 OK
ALLOW: HEAD, GET, OPTIONS
```

Zusätzliche Informationen können im Nachrichtenrumpf der Antwort enthalten sein.

#### GET

Die wohl am häufigsten verwendete Methode im Web ist GET. Sie wird vom Client dafür verwendet, um eine Ressource abzurufen. Mithilfe zweier Headerfelder kann ein sogenanntes *Conditional GET* ausgeführt werden:

- „If-Modified-Since“-Header: Daten nur empfangen, wenn sie sich seit einem bestimmten Zeitpunkt verändert haben.
- „If-Unmodified-Since“-Header: Daten nur empfangen, wenn sie seit einem bestimmten Zeitpunkt nicht verändert wurden.<sup>4</sup>

Durch das Headerfeld „Range“ kann nur ein Teil einer Entität übertragen werden.

<sup>4</sup>In diesem Fall würde der Server mit dem Statuscode 304 NOT MODIFIED antworten (siehe Kapitel 4.1.3)

## HEAD

Laut Spezifikation muss die Methode **HEAD** die gleichen Metadaten wie **GET** liefern. Der Unterschied liegt jedoch darin, dass **HEAD** keinerlei Nachrichtenrumpf an den Client übermittelt, sondern nur die Metadaten, wie die Größe und den Typen der angeforderten Ressource.

## PUT

Die **PUT**-Operation kann als Gegenstück zur **GET** Methode verstanden werden. Hierbei wird eine noch nicht existierende Ressource angelegt oder aber eine existierende aktualisiert. Hierbei dürfen die Daten jedoch vom Server sowohl ignoriert als auch geändert oder ergänzt werden. Hierbei ist wichtig, dass ein **PUT**-Request laut Spezifikation idempotent ist und somit auch bei mehrfacher Ausführung das gleiche Ergebnis zur Folge hat und beispielsweise keine doppelten Einträge erzeugt.

## POST

Während bei der **PUT** Methode der Client die URI der Ressource bestimmt, wählt bei der **POST**-Methode der Server die URI unter der der Datensatz angelegt wird. Diese wird dem Client im „Location“-Header zum Aufruf übermittelt. **POST** wird in der Praxis häufig dazu verwendet beliebige Operationen an einer Ressource durchzuführen, da es keine semantische Garantien erfüllen muss.

## DELETE

Die **DELETE**-Methode ist, wie der Name vermuten lässt, zum Löschen von Ressourcen vorgesehen. Der Aufruf ist auch erfolgreich, wenn die Ressource gar nicht existierte.

## PATCH

In RFC 5789 wird neben den bisher genannten Methoden außerdem **PATCH** definiert. Mithilfe dieser Methode können nur Teile einer Ressource aktualisiert werden anstatt, wie bei **PUT** immer die gesamte Ressource zu ändern. Dies kann den Overload einer Applikation verringern.

## TRACE

Zu guter Letzt sei die Methode **TRACE** erwähnt, die die Anfrage genau so zurück liefert, wie der Server sie empfangen hat. Diese Methode dient hauptsächlich dem Debugging von Verbindungen, um überprüfen zu können, ob die Anfrage auf dem Weg zum Server verändert wurde.

Die folgende Tabelle fasst die Methoden noch einmal zusammen und gibt an, ob diese sicher und idempotent sind. Eine Methode ist dann idempotent, wenn die mehrfache Ausführung der

Operation bei gleichbleibendem Serverstand das gleiche Ergebnis liefert. Als sicher gilt eine Methode dann, wenn sie „read-only“ ist und den Stand am Server unverändert lässt.

Tabelle 4.1: HTTP Methoden nach FIELDING et al. (1999) und SPICHALE (2017)

HTTP-Methode	Sicher	Idempotent
OPTIONS	Ja	Ja
GET	Ja	Ja
HEAD	Ja	Ja
PUT	Nein	Ja
POST	Nein	Nein
DELETE	Nein	Ja
PATCH	Nein	Nein
TRACE	Nein	Ja

### 4.1.3 HTTP Status Codes

Durch die Verwendung des HTTP Protokolls sind REST Architekturen durch die Nutzung von *HTTP Status Codes* (auch Response Codes) gekennzeichnet. Sie spielen bei der Fehlerbehandlung von APIs eine entscheidende Rolle, da neben der eigentlichen Antwort mithilfe der Codes mitgeteilt werden kann, ob die Anfrage erfolgreich oder fehlerhaft war. Zusätzlich werden durch den Statuscode detaillierte Informationen zu der Antwort übermittelt, um diese genauer zu klassifizieren und die Kommunikation von Server und Client zu verbessern. Die Einstufungen „erfolgreich“ oder „fehlerhaft“ sind nicht aussagekräftig genug, um die konkrete Verwendung der API zu beschreiben. Aus diesem Grund sollten einige der mehr als 70 standardisierten Statuscodes verwendet werden, um genauere Rückmeldungen zu geben (SPICHALE, 2017). Die Codes der Status werden in verschiedene Bereiche eingeteilt.

Statuscodes der Zahlenreihe **1xx** dienen der Information mit der Eigenschaft, dass die Bearbeitung bisher nicht abgeschlossen ist.

Die **2xx** Antwortcodes stehen für eine erfolgreiche Operation. Üblicherweise wird bei einem Erfolg, z. B. bei dem Abrufen einer Webressource, der Code **200 OK** verwendet. Hierbei wurde die Anfrage erfolgreich bearbeitet und das Ergebnis gegebenenfalls mit der Antwort übertragen. Es gibt jedoch auch weitere Codes, um das Operationsergebnis zu konkretisieren: **201 Created** – die Anfrage wurde erfolgreich bearbeitet und die angeforderte Ressource vor dem Senden der Antwort erstellt. Das „Location“-Header-Feld enthält eventuell die Adresse der erstellten Ressource. So gibt **204 No Content** an, dass die Anfrage erfolgreich durchgeführt wurde, die Antwort jedoch bewusst keine Daten enthält (RICHARDSON und RUBY, 2007). Es sollte eindeutig vermieden werden, dass Fehler mit dem Statuscode 200 getunnelt werden und die Antwort somit fälschlicherweise als erfolgreich ausgewiesen wird (SPICHALE, 2017). Die folgenden Absätze zeigen geeignete HTTP-Statuscodes für die Verwendung von Fehlern.

Zusätzlich zu den Informations- und Erfolgscodes gibt es Umleitungscodes, die im Bereich **3xx** liegen. Der Code **301 Moved Permanently** gibt beispielsweise an, dass die angeforderte Ressource

nicht mehr an der alten Adresse verfügbar ist und ab sofort unter der angegebenen Adresse im „Location“-Header-Feld verfügbar ist. Dieser Status wird auch *redirect* genannt.

Fehlercodes liegen in den Bereichen **4xx**, wenn der Fehler eher auf den Client zurückzuführen ist oder **5xx**, wenn es sich um einen Serverfehler handelt. Einige der relevanten Response Codes für Fehler können der Tabelle 4.2 entnommen werden. Die mit einem Stern versehenen Status werden in der eigenen Implementierung verwendet.

Die soeben vorgestellten Statuscodes können durch anwendungsspezifische Fehlercodes ergänzt werden, wenngleich dies nicht zwingend für jede Schnittstelle notwendig oder sinnvoll ist. Ein Beispiel hierfür ist die Twitter API, die in Kombination zu den HTTP Statuscodes eigene Fehlercodes zur detaillierten Beschreibung der aufgetretenen Konflikte nutzt. Im Answerheader werden die Fehler im JSON Format (siehe Kapitel 4.2) übertragen. So wird beispielsweise ein zusätzlicher anwendungsinterner Fehlercode 3 verwendet, um anzugeben, dass ungültige Koordinaten in den Parametern übergeben wurden (TWITTER, 2018):

Code 4.1: Fehler ungültige Koordinaten bei Twitter

```
1 {
2   "errors": [
3     {
4       "message": "Invalid coordinates.",
5       "code": 3
6     }
7   ]
8 }
```

Hierdurch wird deutlich, dass proprietäre Codes sinnvoll mit denen der HTTP Spezifikation kombiniert werden können. Zusätzlich kann in sinnvollen Fällen eine Verlinkung zur Dokumentation hinzugefügt werden, um den Nutzer zusätzliche Informationen bereitzustellen und manuelle Recherchen zu reduzieren (SPICHALE, 2017). Zur Fehlerübermittlung wurde nach NOTTINGHAM und WILDE (2016) in RFC 7808 der Mediatyp `application/problem+json` ins Leben gerufen. Mithilfe dieser JSON-Antwort können unter anderem Typ, Titel und Details des Fehlers an den Client übertragen werden.

Tabelle 4.2: HTTP Status Codes nach SELFHTML (2018) und SPICHALE (2017)

Code	Nachricht	Bedeutung
200*	OK	Die Anfrage wurde erfolgreich bearbeitet. Je nach Anfragemethode wird das Ergebnis in der Antwort übertragen.
204	No Content	Die Anfrage wurde akzeptiert und bearbeitet, jedoch enthält die Antwort bewusst keine Daten.
301	Moved Permanently	Auch <i>Redirect</i> genannt: Die angeforderte Adresse ist für die Ressource nicht länger gültig und steht nun unter der im „Location“-Header-Feld genannten URI bereit.
302	Found (Moved Temporarily)	Die angefragte Ressource wurde gefunden, hat derzeit jedoch eine andere URI. Trotz allem soll die verwendete Adresse für die gewünschte Ressource weiterhin angefragt werden.
400*	Bad Request	Die Anfrage war fehlerhaft und konnte nicht bearbeitet werden.
401	Unauthorized	Die Anfrage kann ohne gültige Authentifizierung nicht durchgeführt werden. Wie die Authentifizierung durchgeführt werden soll, wird im „WWW-Authenticate“-Antwortheader übermittelt.
404*	Not Found	Die vom Client angeforderte Ressource wurde nicht gefunden. Dieser Statuscode kann ebenfalls verwendet werden, um eine Anfrage ohne Angabe eines näheren Grundes abzuweisen.
415*	Unsupported Media Type	Das vom Client übertragene Format wird nicht unterstützt.
429	Too Many Requests	Der Client hat zu viele Anfragen in einem bestimmten Zeitraum gesendet.
500*	Internal Server Error	Dies ist ein „Sammel-Statuscode“ für unerwartete Serverfehler. Der Client erhält keinerlei Details über Art und Ursache des Fehler. Der Antwortrumpf kann für Details genutzt werden.
503*	Service Unavailable	Der Dienst steht beispielsweise wegen Überlastung oder Wartungsarbeiten aktuell nicht zu Verfügung.
504	Gateway Timeout	Aufgrund einer Zeitüberschreitung kann der Proxy die Anfrage nicht bearbeiten

## 4.2 JSON Schema Validator

In den letzten Jahren sinkt die Beliebtheit der XML-basierten Dienste stetig und bewegt sich mehr und mehr zu den JSON-basierten Alternativen. Unterstützen 2009 noch über 72 % der auf ProgrammableWeb<sup>5</sup> registrierten Schnittstellen XML waren 2013 mehr als 47 % der Dienste bereits JSON APIs. Ein Viertel der neuen Schnittstellen unterstützen zu diesem Zeitpunkt

<sup>5</sup>ProgrammableWeb ist das größte API Verzeichnis im Internet. Es beinhaltet fast 21.100 Webschnittstellen verschiedenster Kategorien und wird stetig aktualisiert (Stand März 2019).

XML gar nicht mehr (BARBAGLIA, MURZILLI und CUDINI, 2016). Das gewählte Format zur Repräsentation der Ressourcen ist ein wichtiger Faktor für die potenzielle Nutzbarkeit der Clients. Da JSON im Vergleich zu XML weniger wortreich und besser lesbar ist und dabei einen reduzierten Payload/Overhead aufweist, erfreut es sich großer Beliebtheit (SPICHALE, 2017). Offiziell gibt es für JSON jedoch keinen Standard, um JSON Dokumente zu beschreiben. *JSON Schema* kann jedoch genau für diesen Zweck genutzt werden und haben sich etabliert, um Schnittstellen nach dem REST Architektur Pattern zu definieren (BARBAGLIA, MURZILLI und CUDINI, 2016). In Kapitel 2.6.5 und 2.6.5 wurden das Dateiformat JSON und das dazugehörige JSON Schema bereits detailliert vorgestellt. Auf Grund der genannten Vorteile wurde sich für die Entwicklungen dieser Arbeit auf das JSON Format festgelegt.

Zum Stand dieser Arbeit ist der *draft-07* das aktuellste JSON Metaschema, wobei viele der existierenden Open Source JSON Schema Validatoren<sup>6</sup> für Java diese Version noch nicht unterstützen (siehe Tabelle 4.3). Aus diesem Grund wurde für die Schemata der Implementierung dieser Thesis auf den häufig unterstützten *draft-04* zurückgegriffen.

Tabelle 4.3: Java JSON Schema Validatoren (Stand: Februar 2019)

Bibliothek	neuste unterstützte Schema Version
java-json-tools json-schema-validator	draft-04
networknt json-schema-validator	draft-04
leadpony justify	draft-07
everit-org json-schema	draft-07

Aufgrund des Benchmark Projekts NETWORKNT (2018), welches drei der in Tabelle 4.3 aufgezählten Java Validatoren bezüglich Ihrer Geschwindigkeit bei der Validierung gegen ein JSON Schema vergleicht, fiel die Wahl der verwendeten Bibliothek auf den *networknt json-schema-validator*. Wie in der nachfolgenden Tabelle dargestellt, sind die alternativen Bibliotheken nach NETWORKNT (ebd.) fast bis zu 32 Mal langsamer:

Tabelle 4.4: Ergebnisse des Benchmark Tests nach NETWORKNT (2018) (Stand: Juli 2018)

Bibliothek	Benötigte Zeit	Geschwindigkeitsnachteil
java-json-tools json-schema-validator	7130 ms	3197 %
everit-org json-schema	1168 ms	524 %
networknt json-schema-validator	223 ms	0 %

Die dargestellten Benchmark-Ergebnisse wurden mit der Hardware eines Desktop-PCs verifiziert<sup>7</sup>. Da der Validator *Justify* im originalen Vergleich nicht enthalten war, wurde die Bibliothek mit einem entsprechenden Test ergänzt. Die Geschwindigkeitsunterschiede fielen bei der verwendeten Hardware weniger drastisch aus. Trotz allem liegen auch hier bis zu 1789 % Unterschied zwischen dem schnellsten und dem langsamsten Validator (siehe Tabelle 4.5).

<sup>6</sup>Ein JSON Schema Validator überprüft JSON Daten anhand eines angegebenen Schemas und meldet gegebenenfalls Fehler, die bei der Validierung aufgetreten sind. Beispielhafte Fehler wären das Fehlen von Pflichtfeldern, die Verwendung falscher Datentypen oder die Angabe nicht unterstützter Zusatzinhalte.

<sup>7</sup>Verwendet wurde ein Intel® Core™ i5 Prozessor mit 3,40 GHz sowie 16 GB RAM.

Tabelle 4.5: Ergebnisse des verifizierenden Benchmark Tests (Stand: Februar 2019)

Bibliothek	Benötigte Zeit	Geschwindigkeitsnachteil
java-json-tools json-schema-validator	4561 ms	1789 %
everit-org json-schema	898 ms	352 %
leadpony justify	560 ms	220 %
networknt json-schema-validator	255 ms	0 %

Um die erwarteten Schemata für die Comparables-Schnittstelle abzufragen, wurden verschiedene GET-Endpunkte unter dem Pfad `/json` implementiert (mehr dazu in Kapitel 4.4). So kann der Nutzer prüfen in welchem Format die Daten übermittelt werden müssen oder von der Schnittstelle zurückgeliefert werden. Das JSON Schema für einen der Rate-Endpunkte kann beispielhaft Anhang A.2.2 entnommen werden.

### 4.3 Verwendete Tools

Für die Erstellung der Schnittstelle und deren Funktionalitäten wurde auf verschiedene freie Tools und Bibliotheken zurückgegriffen, die in diesem Abschnitt kurz vorgestellt werden.

#### Git

Git ist eine mächtige Open Source Versionsverwaltung, die weltweit von Entwicklern für die verteilte Entwicklung genutzt wird (LOELIGER, 2012). Git schafft die Möglichkeit Tausende von Entwicklern gleichzeitig miteinander an einem Projekt arbeiten zu lassen und dabei schnell und effizient zu sein. Mithilfe des Revisionskontrollsystems hat der Nutzer die Möglichkeit all seine Entwicklungsschritte (üblicherweise an einem Text- oder Codeprojekt) zu historisieren, organisieren und nicht-linear durchzuführen. Hierfür nutzt Git Entwicklungszweige, sogenannte *branches*, die beispielsweise für neue Funktionsimplementierung genutzt und anschließend miteinander vereinigt werden können (*merging*). Für das *merging* existieren hilfreiche Git-interne Tools zur Konfliktlösung (ebd.). Bei der Verwendung von Git besitzt jeder Nutzer ein lokales Duplikat des Repositories<sup>8</sup> und dessen Versionsgeschichte, sodass für die meisten Operationen kein zentraler Server oder ein Internetzugang benötigt wird. Änderungen an den Dateien werden in einem *commit* gespeichert und mit einer Nachricht versehen. Durch die Arbeit mit Git kann jederzeit eine bestimmte Revision genutzt werden und bei fehlerhaften Änderungen auf einen funktionierenden Stand zurückgegriffen werden (ebd.). Abbildung 4.1 zeigt beispielhaft eine Visualisierung mehrerer paralleler *branches*, die zum Teil wieder zusammengeführt worden. Git ist ein komplexes und sehr hilfreiches Werkzeug für Entwicklungsarbeiten von kleinen bis hin zu sehr großen Projekten. Für weitere Details zum Funktionsumfang und den Möglichkeiten von Git wird auf LOELIGER (ebd.) verwiesen.

<sup>8</sup>Ein *Repository* (engl. für Lager) ist ein verwaltetes Verzeichnis bzw. Projektarchiv. Konkreter ausgedrückt ist ein Git-Repository eine Datenbank, welche sämtliche notwendigen Informationen enthält, um den Projektverlauf über Revisionen aufzubewahren und zu verwalten (LOELIGER, 2012).

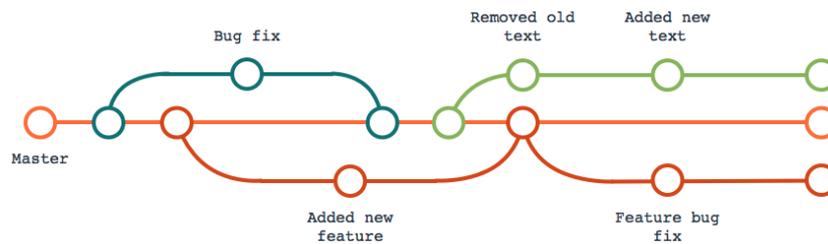


Abbildung 4.1: Beispiel mehrerer Git-Branches (Quelle: cPANEL (2018))

## Maven

Für die Organisation des Java Projektes wird das Build-Management-Tool **Maven**<sup>9</sup> verwendet, welches neben dem Ziel qualitative Projektinformationen bereitzustellen den Build-Prozess des Projektes vereinfacht und vereinheitlicht. So werden beispielsweise auch die Abhängigkeiten von verwendeten Bibliotheken übersichtlich verwaltet, was speziell bei der Arbeit mit großen Projekten und mehreren Modulen von großer Hilfe sein kann. Durch Maven werden *best practice* Richtlinien für Entwickler, die sich durch das standardisierte *project object model* (POM) leicht mit anderen Projekten vertraut machen können, bereitgestellt (THE APACHE SOFTWARE FOUNDATION, 2018). Das folgende Beispiel zeigt, wie in einer solchen `pom.xml` einzelne Bibliotheken, durch eine Gruppen- und eine Artefakt-ID zusammen mit der jeweiligen Versionsnummer, eingebunden werden können.

Code 4.2: Beispiel von dependencies in einer pom.xml

```

1 <dependencies>
2   <!-- JSON Tools -->
3   <dependency>
4     <groupId>com.fasterxml.jackson.core</groupId>
5     <artifactId>jackson-databind</artifactId>
6     <version>2.9.8</version>
7   </dependency>
8   <dependency>
9     <groupId>com.networknt</groupId>
10    <artifactId>json-schema-validator</artifactId>
11    <version>1.0.3</version>
12  </dependency>
13
14  <!-- GeoTools -->
15  <dependency>
16    <groupId>org.geotools</groupId>
17    <artifactId>gt-wfs-ng</artifactId>
18    <version>21.0</version>
19  </dependency>
20  [...]
21 </dependencies>

```

<sup>9</sup>Der Begriff *Maven* kommt aus dem Jiddischen und bedeutet so viel wie „Sammler des Wissens“ (THE APACHE SOFTWARE FOUNDATION, 2018).

## Jersey und Swagger

Für die Erstellung des RESTful Webdienstes wurde im Zuge dieser Arbeit das Open Source Framework **Jersey** für Java verwendet. Es bietet Unterstützung für die sogenannte JAX-RS-API (Java API for RESTful Web Services) und dient als deren Referenzimplementierung, wodurch sich REST-konforme Endpunkte stark vereinfacht erstellen lassen. Durch die von JAX-RS zur Verfügung gestellten Annotationen<sup>10</sup> können der Pfad, die Operationen, gegebenenfalls Parameter sowie die unterstützten Mediatypen leicht definiert werden (ORACLE CORPORATION, 2019).

Für das Design und die Dokumentation der Schnittstellen wurde **Swagger** verwendet. Das Open Source Framework hilft dabei RESTful Webservices zu gestalten, entwickeln, testen und zu dokumentieren. Mithilfe von zusätzlichen Annotationen können weitere Informationen bezüglich der Endpunkte für die Dokumentation und Definition ergänzt werden, beispielsweise welche Statuscodes der Antwort zu erwarten sind. Das Codebeispiel 4.3 zeigt die Kombination von Jersey/JAX-RS- und Swagger-Annotationen für einen Endpunkt<sup>11</sup>. Die ersten beiden Zeilen definieren durch die JAX-RS-Annotationen die Operation und den Pfad des Endpunktes. Im Anschluss erfolgt in den Zeilen 3-8 eine genauere Beschreibung des Endpunktes mithilfe der Swagger-Annotationen – es werden die Funktion sowie mögliche Statuscodes der Antwort mit ihrer Bedeutung beschrieben. In Zeile 9 folgt eine weitere JAX-RS-Annotation die angibt, in welchem Format die Antwort übermittelt wird.

Code 4.3: Beispiel von Annotationen Jersey/Swagger

```

1 @GET
2 @Path("/reference-object/{uuid}")
3 @ApiOperation(value = "Returns reference object by its UUID.")
4 @ApiResponses(value = {
5     @ApiResponse(code = 200, message = "Returns reference object"),
6     @ApiResponse(code = 404, message = "Reference object not existing"),
7     @ApiResponse(code = 500, message = "Internal error")
8 })
9 @Produces(MediaType.APPLICATION_JSON)
10 public Response getReferenceObject(@PathParam("uuid") String uuid){
11     [...]
12 }

```

So können die unterschiedlichen Endpunkte detailliert definiert und dokumentiert werden – schon bereits vor der Implementierung der Methodenlogik. Für die Visualisierung der Dokumentation kann das Projekt *Swagger UI* genutzt werden, um in einer interaktive HTML-basierte Benutzeroberfläche die zum Serverstart automatisch erstellte Swagger-Spezifikation einzusehen und

<sup>10</sup>In der Programmierung werden Annotationen (engl. *annotations*) für die Strukturierung des Quelltextes genutzt. Im Zusammenhang mit der Programmiersprache Java wird ab Version 5.0 als Annotation ein Sprachelement verstanden, mit dem Metadaten eingebunden werden können. Es beginnt mit einem @-Zeichen und wird durch einen Namen ergänzt. Optional kann eine Parameterliste in runden Klammern hinzugefügt werden. Annotationen können auch dazu genutzt werden weiteren Quellcode auszuführen oder automatisiert Dateien zu generieren.

<sup>11</sup>Im Code wird der Begriff *Universally Unique Identifier* (UUID) verwendet. Solch eine von der Open Software Foundation (OSF) standardisierte und in diesem Fall zufällig generierte Version-4 UUID dient häufig der eindeutigen Identifikation eines Datensatzes. Der Vorteil ist, dass sie automatisiert erzeugt werden kann und eine Dopplung als sehr unwahrscheinlich anzusehen ist. Basierend auf den Analysen des *birthday problems* nach (D. WAGNER, 2002) beträgt die Wahrscheinlichkeit eins zu einer Milliarde, um in 103 Billionen Version-4 UUIDs ein Duplikat zu finden.

Anfragen an die Schnittstelle zu schicken (SPICHALE, 2017; SMARTBEAR SOFTWARE, 2019). Abbildung 4.2 zeigt die Darstellung einiger der implementierten Endpunkte. Für die Generierung der `swagger.json`, welche die Definitionen der Schnittstelle für die Darstellung enthält, wurden von Swagger sowohl die JAX-RS- als auch die Swagger-Annotationen verarbeitet.

The screenshot displays the Swagger UI for an API. It lists several endpoints:

- POST** `/json/validate-request`: Validates given JSON against the expected schema for `/rate` endpoint.
- PUT** `/objects/reference-object`: Saves reference object and returns internal UUID.
- GET** `/objects/reference-object/{uuid}`
- DELETE** `/objects/reference-object/{uuid}`: Deletes a reference object by its UUID. This endpoint is highlighted in red.

The **DELETE** endpoint details include:

- Parameters**: A table with one parameter:
 

Name	Description
<b>uuid</b> * required	
string	
(path)	
- Responses**: A table with three responses:
 

Code	Description
200	Reference object deleted
404	Reference object not existing
500	Internal error

Below the **DELETE** endpoint, there is a **status** section with a **GET** endpoint `/status` that verifies if the service is running.

Abbildung 4.2: Endpunkte der Schnittstelle in Swagger UI

## JUnit

Zum Testen der einzelnen Programmkomponenten wurde das Framework **JUnit** verwendet. Es ist speziell dafür geeignet einzelne Programmeinheiten (engl. *units*) wie Klassen oder Methoden automatisiert zu testen. So können beispielsweise vor jedem Maven-Build alle Tests durchlaufen werden, um die Programmfunktionalität auch nach Modifikationen sicherzustellen oder Rückwärtskompatibilitäten, z. B. nach Versionsänderungen, zu überprüfen. Häufig wird getestet, ob durch einen bestimmten einzutretenden Fall ein anderer Teil des Codes aufgerufen wird oder ob zu erwartende Ergebnisse mit dem real resultierenden übereinstimmen. Mittlerweile gibt es Verfahren der „testgetriebenen Entwicklung“ (engl. *test-driven software development*), bei denen zuerst der Test für ein Stück Software geschrieben wird, um anschließend den zu testenden Code zu implementieren. Vorteil ist hierbei, dass möglichst fehlerarmer Code erzeugt wird und außerdem nichts implementiert wird, was nicht getestet wird – die Wahrscheinlichkeit Testfälle zu

übersehen oder eine geringe Testabdeckung zu haben, sinkt. Testklassen werden in einer eigenen Java-Paketstruktur abgelegt und ähnlich wie die Schnittstellen bei Swagger mit Annotationen markiert.

Das Codebeispiel 4.4 zeigt zwei sehr simple Tests, wie eine Hilfsklasse für die JSON-Schema-Validierung getestet wird. Eine JSON-Instanz ist für das getestete Schema valide, ein anderes auf Grund eines fehlenden Feldes fehlerhaft. Mithilfe der JUnit-Testmethode `assertEquals` wird die erwartete Anzahl der zurückgegebenen Fehler, mit dem real auftretenden Wert auf Gleichheit geprüft – unterscheiden sie sich, schlägt der Test fehl. Wenn das Schema dahingehend modifiziert wird, dass ein Feld umbenannt wird ohne dass sich die Instanz verändert, sind die automatischen Tests nicht mehr erfolgreich. Der Entwickler erhält somit Auskunft darüber, dass gegebenenfalls eine fehleranfällige Änderung vorgenommen wurde, da der `ResponseMessageBuilder` eine andere Struktur der Instanz erstellt. Der Fehler fällt schon vor der Ausführung auf und kann an einer der entsprechenden Stelle korrigiert werden: erstens im Schema, zweitens in der `ResponseMessageBuilder`-Klasse, die eine Instanz des Schemas erstellt oder drittens im Test selber, der auf Grund der bewusst durchgeführten Änderungen nicht mehr aktuell ist.

Code 4.4: JUnit Testbeispiele

```

1 public class JSONValidatorTest {
2     private JSONValidator validator = new JSONValidator();
3
4     @Test
5     public void shouldVerifyResponseMessageAsValid() throws IOException{
6         assertEquals("Should have zero errors",
7             validator.validate(ResponseMessageBuilder.build(200, "Success"),
8                 ResourcePath.SCHEMA_RESPONSE_MESSAGE).size(), 0);
9     }
10
11    @Test
12    public void shouldVerifyResponseMessageAsInvalid() throws IOException{
13        String json = "{\"code\":\"200\"}";
14        assertTrue("Should return errors due to missing message",
15            (validator.validate(json,
16                ResourcePath.SCHEMA_RESPONSE_MESSAGE).size() > 0));
17    }
18 }

```

Es können auch Endpunkte einer API getestet werden. Für die Erstellung des RESTful Webdienstes wurde im Zuge dieser Arbeit das Framework *Jersey* verwendet. Um die erstellten Methoden zu testen kann durch die Einbindung des *Jersey-Test-Frameworks* die `JerseyTest`-Klasse erweitert<sup>12</sup> werden und eine implementierte Service-Klasse, welche die Endpunkte beschreibt, als Ressource definiert werden. Das Codebeispiel 4.5 zeigt einen Test, der einen Funktionsablauf auf Richtigkeit testet. In dem Testszenario wird ein Test-Objekt über den `objects/referenceObject`-Endpunkt

<sup>12</sup>Im Quellcode findet in Java und vielen weiteren Programmiersprachen eine Vererbung durch das Schlüsselwort `extends` statt. Die Attribute und Methoden der Eltern- oder Superklasse werden an die Kindsklasse vererbt und können dort um weitere Eigenschaften ergänzt werden.

per PUT gespeichert (Zeile 6). Im Anschluss wird in Zeile 7 geprüft, ob der resultierende Statuscode 200 entspricht und das Speichern somit erfolgreich war. Ferner könnte an dieser Stelle mit einem anschließenden GET-Request überprüft werden, ob die Daten auch korrekt gespeichert wurden und den Eingangswerten entsprechen, da die Rückgabe des Statuscodes 200 hierfür keine Sicherheit gibt. Um das Beispiel schlank zu halten, wurde diese Überprüfung an dieser Stelle nicht vorgenommen. Die gespeicherte ID aus dem Response-Body wird nun verwendet, um das Objekt im Anschluss über die HTTP-Methode DELETE zu löschen. Auch hier wird der resultierende Statuscode auf den Wert 200 überprüft (Zeile 12), denn wenn das vorherige Speichern nicht erfolgreich gewesen wäre, würde laut Schnittstellendefinition beim Aufruf der Löschen-Methode der Fehlercode 404 - Not Found zurückgegeben werden, da das Objekt nicht existieren würde. Wenn das Objekt jedoch erfolgreich gelöscht wurde, wie es die Annahme im Beispiel ist, sollte es nun nicht mehr per GET-Methode abgerufen werden können und stattdessen den 404-Statuscode übergeben (Zeile 13). Mithilfe dieses Tests kann somit überprüft werden, ob das System ein Objekt erfolgreich speichern, die erstellte ID im richtigen Format zurück geben und das Objekt im Anschluss wieder löschen kann. Dabei wird die API direkt per HTTP-Methoden adressiert und die implementierten Statuscodes können bei der Überprüfung verwendet werden.

Code 4.5: JerseyTest Beispiel

```
1 @Test
2 public void shouldSaveAndDeleteValidReferenceObject() throws
    IOException{
3     //PUT-request for saving a test-object
4     ComparableObject testObject = new ComparableObject("Test", new
        Position(52.1321, 13.5245));
5     Entity<String> jsonEntity =
        Entity.entity(objectMapper.writeValueAsString(testObject),
            MediaType.APPLICATION_JSON);
6     Response response =
        target("objects/reference-object").request().put(jsonEntity);
7     assertEquals("Should return status 200", 200, response.getStatus());
8
9     //Using response body with saved id for DELETE-request
10    ResponseMessage message =
        objectMapper.readValue(response.readEntity(String.class),
            ResponseMessage.class);
11    String uuid = message.getMessage();
12
13    //Tests if the DELETE-request returns statuscode 200 ("OK")
14    assertEquals("Should return status 200", 200,
        target("objects/reference-object/" +
            uuid).request().delete().getStatus());
15    //Tests if now the GET-request returns statuscode 400 ("Not found")
16    assertEquals("Should return status 404", 404,
        target("objects/reference-object/" +
            uuid).request().get().getStatus());
17 }
```

## GeoTools

**GeoTools** ist eine freie Java-Bibliothek, die für geographische Operationen verwendet werden kann. Sie nutzt die *JTS Topology Suite*, welche ebenfalls eine Open Source Bibliothek ist und grundlegende geometrische Funktionen bereit stellt. Durch die Bibliotheken wird die *Simple Feature Access* Spezifikation des OGC implementiert und die üblichen Geometrietypen, wie Punkte, Linien und Polygone sowie deren Multi-Varianten, unterstützt. Räumliche Operationen sowie die Arbeit mit OGC standardisierten Filtern, Attributen und Formaten wie GML werden mithilfe von *GeoTools* erleichtert. In dieser Arbeit wird speziell das *gt-wfs-ng*-Modul für die Kommunikation mit den WFS-Diensten genutzt.

### 4.4 API-Aufbau

In den nachfolgenden Kapiteln wird der grobe Aufbau und die Funktionsweise der entwickelten Comparables-Schnittstelle dargestellt. Die Struktur basiert größtenteils auf einer *Model-View-Controller-Service*-Architektur (MVCS) mit einer *View-Controller-Service-Model*-Hierarchie (VCSM-Pattern) (siehe Abbildung 4.3). Ein *Controller* kann sowohl direkt auf ein Model (oder Modell) zugreifen als auch indirekt mithilfe eines Service-Moduls. Das *Service*-Modul kennt die darunterliegenden Modelle, kann auf diese zugreifen und sie modifizieren. Ferner ist durch den Service eine Kommunikation zu externen Datenquellen möglich. Neben den Programmabschnitten der Steuerung und Datenverarbeitung umschreiben *Model*-klassen hauptsächlich Arbeitsdaten des Programms. Sie umfassen die Eigenschaften von Objekten und dienen primär der Festlegung und dem Abrufen von Attributen einer Instanz. Sie kennen die über ihnen liegenden Services und Controller nicht und können nur indirekt z. B. durch Antworten mit ihnen kommunizieren (KOWARSCHICK, 2017). Zusätzlich kann ein *View* als Darstellungslayer zur Verfügung stehen. In diesem Fall kommuniziert der View nur mit den Controllern (oder in speziellen Fällen direkt mit den Modellen). Die Schicht des Views wird in dieser Arbeit ausgeklammert, da der Nutzer direkt mit dem Controller kommuniziert<sup>13</sup>.

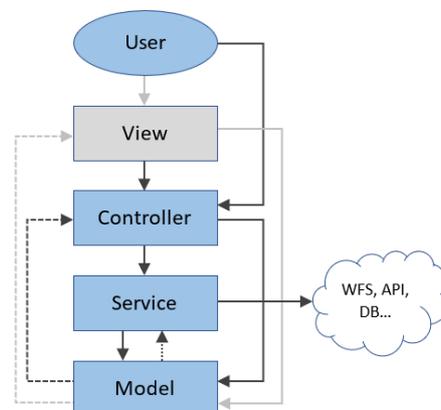


Abbildung 4.3: Schnittstellenarchitektur (eigene Darstellung nach KOWARSCHICK (2017))

<sup>13</sup>In gewisser Form kann für ein besseres Verständnis jedoch die im vorherigen Kapitel vorgestellte Swagger UI als View angesehen werden, da mit ihrer Hilfe über eine grafische Oberfläche eine Kommunikation mit dem Controller stattfinden kann. Jedoch ist dieser View nicht eigens implementiert und kein direkter Teil der Projektarchitektur.

## Endpunkte

Für die Implementierung der Endpunkte gibt es vier gesonderte *Controller*-Klassen, die die Schnittstelle nach Funktionen logisch gliedern:

- **StatusController** (`/status`): Gibt an, ob die Schnittstelle arbeitet und für Anfragen bereit ist. Zusätzlich kann beispielsweise die Erreichbarkeit der externen Dienste geprüft werden. In Zukunft könnten hier auch Status aktueller Operationen integriert werden.
- **JsonController** (`/json`): Implementiert Endpunkte zum Abruf von Schemata, die von der Schnittstelle erwartet oder als Antwort übermittelt werden. Ferner gibt es Schnittstellen, um das zu übermittelnde JSON gegen das erwartete Schema zu prüfen.
- **ObjectController** (`/objects`): Gibt dem Nutzer die Möglichkeit Objekte zu speichern, abzurufen und zu löschen.
- **RateController** (`/rate`): Implementiert Schnittstellen, um den Lagevergleich der Vergleichsobjekte zu einem Referenzobjekt vorzunehmen.

Zusätzlich existiert ein **TestController** (`/test`), der ausschließlich für den Entwicklungsbetrieb verwendet wird, um Benchmarks durchzuführen, fiktive Objekte zu verwalten oder einzelne Funktionsweisen zu testen. Diese Klassen enthalten hauptsächlich die Schnittstellendefinitionen und wenige Logik-Komponenten, um die Ergebnisse der Service-Klassen zu verarbeiten und an den Client zu übermitteln. So wird beispielsweise je nach Ergebnis der korrekte Statuscode in der Antwort gewählt. Durch diesen reduzierten Aufbau der Controller-Klassen ist der Vertrag zwischen Server und Client klar definiert und Programmteile, die Verarbeitungslogiken implementieren, können einfach ausgetauscht werden. Der Controller bleibt dabei recht klein und aufgeräumt.

Eine Liste der wichtigsten implementierten Endpunkte mit einer Kurzbeschreibung kann dem Anhang A.1 entnommen werden. Von einer detaillierten Beschreibung der erwarteten und verwendeten Mediatypen, Statuscodes usw. wurde auf Grund des Umfangs abgesehen. Die Swagger-Definition mit allen Endpunkten ist jedoch vorliegend. Die Schnittstelle wurde bisher nicht veröffentlicht. Mit einer Veröffentlichung sollte jedoch auch die Implementierung eines Lizenz-Endpunkts einhergehen, der sämtliche Informationen zu den Nutzungsrechten übersichtlich ausgibt. In Kapitel 4.5 werden einige der wichtigsten Funktionalitäten der entwickelten API anhand des Ablaufschemas dargestellt.

## Services

Die Service-Klassen werden primär von den Endpunkten aus aufgerufen und übernehmen den Großteil der Logikoperationen. Auch sie sind in verschiedene Funktionsbereiche gegliedert:

- **FilterService**: In dieser Klasse werden die Filteroperationen durchgeführt. Für spätere Erweiterungen ist hier beispielsweise auch die Filterung möglicher Vergleichsobjekte anhand optional übermittelter Attribute angedacht.
- **RateService**: Steuert den Prozess des Ratings. Stößt die Filterung und die Datenabrufe an, um anschließend die resultierenden Rückgaben zu vergleichen und zu klassifizieren.

- **WFSService**: Ruft die Daten von den WFS-Diensten ab und verarbeitet sie entsprechend. Auch die Zuordnung der Feature-Attribute zu den Objekten erfolgt in dieser Klasse.
- **OverpassService**: Ruft die relevanten POIs von der Overpass API ab und verarbeitet die Ergebnisse.

## Hilfsklassen

Zusätzlich gibt es unterschiedliche Hilfsklassen in dem sogenannten `util`-Package (Abkürzung für engl. *utility class* – die Hilfsklasse). Das `ProjectionUtil` konvertiert beispielsweise ein Koordinatenpaar in eine andere geographische Projektion und der `StudyAreaVerifier` überprüft mithilfe der `Contains`-Funktion, ob ein Punkt im Untersuchungsgebiet liegt. Der `DistanceCalculator` stellt für verschiedene Input-Parameter eine Distanzberechnung bereit. Auch der `JSONValidator`, welcher unter Verwendung des erwähnten *networknt json-schema-validator* eine JSON-Instanz gegen ein Schema prüft, ist in diesem Paket enthalten. Außerdem wurde ein `StorageHelper` implementiert, der für das Speichern verschiedener Instanzen (Objekte, Features der WFS-Dienste, POIs) verantwortlich ist. Wie das Speichern, Auslesen und Löschen einzelner Datensätze implementiert ist, kann angepasst werden, solange die zu implementierenden Methoden des Interface definiert sind. Somit kann beispielsweise eine PostgreSQL-Instanz zum Speichern verwendet werden oder aber die lokale Vorhaltung im Arbeitsspeicher. Je nach Anforderungen können hier Anpassungen vorgenommen werden ohne den Rest des Codes modifizieren zu müssen.

## Models

Wie bereits erwähnt, dienen Model-Klassen hauptsächlich der Speicherung von Arbeitsdaten. Das `ComparableObject` beispielsweise beschreibt ein Immobilienvergleichsobjekt. Die Klasse `ComparableObjectBase` besitzt eine ID als String sowie die Position. Für die Position wurde eine simple Klasse entwickelt, die ein Koordinatenpaar und den dazugehörigen EPSG-Code<sup>14</sup> enthält. Das `ComparableObject` erweitert diese Basisklasse um die Indikatoren und Entfernungen zu POIs sowie das Rating.

Eine weitere Modelklasse ist der `WFS`. Diese Klasse wird rein serverseitig verwendet und nicht an den Client übermittelt. Er enthält die Basis-URL für den `GetCapabilities`-Request, den relevanten `FeatureType`-Namen und dessen Attribute, die zu verwendende Version sowie den EPSG-Code (wird beispielsweise verwendet, um die übermittelte Position der `ComparableObjects` mit der Projektion des WFS zu vergleichen und diese ggf. zu transformieren). Über die Enumeration<sup>15</sup> `ExternalWFS` werden die unterschiedlichen verwendeten externen WFS mit ihren Eigenschaften definiert.

<sup>14</sup>Die Arbeitsgruppe *European Petroleum Survey Group Geodesy* entwickelte ein System weltweit eindeutiger Schlüssel für Koordinatenreferenzsysteme, Referenzellipsoide und Projektionen, welches seit 2005 unter gleichem Namen von der Nachfolgeorganisation *International Association of Oil & Gas Producers* weitergeführt wird. Aus einer Datenbank können sämtliche Informationen zu den EPSG-Codes ausgelesen werden, die sich als internationaler Standard durchgesetzt haben. Die zweidimensionalen Koordinaten von OSM basieren beispielsweise auf dem geographischen Koordinatenreferenzsystem WGS-84 und besitzen den Code EPSG:4326 (INTERNATIONAL ASSOCIATION OF OIL & GAS PRODUCERS, 2019).

<sup>15</sup>Eine Enumeration (oder *Enum*) ist eine Aufzählung.

Für die Verwendung der Overpass API wurde eine Model-Klasse für das Ergebnis und dessen enthaltene Elemente erstellt. Nach dem Aufbereiten der Rohdaten, können diese in einen POI transformiert werden. Diese POI-Elemente enthalten den Namen und die Position sowie ihren Typ (Schule, Krankenhaus usw.), der wiederum in einer Enumeration definiert ist.

## 4.5 Funktionsweise

Um die Performanz der implementierten Comparables-Schnittstelle zu erhöhen und dabei die Anfragen an die Overpass API möglichst gering zu halten, werden sämtliche benötigten Daten für das Untersuchungsgebiet nur beim Start bzw. einmal am Tag abgerufen. So wird eine tagesaktuelle Datenlage gewährleistet und die Verarbeitungszeit pro Request reduziert, da die Daten nicht bei jeder Anfrage erneut, sondern nur einmal täglich abgerufen und geparkt werden müssen. Der Abruf via Bounding Box, das Parsing und die anschließende Filterung für den genauen Polygonumgriff Berlins für alle berücksichtigten POIs benötigen unter Verwendung des Hauptservers der Overpass API im Durchschnitt  $< 10$  s. Hierbei werden rund 12.700 POIs verarbeitet, von denen 1850 in der Bounding Box aber im Berliner Umland liegen und somit für weitere Prozessschritte herausgefiltert werden. Diese Variante wurde gewählt, da auch die anderen Datenquellen die exakten Ländergrenzen Berlins verwenden, wenngleich aus Sicht der Immobilienwirtschaft die Verwendung der politischen Grenzen nicht die sinnvollste Wahl darstellt (FELDMANN et al., 2016). Eine Anpassung ist hier in Zukunft aber problemlos möglich, sodass auch benachbarte Lagen in Brandenburg berücksichtigt werden können.

Die gleiche Vorgehensweise wird auch bei den Daten der WFS-Dienste durchgeführt. Durch die Implementierung eines sogenannten `ServletContextListener` wird direkt beim Start des Webservices die Verbindung zu den einzelnen eingebundenen externen Diensten überprüft und deren Features abgerufen. Eine Aktualisierung der Features kann stetig stattfinden, jedoch wird auf diesem Weg der Aufwand für das Parsing pro Request sowie die Gesamtzahl an Anfragen an die externen Geodatendienste reduziert. Eine Alternative stellt die Verwendung der OGC-Filter bei einzelnen `GetFeature`-Requests pro Objekt dar – jedoch kommt es dabei zu unnötigem Netzwerkverkehr und die Verwendung der Filter verlief bei den Tests nicht immer fehlerfrei. Bei verschiedenen Filtern unterschiedlicher Dienste kam es immer wieder zu Fehlermeldungen bei vereinzelt Diensten. Zusätzlich treten hin und wieder Verzögerungen bei den WFS-Anfragen auf. Bei der Verarbeitung von 100 Objekten führen längere Bearbeitungszeiten bei nur wenigen Objekten bereits zu einem deutlich längeren Prozess. Um diese Fehlerquellen auszuschließen und die Anzahl an Schnittstellenanfragen gering zu halten, werden alle Features pro Dienst auf einmal abgefragt und temporär vorgehalten. Dieser Arbeitsschritt dauert für alle Dienste in Summe im Durchschnitt 20 Sekunden, weshalb beispielsweise eine tägliche Aktualisierung keinerlei Probleme darstellt.

Über die `/objects`-Endpunkte kann der Nutzer sowohl ein Referenzobjekt als auch ein Set an Vergleichsobjekten an das System übertragen. Hierfür wird die HTTP-PUT-Methode des jeweiligen Endpunktes verwendet: `/objects/reference-object` für das Referenzobjekt oder `/objects/comparables` für die Vergleichsobjekte. Beide Endpunkte liefern jeweils eine UUID zurück (siehe Fußnote auf Seite 68), mit deren Hilfe die Objekte nicht mehrfach übertragen

werden müssen. Stattdessen können die UUIDs dazu verwendet werden die Objekte per GET-Request wieder abzurufen, den Vergleichsprozess unter Angabe einer Referenzobjekt-UUID und einer Vergleichsobjektliste-UUID zu starten oder die Objekte wieder aus dem System zu löschen. Alternativ können die Objekte auch ohne Speicherung zur Verarbeitung an die API gesendet werden – für diesen Fall existiert ein weiterer Endpunkt mit dazugehörigem JSON-Schema. Alle Schemata, die von der API erwartet oder als Antwort geschickt werden, können über die jeweiligen `/json`-Endpunkte per GET-Request abgefragt werden.

Nachdem die Objekte übertragen wurden, kann mithilfe der IDs das Rating gestartet werden (`/rate/by-ids/{uuid-reference}/{uuid-comparables}`). Die Variablen in den geschweiften Klammern können mithilfe der `@PathParam`-Annotation ausgelesen werden. Intern werden die bereits verarbeiteten Objekte an Hand der übermittelten IDs abgerufen und an den `RateService` übermittelt. Hier findet vorerst eine Filterung der Objekte durch den `FilterService` statt. Die `ComparableObjects` mit der gleichen ID wie das Referenzobjekt sowie Objekte außerhalb des Untersuchungsgebiets Berlin, werden aus der Liste entfernt. Weitere Filtermöglichkeiten bezüglich optional übermittelter Attribute könnten in Zukunft an dieser Stelle ergänzt werden. Nun ermittelt der `WFSService` für jede Position der Vergleichsobjekte die Features der Open Geo Data Services, die die jeweiligen Punkt-Koordinaten enthalten, sofern dies nicht bereits bei der Speicherung der Objekte abgeschlossen wurde. Im Anschluss erfolgt die Zuordnung der relevanten Attribute der jeweiligen Features zu den Vergleichsobjekten – gleiches passiert auch für das Referenzobjekt. Parallel dazu werden für sämtliche Objektposition die Luftliniendistanzen zu den POIs aus den OSM Daten berechnet und ebenfalls mit dem Objekt verknüpft. Diese Distanzwerte ergeben sich aus der jeweiligen Luftlinienverbindung zwischen Start und Ziel, die, wie in Abbildung 4.4 gezeigt, mithilfe einer Linie in einem GIS visualisiert werden kann (BURGDORF, KRISCHAUSKY und MÜLLER-KLEISSLER, 2015). Die Distanzermittlung per Luftlinie ist ein etablierter Ansatz, der leicht implementierbar ist. Obleich der Nachteile, dass die Luftlinie nicht der Wegstrecke entspricht und diese Abweichung mit größerer Distanz steigt, verfolgt beispielsweise auch das Bundesinstitut für Bau-, Stadt- und Raumforschung (BBSR) diesen Ansatz für Analysen der Nahversorgung (ebd.). In der vorliegenden Arbeit werden keine Barrieren, wie Flüsse oder Waldflächen, berücksichtigt, sondern die vereinfachte Luftdistanz verwendet.

Nach der Verarbeitung aller Informationen wird die Liste der Vergleichsobjekte zurück an den `RatingService` übergeben. Dort werden alle Objekte, für welche relevante Indikatoren ermittelt werden konnten, mit dem Referenzobjekt verglichen und ein Rating erstellt. Die Kalkulation eines umfassenden Vergleichsratings kann auf Grund verschiedener Modellentwicklungen und Gewichtungen sicherlich eine komplette Forschungsarbeiten umfassen. Nichtsdestotrotz soll dieser Aspekt bei den Untersuchungen in einem vertretbaren Maß berücksichtigt werden. Auf Grund der vielfältigen Attribute der unterschiedlichen Indikatoren (siehe Tabelle 3.1) liegen unterschiedliche Skalenniveaus vor. Zum einen gibt es die *Nominalskala*. Dieses niedrigste Skalenniveau entscheidet lediglich über die Gleichheit oder Ungleichheit eines Merkmals und es kann keine Sortierung erfolgen (OESTREICH und ROMBERG, 2018). Im vorliegenden Anwendungsbeispiel tritt diese Skala bei der Nutzung auf, die mit dem Bodenrichtwert übergeben wird, z. B. Wohnen, Büro, Forst. Wenn es eine Rangordnung bei den Merkmalsausprägungen gibt, beispielsweise *gering*, *mittel*, *stark*, kann die *Ordinalskala* verwendet werden. Hierbei handelt es sich meist um qualita-

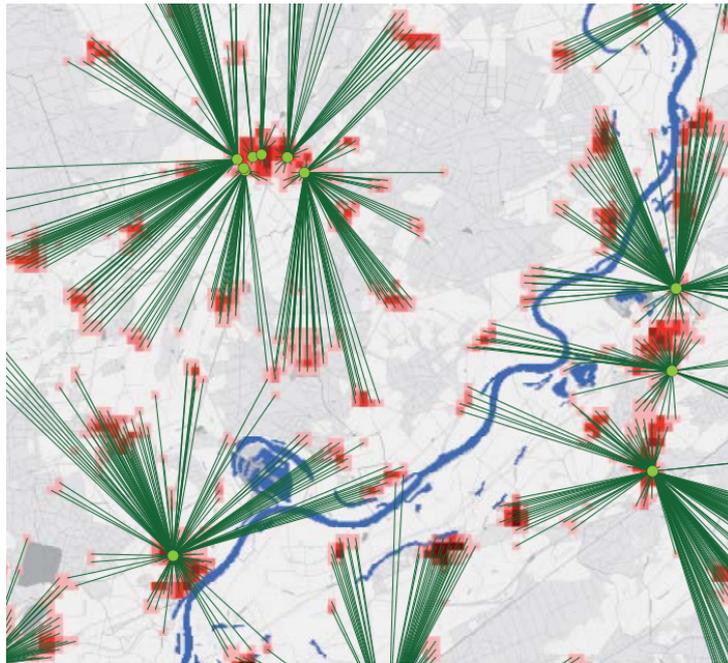


Abbildung 4.4: Visualisierung von Luftliniendistanzen (Quelle: BURGDORF, KRISCHAUSKY und MÜLLER-KLEISSLER (2015))

tive Merkmale (ebd.). Beispielsweise wird die Luftqualität oder die Lärmbelastung neben den quantitativen Messwerten über eine ordinale Skala abgebildet. Quantitative Merkmale hingegen verwenden Zahlenwerte, die gemessen oder gezählt werden können (ebd.). Die gerade erwähnten Messwerte der Luftverschmutzung wären neben der Anzahl der Arbeitslosen oder den Distanzen zu den POIs ein Beispiel für metrische Daten. Die dazugehörige Skala ist die *metrische Skala*, bei der neben der Rangordnung auch Abstände zwischen den Ausprägungen gemessen werden können. Diese Skala kann noch einmal in die *Intervallskala* und die *Verhältnisskala* unterteilt werden. Bei der Intervallskala können zwar Abstände bestimmt werden, aber Multiplikation und Division sind nicht sinnvoll (beispielsweise bei Jahreszahlen). Bei der Skala mit dem höchsten Skalenniveau, der Verhältnisskala, lassen sich neben Größenvergleichen, Addition und Subtraktion auch Multiplikationen und Divisionen sinnvoll durchführen (ebd.).

Unter Berücksichtigung der vorgestellten Skalen, werden die lagerelevanten Indikatoren miteinander verglichen. Hieraus lassen sich Ähnlichkeiten und Unterschiede ableiten, die im Durchschnitt wiederum in eine ordinale Skala *vergleichbar* (A), *weniger vergleichbar* (B), *kaum vergleichbar* (C) übertragen werden können. Nach BURGDORF, KRISCHAUSKY und MÜLLER-KLEISSLER (2015) und SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN (2015b) wird ein Einzugsbereich von 500 m als wohnungsnah/fußläufig betrachtet. Entfernungen von 1.000 bis 1.500 gelten nach SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN (2015b) als siedlungsnah und werden beispielsweise auch bei der Berechnung der Grünversorgung verwendet. Aus diesem Grund wurde sich für die Luftliniendistanzen zu den POIs für die gleichen ordinalen Klassen entschieden: fußläufig ( $< 500$  m), siedlungsnah ( $\geq 500$  m bis  $< 1.500$  m) und siedlungsfern ( $\geq 1.500$  m). Bei dem Vergleich zum Referenzobjekt wird der Abstand zwischen den Klassen verwendet. Befinden sich der betrachtete POI bei dem Referenzobjekt im fußläufigen Einzugsgebiet und beim Vergleichsobjekt ebenso, wird die Klasse

A (vergleichbar) gewählt. Befindet sich der nächste POI bei dem jeweiligen Vergleichsobjekt jedoch lediglich in einer siedlungsnahen Entfernung, wird die Klasse B (weniger vergleichbar) zugeordnet usw. In Tabelle 4.6 sind Beispiele für den Vergleich nach Klassen aufgelistet.

Es ist offensichtlich, dass diese Methode speziell bei Entfernungen nah der Klassengrenzen ihren Schwachpunkt hat. Eine Differenz von lediglich zwei Metern (499 m verglichen mit 501 m) ist ausreichend, um nur ein B-Rating bei der Vergleichbarkeit zu erhalten. Aus diesem Grund kann auch eine differenzbezogene Alternative verwendet werden, bei der die Unterschiede zwischen den verglichenen Entfernungen klassifiziert werden: geringe Differenz ( $< 500$  m), mittlere Differenz ( $\geq 500$  m bis  $< 1.500$  m), hohe Differenz ( $\geq 1.500$  m). Hierdurch verändern sich die Ergebnisse wie in Tabelle 4.7 dargestellt. Beide Varianten haben ihre Vor- und Nachteile. Wie im vorherigen Kapitel erwähnt, können bei richtiger Implementierung einzelne Logikkomponenten untereinander ausgetauscht werden, ohne die restlichen Prozesse anpassen zu müssen. Somit kann der Nutzer beim Aufruf einer der `/rate`-Endpunkte über einen URL-Parameter die bevorzugte Variante auswählen.

Tabelle 4.6: Entfernungsvergleich nach Klassen

Referenz in m	Referenzklasse	Vergleich in m	Vergleichsklasse	Ergebnis
354	fußläufig	286	fußläufig	A
354	fußläufig	512	siedlungsnah	B
354	fußläufig	2.150	siedlungsfern	C
1.009	siedlungsnah	487	fußläufig	B

Tabelle 4.7: Entfernungsvergleich nach Differenzen

Referenz in m	Vergleich in m	Differenz in m	Differenzklasse	Ergebnis
354	286	68	gering	A
354	512	158	gering	A
354	2.150	1.796	hoch	C
1.009	487	522	mittel	B

Für Datensätze, die bereits ordinale Attribute in Form von Strings zurückgeben, wurde ebenfalls die Vergleichslogik nach Klassen implementiert. Der Datensatz der Grünversorgung beispielsweise gibt nach SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN (2015b) drei Versorgungsstufen an den Client aus:

- gut, sehr gut
- mittel
- schlecht, sehr schlecht, nicht versorgt

Für die unterschiedlichen Rangfolgen wurde je nach Datensatz ein Mapping<sup>16</sup> implementiert, sodass die Vergleiche automatisiert berechnet werden können.

<sup>16</sup>Ein Mapping ist eine Wertezuordnung. Beispielsweise wird der Klasse „gut, sehr gut“ der Wert 1 und der Klasse „schlecht, sehr schlecht, nicht versorgt“ der Wert 3 zugeordnet. Somit können durch die numerische Zuordnung die Abstände zueinander berechnet werden. Diese sind wiederum auf die Ratings 0=A, 1=B, 2=C gemapped.

Für numerische Werte, beispielsweise die Einwohnerdichte, den Bodenrichtwert oder den Arbeitslosenanteil, werden, wie bei der Distanzberechnung, Grenzwerte festgelegt. Je nach Größe der Abweichung zueinander, werden die Differenzen vom Referenzwert einem Rating zugeordnet. Für alle Objekte und alle Attribute wird eine Einschätzung für die Eignung des Vergleichs berechnet. Für das finale Rating werden die numerischen Repräsentationen der Ratings addiert und der Mittelwert gebildet. In Zukunft könnten, wie bereits erwähnt, Gewichtungen integriert werden. Sollte für bestimmte Indikatoren an der vorliegenden Position kein Wert vorliegen, wird auch kein Vergleich vorgenommen. Die Anzahl der berücksichtigten Kennzahlen wird bei dem Rating-Ergebnis mit übergeben. Ein A-Rating mit 100 % vorliegenden Indikatoren ist also qualitativ höher einzuschätzen, als ein Rating mit nur 60 % vorliegender Vergleichsattribute.

Optional werden die Vergleichsobjekte nun basierend auf den Rating-Ergebnissen sortiert. Abschließend werden die in den Nutzungsrechten festgehaltenen Quellenangaben der Antwort hinzugefügt und an den Client gesendet. Abbildung 4.5 visualisiert beispielhaft die Rating-Resultate der implementierten API unter Verwendung der Differenz-Variante für den Vergleich der berechneten Luftliniendistanzen.

Standardmäßig erfolgt die Übermittlung mit den Rohdaten der einzelnen Indikatoren und dem berechneten Rating. Optional kann über einen URL-Parameter die Option `style=compact` an den Endpunkt übergeben werden, die eine komprimierte Übermittlung ohne Rohdaten aktiviert. Solche URL-Parameter werden durch ein Fragezeichen von der Basis-URL getrennt (`/rate/by-ids/{uuid-reference}/{uuid-comparables}?style=compact`) und können auf der Serverseite durch die JAX-RS `@QueryParam`-Annotation ausgelesen werden. Als weitere Option steht `outputFormat=geoJson` zur Verfügung, wodurch die Ergebnisse in einem gültigen GeoJSON-Format bereitgestellt und somit einfach in ein GIS-System integriert werden können, um eine Interoperabilität durch bereits existierende Standards zu gewährleisten.

Zusätzlich zu der objektbasierten Nutzung der API, gibt es eine Methode, die nur anhand der übermittelten Koordinaten in der EPSG:4326-Projektion die relevanten Standortindikatoren von den WFS-Diensten abrufen und die Daten an den Nutzer übermittelt.

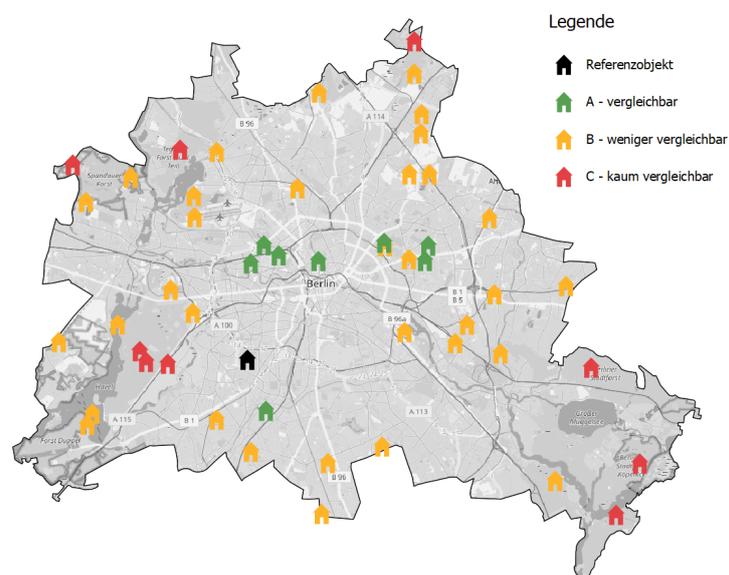


Abbildung 4.5: Rating-Resultate der implementierten API

## 5 Ergebnisse

Bei den Recherchen dieser Arbeit wurde festgestellt, dass es eine Entwicklung hin zu Darstellungs- und Downloaddiensten und weg von herunterladbaren Shapefiles oder ähnlichen gibt (siehe Kapitel 2.4). Im Untersuchungsgebiet Berlin wurden für einige wertrelevante Standortindikatoren freie Open Geo Data Services ausfindig gemacht. Insgesamt wurden über 20 Datensätze identifiziert, die sich für eine Einbindung über WFS in die Vergleichsschnittstelle aus lizenzrechtlichen sowie thematischen Gesichtspunkten eignen. Hiervon wurden im Zuge der Implementierungen neun Datensätze eingebunden, um den Fokus vorerst auf eine kleine Auswahl zu legen. Es wurden keine geeigneten WMS mit *GetFeatureInfo*-Unterstützung identifiziert, die nicht zusätzlich einen dedizierten WFS zur Verfügung stellten. Zusätzlich wurde die Overpass API für die OSM Daten auf Eignung geprüft und ebenfalls für integrierbar befunden, wenngleich die Verwendung dieser API nicht auf einem OGC Standard basiert. Hieraus ergeben sich weitere sechs integrierte Datensätze und aus der Kombination von WFS und Overpass API eine Summe von 15 Datensätzen für elf Indikatorenkategorien. Bei der Betrachtung der Bezugsjahre fällt auf, dass diese teils bis zu 10 Jahre auseinander liegen. Während es Datensätze wie den Bodenrichtwert gibt, die jährlich fortgeschrieben werden, steht der Großteil der Dienste nur für vereinzelte Jahre zur Verfügung. Aus diesem Grund kann für den Vergleich aktuell kein Stichtag eines bestimmten Jahres berücksichtigt werden. Stattdessen wir jeweils auf die neusten verfügbaren Datensätze zurückgegriffen, um bestmöglich einen vergleichsweise aktuellen Stand abzubilden.

Für die jeweiligen Einbindungen wurde die Performanz analysiert, deren Ergebnisse im folgenden erläutert werden. Für den Abruf der OSM POIs über die Overpass API konnte eine mittlere Bearbeitungszeit von etwa 1,6 s festgestellt werden. Die mit einer Abfrage abgerufene Anzahl an POIs lag zwischen 61 und 7.412 Elementen. Es zeigte sich eine Standardabweichung<sup>1</sup> von etwa 500 ms, jedoch gab es auch vereinzelte Ausreißer, die eine etwa 3 s längere Zeit benötigten, um vom Server verarbeitet zu werden. Ferner ergab sich eine signifikante mittlere bis hohe Korrelation<sup>2</sup> von  $r = 0.49$  für den Zusammenhang der Requestzeit und der Feature-Menge. Hierbei wurden für vier verschiedene POI-Klassen jeweils 50 Anfragen an die Schnittstelle gesendet und die Ergebnisse auf Korrelation und die Kennzahlen der deskriptiven Statistik überprüft. Diese Ergebnisse wurden mehrfach werktags zwischen 10 und 11 Uhr sowie am Abend gegen 22 und 24 Uhr durchgeführt, um tageszeitliche Schwankungen zu berücksichtigen. Die Zeiten ergaben ähnliche Werte, wobei die Standardabweichung am Abend geringer war (etwa 200 ms statt 500 ms bei einer maximalen Abweichung vom Mittelwert von nur 1 s). Als Server wurde der Hauptserver der Overpass API nach OPENSTREETMAP (2019) verwendet, der eine selbstständige Lastverteilung auf aktuell zwei Server vornimmt.

Die gleiche Untersuchung wurde ebenfalls für die Abrufzeiten der Daten über WFS-Dienste

<sup>1</sup>Die Standardabweichung gibt an wie groß die mittlere Abweichung vom Mittelwert für die gegebene Stichprobe ist (OESTREICH und ROMBERG, 2018).

<sup>2</sup>Die Korrelation gibt den Zusammenhang zweier Variablen an. Dabei bedeuten ein Korrelationskoeffizient von  $r = 0$ , dass keinerlei Zusammenhang besteht, während  $r = 1$  einen stark positiven Zusammenhang und  $r = -1$  ein stark negativen Zusammenhang beschreibt (ebd.).

durchgeführt. Dabei wurden pro Request zwischen 434 und 14.730 Features abgerufen. Im Mittel zeigten sich hier Bearbeitungszeiten von etwa 850 ms, wobei die Standardabweichung bei etwa 1 s und die maximale Requestdauer bei 8,5 s liegen. Außerdem sollte berücksichtigt werden, dass im Vergleich zu den Overpass Daten auf Grund der höheren Komplexität der Antwort und der hauptsächlichlichen Verwendung von Polygonen statt Punktkoordinaten das Parsing bei den WFS Anfragen deutlich mehr Zeit in Anspruch nimmt (über 90 % des Zeitanteils<sup>3</sup>). Eine Korrelation zwischen der Feature-Anzahl und den Bearbeitungszeiten ließ sich nicht feststellen ( $r = -0.02$  bei einer Signifikanz von 0.792 = nicht signifikant), wobei die Standardabweichung bei dem Dienst mit der höchsten Anzahl an Features am größten war. Es lässt sich festhalten, dass die

Tabelle 5.1: Deskriptive Statistik der Abfragezeiten in ms

Kennzahl	N	Mittelwert	Std.-Abw.	Min	Max
Abfragezeit (Overpass)	200	1590	497	760	4418
Abfragezeit (WFS)	200	853	1054	84	8487

Streuung bei den Requestzeiten der WFS-Dienste deutlich höher ist und daher einzelne Abfragen fast 10 s benötigen, wenngleich im Mittel geringe Zeiten erzielt werden. Diese größere Spannweite schmälert die Zuverlässigkeit der schnellen Abfrageergebnisse. Die Abbildungen 5.1 und 5.2 visualisieren noch einmal die ermittelten Abrufzeiten eines Beispielbenchmarks werktags gegen 11 Uhr in Form eines Scatterplots.

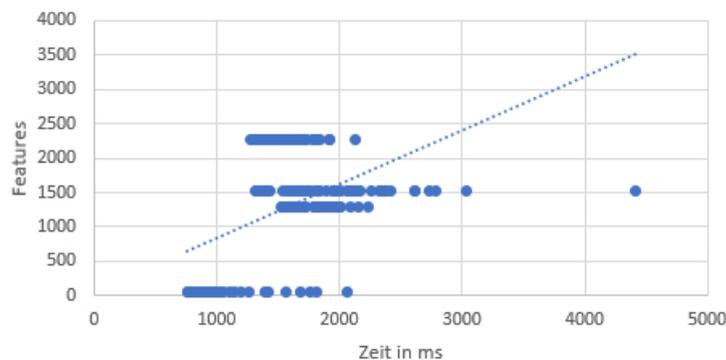


Abbildung 5.1: Abrufzeiten der Overpass Daten

Zur Verifizierung der Performanz wurde neben der verwendeten Desktop-PC-Hardware ein virtueller Server (vServer) in einem Rechenzentrum in Frankfurt getestet. Verwendet wurde eine Maschine mit sechs virtuellen CPUs, einer SSD-Festplatte und 16 GB RAM. Während die Ergebnisse der Overpass API vergleichbar waren, zeigte sich in Hinblick auf die Zeiten der WFS-Operationen ein deutlicher Unterschied. Der Mittelwert war etwa 600 ms geringer und die Standardabweichung lag bei nur 0,5 s anstatt bei fast 1,1 s, wodurch sich im Durchschnitt eine Zeitersparnis von etwa 68 % ergab. Die Ergebnisse ließen sich zu anderen Zeitpunkten mit ähnlichen Resultaten reproduzieren, sodass eventuell die schnellere Netzwerkanbindung des virtuellen Servers ausschlaggebend für diese Verbesserung ist.

<sup>3</sup>Bei der Verwendung eines Desktop-PCs. Die Hardwarespezifikationen können der Fußnote auf Seite 65 entnommen werden.

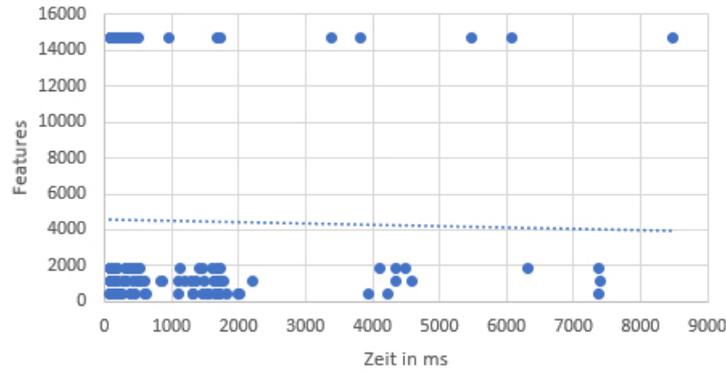


Abbildung 5.2: Abrufzeiten der WFS Daten

Das Parsing konnte wie zu erwarten auf dem performanteren vServer deutlich schneller verarbeitet werden. Der Mittelwert für die Bearbeitungszeit ist um 6 s gesunken, die Standardabweichung um etwa 3 s. Sets mit fast 15.000 Features wurden maximal in unter 9 s verarbeitet, was je nach Anforderung selbst für eine Live-Bearbeitung noch einen akzeptablen Wert darstellt.

Tabelle 5.2: Deskriptive Statistik der Parsingzeiten in ms

Kennzahl	N	Mittelwert	Std.-Abw.	Min	Max
Parsing (WFS) Desktop-PC	200	8.137	4.457	2.223	21.925
Parsing (WFS) vServer	200	2.072	1.340	732	8.793

Die in Kapitel 4.5 vorgestellten VergleichsprozEDUREN für das Rating können in vergleichsweise geringer Zeit berechnet werden. Mit dem Desktop-PC dauerte ein Vergleich von 1.000 Objekten im Durchschnitt 2 s, während der vServer bereits nach nur ca. 600 ms die Berechnungen abgeschlossen hat. Etwas länger dauert auf Grund der geometrischen *Contains*-Funktion der Abruf der Feature-Attribute und die Zuordnung zu den Vergleichsobjekten. Hier benötigt der Desktop-PC etwa 5 s und der vServer lediglich 1,4 s. Diese Zeit hat jedoch nur dann Auswirkungen für die Rating-Bearbeitungszeit des Nutzers, wenn die Objektpositionen nicht per PUT-Request in der API vorgehalten werden. Andernfalls werden bereits beim Speichern der Objekte sämtliche benötigte Kennzahlen abgerufen und zugewiesen. Somit lassen sich für den Nutzer, je nach verwendetem Endpunkt, Bearbeitungszeiten von < 1 s bis 2 s mit einer vergleichbaren Server-Hardware realisieren, um 1.000 Immobilienobjekte anhand ihrer Positionen zu vergleichen.

Ein weiteres wichtiges Ergebnis ist, dass in Hinblick auf Leistungssteigerungen (und somit im Umkehrschluss auf die Nutzbarkeit) bei der Einbindung und Verarbeitung von Open Geo Data Diensten die Verwendung von **Parallelität** in der Implementierung eine wichtige Rolle spielt. Durch das parallele Abrufen und Verarbeiten der externen Datenquellen konnten die Bearbeitungszeiten mehr als halbiert werden. So wird nicht ein Datendienst nach dem anderen abgerufen, sondern die Comparables-Schnittstelle fragt zeitgleich mehrere ab. Auch die im Anschluss stattfindende Verarbeitung der Daten, wie beispielsweise das Parsing der einzelnen Elemente des Ergebnisssets, erfolgt parallel und nicht sequentiell. Durch dieses Vorgehen konnte z. B. die Distanzberechnung zu 10.831 POIs für 1.000 Objekte von  $\varnothing$  2.413 ms um ca. 74 % auf  $\varnothing$  639 ms reduziert werden. Für die technische Umsetzung gibt es hierbei mittlerweile mehrere

---

Möglichkeiten. Seit der Java Version 8 können beispielsweise *Parallel Streams* für diese Parallelität genutzt werden, bei der die Operationen automatisch auf unterschiedliche *Threads* (engl. für Ausführungsstrang) verteilt werden (M. MÜLLER, 2016). Sind mehrere Prozessorkerne verfügbar, können die unterschiedlichen Ausführungsstränge mit sequentiellen Programmfunktionen unter den Kernen aufgeteilt und somit parallel bearbeitet werden (TICKOO, 2008). Das folgende einfache Codebeispiel 5.1 zeigt den simplen Unterschied zwischen den beiden Implementierung<sup>4</sup>. Es soll jedoch nicht unerwähnt bleiben, dass es keinen Sinn macht grundsätzlich Parallel Streams zu verwenden, da bei deren Nutzung der Overhead deutlich höher ist, als bei einer sequentiellen Bearbeitung. So kann es bei manchen Operationen sogar einen Geschwindigkeitsnachteil bringen, wenn die Prozesse parallel ausgeführt werden (M. MÜLLER, 2016).

Code 5.1: Beispiel Parallel Streams

```
1 public static Map<POI, Double> calculatePoiDistances(Position
   referencePosition, List<POI> pois) {
2     Map<POI, Double> poiDistances = new HashMap<>();
3     //Sequentiell
4     pois.forEach(poi -> {
5         double distance = distance(referencePosition, poi.getPosition());
6         poiDistances.put(poi, distance);
7     });
8
9     //Parallel durch Parallel Stream
10    pois.parallelStream().forEach(poi -> {
11        double distance = distance(referencePosition, poi.getPosition());
12        poiDistances.put(poi, distance);
13    });
14    return poiDistances;
15 }
```

Grundsätzlich ist die Performanz der eingebundenen Dienste im Durchschnitt zufriedenstellend, jedoch ist die Spannweite und Varianz der Abfragezeiten recht hoch, wodurch es bei den Tests immer wieder zu langsameren Bearbeitungszeiten kam. Zusammenfassend wird für das Parsing der WFS Ergebnisse sowie für die erwähnten GIS-Analysen am meisten Zeit in Anspruch genommen. Mit performanterer Hardware können diese Verarbeitungszeiten entsprechend reduziert werden, sodass ab diesem Zeitpunkt die Leistungsfähigkeit der Dienste den limitierenden Faktor darstellt. Außerdem waren während der Bearbeitungszeit dieser Thesis die WFS-Dienste der Stadt Berlin von Zeit zu Zeit kurzzeitig gar nicht erreichbar<sup>5</sup> oder so langsam, dass ein Timeout von 60 Sekunden gegriffen hat. Dieser Fall ist bei der Overpass API während der Untersuchungen nicht aufgetreten, wenngleich auch hier die Bearbeitungszeiten nicht durchgehend konstant waren. Da für den vorliegenden Anwendungsfall tagesaktuelle Daten als ausreichend angesehen werden, kann über die ausbaufähige Leistung der Dienste hinweggesehen werden – speziell unter dem

---

<sup>4</sup>In dem gezeigten Beispiel müsste das Resultat der `distance`-Funktion keiner Variable zugeordnet werden, sondern könnte direkt in der `Map` gespeichert werden. Diese Vorgehensweise dient lediglich der besseren Lesbarkeit.

<sup>5</sup>So trat beispielsweise über 48 Stunden bei allen WFS-Diensten des FIS-Brokers der Stadt Berlin bei der Verwendung der `GetFeature`-Operation der Fehler `Error connecting to database. Cause: I/O-Fehler: The Network Adapter could not establish the connection` auf. Der FIS-Broker ist das GIS der Stadt Berlin. FIS steht für „Fachübergreifendes Informationssystem“.

Gesichtspunkt, dass in Berlin für neue Stichtage üblicherweise ein neuer Dienst mit eigenem FeatureType aufgesetzt wird und daher lediglich Korrekturen oder ähnliches über einen erneuten Abruf wiedergespiegelt werden. Im Falle der Overpass API ist eine höher frequentierte Aktualisierung durchaus von Interesse, da sich die Daten stetig verändern und den aktuellen Stand von OSM wiedergeben. Jedoch kommt es auch hier nicht auf minuten- oder sekundengenaue Daten an. Für andere Anwendungsfälle wäre eine weitere Verbesserung und besonders eine gesteigerte Konstanz wünschenswert, da in diesem Fall stetig die aktuellsten Daten abgerufen werden könnten, sofern die Anzahl der einzelnen Anfragen nicht zu hoch wird (siehe beispielsweise die Limitierungen der Overpass API in Kapitel 3.2).

Bei der Arbeit mit den WFS Diensten ergab sich ein kritisches Ergebnis in Hinblick auf die *DescribeFeatureType*-Operationen. Wie bereits in Kapitel 2.6.3 erwähnt, ist es seit der WFS Version 1.1.0 möglich den einzelnen Feldern eine Beschreibung hinzuzufügen. Bei einigen Diensten wird diese Beschreibung auch vorbildlich verwendet, so z. B. bei dem Datensatz „Monitoring Soziale Stadtentwicklung 2017: Kontext-Indikatoren“, der unter anderem den Ausländeranteil zur Verfügung stellt. Jedoch gibt es auch andere Dienste, bei denen neben dem Feldtyp und -namen keinerlei Beschreibungen für die einzelnen Felder vorliegen. Hieraus ergeben sich für den Nutzer deutlich größere Aufwände die richtige Kennzahl zu identifizieren, da die Namen oftmals nicht selbsterklärend sind, der Abstract des *GetCapabilities*-Requests keine hilfreichen Hinweise bietet und beispielsweise auch die Einheit der Ausgabe nicht bekannt ist. Der Anwender muss in diesem Fall weitere beschreibende Webseiten aufsuchen oder den Dienst im FIS-Broker laden, um sich weitere Informationen zu den Feldern einzuholen. Hier besteht zweifelsohne Nachholbedarf, um die Arbeit mit den Diensten einfacher zu gestalten.

Wenngleich die Integration der Open Geo Data Dienste schlussendlich erfolgreich funktionierte, zeigte sich, dass an verschiedenen Stellen Unregelmäßigkeiten auftraten, die zu Fehlermeldungen, Workarounds oder Rechercheaufwand und somit zu einem erhöhten Aufwand beim Nutzer führten. Neben der leistungsbezogenen Nutzbarkeit sollte sowohl die Auffindbarkeit, als auch die Eignungsprüfung sowie die letztendliche Nutzung für den Endanwender einfach und ohne zu großen Aufwand möglich sein, da andernfalls eine sinnhafte Verwendung des Angebots eher unwahrscheinlich ist. Bezüglich der Eignungsprüfung konnte bei den Recherchen zu verfügbaren Lizenzen herausgefunden werden, dass es für die Datenbereitsteller einige Möglichkeiten zur Lizenzierung von Open Geo Data gibt. Deren Kenntnis ist auch für die Nutzer zur korrekten Verwendung der jeweiligen Daten notwendig, um nicht gegen die Nutzungsvereinbarungen zu verstoßen. Im Untersuchungsgebiet trat keine Vielzahl unterschiedlicher Lizenzmodelle auf, sondern es wurde vorbildlich auf eine einheitliche Lizenzierung zurückgegriffen. Im Geoportal Berlins liegen hauptsächlich Datensätze vor, die mit der GeoNutz-BE oder CC BY(-SA) lizenziert sind.

---

# 6 Fazit

## 6.1 Zusammenfassung

Die Literaturanalyse hat ergeben, dass von Open Geo Data bzw. Open Data allgemein ein großes wirtschaftliches Potenzial ausgeht. Auf Grund der gesteigerten Nachfrage nach dem „Rohstoff des 21. Jahrhunderts“ zeichnen sich positive Entwicklungen durch rechtliche Maßnahmen und organisatorische Initiativen ab. Wenngleich beispielsweise die Umsetzung der INSPIRE-Richtlinie noch nicht abgeschlossen ist und es noch immer offene Kritikpunkte bezüglich inkompatiblen Lizenzmodellen oder der mangelnden Verpflichtung/Umsetzung des „open by default“-Ansatzes gibt, sorgen aktuelle Diskussionen für eine Auseinandersetzung mit einigen der problematischen Punkte. Die Möglichkeiten für Ausnahmen mit einschränkenden Auswirkungen, beispielsweise dem Verlangen von Gebühren, werden ebenso reduziert, wie die potenziell möglichen Kosten und Hindernisse. Die Forderung nach offenen Standards und Formaten schlägt die Richtung zu Open Data 3.0 ein. Die wachsende Verbreitung ermöglicht es interoperable Fragestellungen zu realisieren. Hierbei ist grundsätzlich eine Bewegung weg von reinen downloadbaren Datensätzen hin zu miteinander verknüpfbaren Diensten erkennbar. Dieser Ansatz ist nicht zuletzt in der Neufassung der PSI-Richtlinie erkennbar.

Neben der Open Geo Data Situation wurde mithilfe der Literaturrecherche ebenso untersucht, welche wertbeeinflussenden Standortfaktoren für eine Immobilie relevant sind. Die sich in Deutschland in Verwendung befindenden Richtlinien für die Immobilienwertermittlung (ImmoWertV und BelWertV) nennen die Lage jeweils als einen wichtigen Faktor, der bei dem Vergleichsverfahren maßgeblich zwischen Referenz- und Vergleichsobjekt übereinstimmen sollte. Eine konkretere Definition, welche Indikatoren für die Lageeigenschaften als relevant angesehen werden, erfolgt jedoch nicht. Grundsätzlich wurden lagerelevante Indikatoren, die den Preis einer Immobilie bestimmen, bereits in anderen Arbeiten untersucht und deren Ergebnisse dienen somit als Grundlage dafür, welche Dienste innerhalb der entwickelten Schnittstelle für einen Vergleich Verwendung finden. Dabei haben beispielsweise die Distanz zu Bildungseinrichtungen und Erholungs- bzw. Grünflächen, die Lärm- und Feinstaubbelastung sowie die Zentralität eine entscheidende Bedeutung für die Attraktivität einer Immobilie, die sich in ihrem Preis widerspiegelt.

Eine weitere Zielsetzung der Arbeit war es, zu untersuchen, inwiefern die Open Geo Data Dienste für den Anwendungsfall der Vergleichsobjektanalyse im Untersuchungsgebiet Berlin ein ausreichendes Datenangebot bieten und ob sie performant genug sind, um dynamische Zugriffe für die Prozesse zu realisieren. Hierfür wurde für die ausgewählten Themenbereiche nach Geodaten-Diensten gesucht, um diese im Anschluss in einer Web-Schnittstelle mithilfe der existierenden OGC-Standards wie WFS einzubinden. Es stellte sich heraus, dass die Einbindung der Dienste und die Verschneidung der gelieferten Daten zu großen Teilen performant realisierbar sind. Die einheitlichen Nutzungsrichtlinien wirken sich positiv auf die zeitliche Effizienz bei der Eignungsprüfung aus und die durchschnittlichen Abfragezeiten sind größtenteils zufriedenstellend. Nichtsdestotrotz gab es teilweise deutliche Varianzen und mitunter temporäre Ausfälle, die in

Einzelfällen über 48 Stunden am Wochenende andauerten. Nicht zuletzt aus diesem Grund wurde sich für eine lediglich tägliche Aktualisierung der Daten entschieden. Für den Anwendungsfall werden tagesaktuelle Informationen als ausreichend angesehen und die Performanz kann für den Endnutzer so deutlich erhöht werden. Bei der Einbindung der Overpass API zeigte sich ein konstantes Bild in Bezug auf die Bearbeitungszeiten am Server. Ein weiterer Kritikpunkt ergab sich durch fehlende Feldbeschreibungen bei den *DescribeFeatureType*-Operationen einiger Dienste. Nur zu Teilen wurden die Möglichkeiten der Beschreibung genutzt, sodass in den anderen Fällen die Nutzer einen größeren Rechercheaufwand in Kauf nehmen müssen, um alle benötigten Informationen für eine korrekte Einbindung zu erhalten.

Neben der Betrachtung der offenen Geodatendienste wurde bei der Entwicklung der Comparableschnittstelle festgestellt, dass für die Verarbeitung der bereitgestellten Informationen die Verwendung von parallelen Arbeitssträngen maßgeblich für die Realisierung akzeptabler Bearbeitungsdauern ist. Durch die Verwendung von Parallelität konnten die Zeiten der GIS-Analysen und der vergleichenden Berechnungen zum Teil über 70% reduziert werden.

## 6.2 Diskussion

Mit einer steigenden Anzahl von integrierten Diensten besteht ein deutlich erhöhter Pflegeaufwand – speziell dann, wenn nicht nur ein Dienst mit mehreren FeatureTypes verwendet wird, um beispielsweise Daten mehrerer Jahre abzubilden, sondern jeweils ein Dienst pro Jahr. In Hinblick auf diesen Sachverhalt könnte es von Interesse sein, verstärkt multi-dimensionale Datenformate zu verwenden. GML unterstützt in den neueren Versionen beispielsweise die Möglichkeit, die Dimension der Zeit abzubilden (OPEN GEOSPATIAL CONSORTIUM, 2019b). Es wäre wünschenswert, wenn verstärkt auf Möglichkeiten zur Betrachtung von Historien gesetzt werden würde, um die Entwicklung eines Features über mehrere Jahre betrachten zu können. Das mehrdimensionale und offene Datenformat *NetCDF* (Network Common Data Form) zur Speicherung von wissenschaftlichen Daten ermöglicht es den Anwendern beispielsweise Daten über ein Gebiet, unter Berücksichtigung der Höhe und der Zeit, zu betrachten.

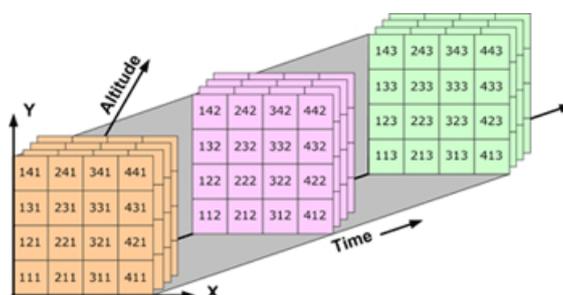


Abbildung 6.1: 4D-Daten im Format NetCDF – Daten unter Berücksichtigung der Lage und Zeit (ENVIRONMENTAL SYSTEMS RESEARCH INSTITUTE (ESRI), 2016)

NetCDF ist ein internationaler Standard des OGC und wird mittlerweile beispielsweise von der Open Source Software GeoServer unterstützt, welches die Implementierung von unter anderem WFS und WMS ermöglicht. Somit können NetCDF-Datensätze in der Anwendung importiert werden und als WFS-Record in dem Format GML zur Verfügung gestellt werden, wobei die Zeit

als eine Dimension enthalten ist. Doch losgelöst von einem speziellen Format oder einer konkreten Implementierung, würde eine breitflächigere Berücksichtigung des Aspekts Zeit bzw. Werthistorie die Arbeit mit den Diensten weiter verbessern, da lediglich ein Dienst für die Betrachtung der Wertentwicklung eines Datensatzes verwendet werden könnte.

Ein weiteres Problem, welches bei der Pflege von verwendeten Diensten auftreten kann, ist der Sachverhalt, dass sich FeatureTypes und Attribute über die Zeit serverseitig ändern können, ohne dass die Clients darüber informiert werden bzw. wenn sie informiert werden, dies meist über eine Neuigkeit auf dem zuständigen Geoportal oder mithilfe eines Zeitstempels auf der Informationsseite des jeweiligen Datensatzes veröffentlicht wird. Diese Webseiten werden jedoch, speziell bei der Verwendung einer Vielzahl von Datensätzen und Diensten, meist nicht täglich von den Nutzern dahingehend untersucht<sup>1</sup>. Während der Bearbeitung der Master Thesis und bereits durch vergangenen Erfahrungen zeigte sich, dass die URLs von Diensten geändert werden, ohne die alte URL mit einer Weiterleitung (siehe Kapitel 4.1.3) beizubehalten. Außerdem ist es möglich, dass es Attribute oder FeatureTypes über die Zeit umbenannt werden. Ein Fakt der über Operationen wie *GetCapabilities* überprüfbar, aber ohne komplexere Aufwände schwer bis gar nicht automatisierbar zu korrigieren ist – es Bedarf manueller Anpassungen. Für diesen Fall wäre eine Versionierung der Dienste wünschenswert, bei der unterschiedliche Versionen verschiedene Outputs generieren könnten. Für eine Übergangszeit könnte noch mit der alten Version gearbeitet werden (mit dem maschinenlesbaren Hinweis, dass diese Version veraltet ist und durch eine neue Version ersetzt wird). Weiterführend könnten definierte Informationen, wie Umbenennungen von FeatureTypes, standardisiert kommuniziert werden, sodass bis zu einem gewissen Änderungsgrad eine automatisierte Aktualisierung auf die neuere Version erfolgen könnte. Eine mögliche Operation könnte *GetVersionInformation* heißen, die neben der Information, ob es sich bei der verwendeten Version um eine veraltete handelt, zusätzlich die Änderungen ähnlich der JSON Patches<sup>2</sup> übermittelt:

Code 6.1: Mögliche Änderungshistorie im JSON Format

```
1 {
2   "version": "1.0",
3   "deprecated": true,
4   "mostRecentVersion": "2.0",
5   "patches": [
6     {
7       "op": "replace",
8       "path": "/featureTypes/1/name",
9       "value": "myRenamedFeatureType"
10    },
11    {
12      "op": "remove",
13      "path": "/featureTypes/5"
14    }
15  ]
}
```

<sup>1</sup>Das Open Data Portal Berlins bietet einen *Really Simple Syndication* (RSS)-Feed an. Dieser Dienst kann abonniert werden, um über neu veröffentlichte Datensätze informiert zu werden.

<sup>2</sup>Mithilfe von JSON Patches können Daten im JSON Format manipuliert werden (siehe BRYAN und NOTTINGHAM (2013)).

Bei den eingebundenen Diensten der Stadt Berlin wird für die Lizenzierung die GeoNutz-BE verwendet. Diese fordert einen Quellenvermerk bei der Nutzung der Daten. In den *GetCapabilities*-Dokumenten der WFS-Dienste wird unter dem Tag *Fees* (engl. für Gebühren) auch der Hinweis „Nutzungsbedingungen: Für die Verwendung der Daten gelten folgende Nutzungsbestimmungen: <http://www.stadtentwicklung.berlin.de/geoinformation/download/nutzIII.pdf> - Der Quellenvermerk gemäß §2 lautet "Geoportal Berlin / [Titel des Datensatzes]" gegeben. Jedoch lässt sich dieser Quellenvermerk im Gegensatz zu beispielsweise den Kontaktdetails des Diensteanbieters erschwert automatisiert auslesen. Der Titel des Datensatzes ist entweder über die Liste der FeatureTypes oder direkt zu Beginn in dem dafür vorgesehenen Tag festzustellen. Anschließend müsste der Beginn des Hinweises entfernt und der Platzhalter mit dem Titel ersetzt werden. Bei Änderungen der Formulierungen müssten codeseitige Anpassungen erfolgen, um die korrekte Quellenangabe nach den Nutzungsbedingungen zu gewährleisten – der gleiche Aufwand gilt für die manuelle Einpflege des Vermerks. Dieser Sachverhalt lässt sich auch bei anderen freien Angeboten feststellen, bei denen beispielsweise der Quellenvermerk nur im Geoportal vermerkt ist, aber nicht im entsprechenden Tag der Metainformationen oder aber in einem Fließtext der Beschreibung. Hierdurch erhält das Thema Lizenzierung bei der Eignungsprüfung und der Integration in automatisierte Prozesse eine größere Komplexität als notwendig.

Ferner wurde herausgefunden, dass in Berlin zu großen Teilen die Lizenzen GeoNutz-BE sowie CC-BY(-SA) Verwendung finden. Jedoch ändert sich die Situation schnell, wenn Dienste anderer Bundesländer betrachtet werden – es ergibt sich ein heterogenes Bild an Lizenzen (SEUSS, 2015). Um den Aufwand speziell auf der Nutzerseite möglichst gering zu halten, wäre hier eine Vereinheitlichung wünschenswert, die jedoch voraussichtlich einige Zeit und viele Diskussionen unter den beteiligten Akteuren in Anspruch nehmen wird.

In Hinblick auf die Bezugsjahre wäre es wünschenswert eine dichtere Datenlage vorzufinden, um in der Lage zu sein, Stichtage basierend auf dem jeweiligen Jahr besser berücksichtigen zu können. Hier ergeben sich auf Grund der aktuellen Datenlage Unsicherheiten. Auch wenn für manche Datensätze eine jährliche Fortschreibung auf Grund geringer Veränderungen oder zur Kostenminimierung nicht durchgeführt wird, wäre es wünschenswert von der datenbereitstellenden Stelle zu erfahren, ob die vorliegenden Daten eventuell auch für andere Jahre zu Großteilen zutreffend sind und mit geringen Abweichungen in der Datenqualität verwendet werden können.

### 6.3 Ausblick und Ansätze für weitere Forschungen

Die vorliegende Arbeit hat ausschließlich 2D-Daten und keine dreidimensionalen Modelle berücksichtigt. Da jedoch Faktoren, wie z. B. die Aussicht auf andere Gebäude oder weite Landschaften, eine entscheidende Rolle für die Attraktivität einer Immobilie spielen, wäre die zusätzliche Berücksichtigung von 3D-Daten sehr interessant. HÜRZELER (2012) hat in seiner Masterarbeit mit dem Titel „Der Preis der Aussicht“ 3D-Stadtmodelle und Sichtbarkeitsanalysen in Hinblick auf ein hedonisches Preismodell untersucht. Der Einbezug solcher Sichtanalysen in den Vergleich von Lageparametern könnte eine interessante weitere Forschung darstellen und bei der Ermittlung geeigneter Vergleichsobjekte eine weitere Qualitätssteigerung herbeiführen. Auf Grund der benötigten Ressourcen müsste eine solche Analyse jedoch im Vorfeld berechnet und nicht in

Echtzeit pro Objekt kalkuliert werden. Die Ergebnisse könnten ebenso über einen Webdienst abrufbar gemacht werden.

Wie bereits in vorherigen Kapiteln angeschnitten, wurde die vergleichende Analyse für objektbezogene Primärdaten auf Grund des Ausmaßes in dieser Thesis nicht berücksichtigt. Grundsätzlich ist davon auszugehen, dass Anwender individuelle Daten miteinander vergleichen möchten, die mit einem Objekt verknüpft sind. So kann es sein, dass ein Nutzer keine Primärdaten für einen detaillierten Vergleich vorliegen hat und sich in einem ersten Schritt hauptsächlich auf externe und offene Daten stützen möchte, während ein anderer Nutzer pro Objekt 50 Kennzahlen erfasst hat, die zusätzlich miteinander verglichen werden können. In diesem Hinblick wäre eine Weiterentwicklung der API bzw. die Entwicklung eines ergänzenden Services eine interessante Folgeaufgabe. Hierbei könnte beispielsweise untersucht werden, wie dynamisch übermittelte und heterogene Daten an Hand ihrer Skalenniveaus bestmöglich im Zuge des Vergleichsverfahrens verglichen werden können und wie der Anwender bestimmte Regeln gegebenenfalls eigenständig über die Schnittstellen beeinflussen kann. Hierdurch könnten interne und externe Daten zeitgleich verarbeitet und miteinander verschnitten werden. Eine zusätzliche vor- oder nachgelagerte Filterung der Objekte anhand intern vorliegender Kennzahlen könnte somit vermieden werden.

Eine Übertragung der in dieser Arbeit durchgeführten Untersuchungen auf das Bundesgebiet unter Berücksichtigung weiterer Dienste stellt eine spannende Folgeaufgabe dar. Hierdurch ergeben sich neue Herausforderungen, da die Datenlage aufgrund des föderalen Staatsaufbaus im Bundesgebiet sehr unterschiedlich ist und andere Dienste mit gegebenenfalls anderen Klassifizierungen für eine Analyse zur Verfügung stehen. Speziell in Gebieten entlang der Landesgrenzen ergeben sich somit komplexe Fragestellungen, da die Methodiken bei der Datenerhebung unterschiedlich sein können. Hieraus resultiert, dass sich Datenlagen und beispielsweise Skalen von Ratings stark unterscheiden können, obwohl die Gebiete eigentlich starke Ähnlichkeiten aufweisen. Ob und inwiefern hier eine Transformation stattfinden kann, müsste für den jeweiligen Einzelfall untersucht werden. Nicht zuletzt aus diesem Grund kann bei Immobilienmarkt bezogenen Vorträgen auf Konferenzen wie der Intergeo verstärkt beobachtet werden, dass Nutzer der Open Data bundesweite Standards fordern anstatt Insellösungen pro Bundesland. Da eine solche Umstellung jedoch einen großen Ressourcenaufwand bedarf und die Kommunikationswege zwischen den Beteiligten zum Teil noch nicht gegeben sind, wird es voraussichtlich noch einige Jahre dauern, bis eine solche Struktur geschaffen ist. Zwar ist die bisherige Orientierung an politischen Grenzen, speziell in Hinblick auf die Verwendung von statistischen Daten, die einfachere Variante, dennoch wäre beispielsweise die Nutzung von wirtschaftlichen Standortabgrenzungen, wie dem Rhein-Main-Gebiet, aus Sicht der Immobilienwirtschaft die sinnvollere Lösung (FELDMANN et al., 2016).

In der Immobilienbranche zeigen sich aktuell unterschiedlichste Entwicklungen in Hinblick auf eine voranschreitende Digitalisierung. Es bleibt interessant, wie sich die Rolle eines Immobiliengutachters im Laufe dieser fortschreitenden Entwicklung verändern wird und welche Arbeitsschritte durch die standardisierte Bereitstellung relevanter Geodaten automatisiert werden (können). Es stellt sich die Frage inwiefern sich Prozesse zukünftig in Hinblick auf die Fakten Zeitersparnis und Wertgenauigkeit optimieren lassen. Trotz der positiven Entwicklungen bezogen auf Open Data und Open Government, gibt es noch immer einige offene Punkte, die in den kommenden Jahren für die Phase Open Data 3.0 angegangen werden müssen. Der Trend hin zu miteinander verbun-

denen Diensten, basierend auf frei verfügbaren Daten mit offenen Standards, ist unverkennbar und stärkt die Hoffnung, dass in Zukunft noch performantere und nutzenbringendere Projekte mit standardisierten Datenmodellen umgesetzt werden können, als dies aktuell der Fall ist. Seitens der daten- und dienstbereitstellenden Institutionen sind Verbesserungen bezüglich einer Vereinheitlichung ebenso wie bei der konstanten Performanz, den lizenzrechtlichen Bestimmungen und der Datenlage gefordert, um dieses Ziel zu erreichen.

## Literatur

- ARENS, JENNY (2016): „Unterscheidung nach Immobilienarten“. In: *Immobilienökonomie. Band I. Betriebswirtschaftliche Grundlagen*. Hrsg. von SCHULTE, KARL-WERNER; BONE-WINKEL, STEPHAN und SCHÄFERS, WOLFGANG. 5. grundlegend überarbeitete Auflage. Berlin: Walter de Gruyter GmbH, S. 83–108.
- BAFU (BUNDESAMT FÜR UMWELT) (2009): Lärmbelastung in der Schweiz. Ergebnisse des nationalen Lärmmonitorings SonBase. Umwelt-Zustand Nr. 0907.
- BALLATORE, ANDREA und MOONEY, PETER (2015): „Conceptualising the geographic world: the dimensions of negotiation in crowdsourced cartography“. In: *International Journal of Geographical Information Science* 29.12, S. 2310–2327.
- BARBAGLIA, GUIDO; MURZILLI, SIMONE und CUDINI, STEFANO (Dez. 2016): „Definition of REST web services with JSON schema“. In: *Software: Practice and Experience* 47.6, S. 907–920.
- BEHRENS, LARS; IBENDORF, JENS und WÜRRIEHAUSEN, FALK (2017): Nutzungsregelungen in der GDI-DE. Aufgerufen: 15.01.2019. URL: [https://www.geodaten.niedersachsen.de/download/124230/Nutzungsregelungen\\_in\\_der\\_GDI-DE.pdf](https://www.geodaten.niedersachsen.de/download/124230/Nutzungsregelungen_in_der_GDI-DE.pdf).
- BERLIN HYP & CBRE (2019): Wohnmarktreport Berlin 2019. Aufgerufen: 27.03.2019. URL: <https://www.cbre.de/de-de/research/cbre-berlin-hyp-wohnmarktreport-berlin-2019>.
- BERNARD, LARS; FITZKE, JENS und WAGNER, ROLAND M. (2004): Geodateninfrastruktur – Grundlagen und Anwendungen. Heidelberg: Wichmann.
- BILL, RALF (Mai 2014): „Geoinformation für jedermann – goldene Zeiten für die empirische Raumforschung“. In: *Raumforschung und Raumordnung* 72.3, S. 177–178.
- BLÜML, ANDREAS (2014): Immobilienwirtschaftliche Investmentstile – Eine theoretische und empirische Untersuchung am Beispiel der Präferenzstrukturen institutioneller Immobilieninvestoren. Regensburg: Universitätsbibliothek Regensburg.
- BONE-WINKEL, STEPHAN; FOCKE, CHRISTIAN und SCHULTE, KARL-WERNER (2016): „Begriff und Besonderheiten der Immobilie als Wirtschaftsgut“. In: *Immobilienökonomie. Band I, Betriebswirtschaftliche Grundlagen*. Hrsg. von SCHULTE, KARL-WERNER; BONE-WINKEL, STEPHAN und SCHÄFERS, WOLFGANG. 5. grundlegend überarbeitete Auflage. Berlin: Walter de Gruyter GmbH, S. 3–24.
- BRAY, TIM; PAOLI, JEAN und SPERBERG-MCQUEEN, C. MICHAEL (1998): Extensible Markup Language (XML) 1.0. Aufgerufen: 06.01.2019. URL: <https://web.archive.org/web/20060615212726/http://www.w3.org/TR/1998/REC-xml-19980210>.
- BRYAN, PAUL C. und NOTTINGHAM, MARK (2013): JavaScript Object Notation (JSON) Patch. Aufgerufen: 21.02.2019. URL: <https://tools.ietf.org/html/rfc6902>.
- BULWIENGESA AG (2018): Wohnungsmarktstudie Berlin. Aufgerufen: 10.01.2019. URL: [https://www.bulwiengesa.de/sites/default/files/bfw\\_wohnungsmarkt-studie\\_20180919.pdf](https://www.bulwiengesa.de/sites/default/files/bfw_wohnungsmarkt-studie_20180919.pdf).
- BUNDESAMT FÜR KARTOGRAPHIE UND GEODÄSIE (2017): Geodatenkatalog.de. Aufgerufen: 22.02.2019. URL: <https://www.geoportal.de/DE/GDI-DE/Komponenten/Geodatenkatalog-DE/geodatenkatalog-de.html?lang=de>.

- BUNDESGERICHTSHOF (o.D.): Urteil des V. Zivilsenats vom 12.1.2001 - V ZR 420/99. Aufgerufen: 12.01.2019. URL: <http://juris.bundesgerichtshof.de/cgi-bin/rechtsprechung/document.py?Gericht=bgh&Art=en&nr=24194&pos=0&anz=1>.
- BUNDESMINISTERIUM DES INNEREN (2017): Geoinformationen einfach genutzt. Vierter Bericht der Bundesregierung über die Fortschritte zur Entwicklung der verschiedenen Felder des Geoinformationswesens im nationalen, europäischen und internationalen Kontext. 4. Geofortschrittsbericht der Bundesregierung. Aufgerufen: 20.02.2019. URL: [https://www.bmi.bund.de/SharedDocs/Downloads/DE/Nachrichten/Pressemitteilungen/2017/06/geofortschrittsbericht.pdf?\\_\\_blob=publicationFile](https://www.bmi.bund.de/SharedDocs/Downloads/DE/Nachrichten/Pressemitteilungen/2017/06/geofortschrittsbericht.pdf?__blob=publicationFile).
- BUNDESMINISTERIUM FÜR UMWELT, NATURSCHUTZ, BAU UND REAKTORSICHERHEIT (2014): Bekanntmachung - Richtlinie zur Ermittlung des Vergleichswerts und des Bodenwerts (Vergleichswertrichtlinie – VW-RL).
- BUNDESMINISTERIUMS DER JUSTIZ UND FÜR VERBRAUCHERSCHUTZ (2009): Beleihungswertermittlungsverordnung vom 12. Mai 2006 (BGBl. I S. 1175), die durch Artikel 1 der Verordnung vom 16. September 2009 (BGBl. I S. 3041) geändert worden ist. Stand: Geändert durch Art. 1 V v. 16.9.2009 I 3041. Aufgerufen: 07.02.2019. URL: <https://www.gesetze-im-internet.de/belwertv/BJNR117500006.html>.
- BURGDORF, MARKUS; KRISCHAUSKY, GESINE und MÜLLER-KLEISSLER, RENATE (2015): „Indikatoren zur Nahversorgung“. In: *BBSR-Analysen KOMPAKT* 10/2015. Aufgerufen: 25.03.2019, S. 1–20.
- BUTLER, HOWARD et al. (2016): RFC-7946: The GeoJSON Format. Aufgerufen: 07.01.2019. URL: <https://tools.ietf.org/html/rfc7946>.
- CLOUD ELEMENTS (2017): The State of the API Integration Report 2017. Forschungsber.
- COPERNICUS.EU (2018): Copernicus In Brief. Aufgerufen: 01.02.2019. URL: <https://www.copernicus.eu/en/about-copernicus/copernicus-brief>.
- CPANEL (2018): Git Version Control series: What is Git? Aufgerufen: 02.02.2019. URL: <https://blog.cpanel.com/git-version-control-series-what-is-git/>.
- CREATIVE COMMONS (2018): Data. Aufgerufen: 12.02.2019. URL: <https://creativecommons.org/publicdomain/zero/1.0/deed.de>.
- CREATIVE COMMONS DEUTSCHLAND (2018a): CC0 1.0 Universell (CC0 1.0) Public Domain Dedication. Aufgerufen: 12.02.2019. URL: [https://wiki.creativecommons.org/wiki/Data#What\\_is\\_the\\_difference\\_between\\_the\\_Open\\_Data\\_Commons\\_licenses\\_and\\_the\\_CC\\_4.0\\_licenses.3F](https://wiki.creativecommons.org/wiki/Data#What_is_the_difference_between_the_Open_Data_Commons_licenses_and_the_CC_4.0_licenses.3F).
- (2018b): Was ist CC? Aufgerufen: 13.02.2019. URL: <https://de.creativecommons.org/index.php/was-ist-cc/>.
- DE LANGE, NORBERT (2013): Geoinformatik in Theorie und Praxis. 3. Auflage. Berlin, Heidelberg, New-York: Springer.
- DIETRICH, DANIEL (2013): Offene Lizenzen für Daten und Dokumente der Deutschen Verwaltung. Aufgerufen: 27.02.2019. URL: <https://okfn.de/blog/2013/05/offene-lizenzen-fuer-daten-und-dokumente-der-deutschen-verwaltung/>.
- DIN, ALLAN; HOESLI, MARTIN und BENDER, ANDR (2001): „Environmental Variables and Real Estate Prices“. In: *Urban Studies* 38.11, S. 1989–2000.

- ECMA INTERNATIONAL (2017): Standard ECMA-404. The JSON Data Interchange Syntax. 2. Auflage. Aufgerufen: 15.02.2019. URL: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>.
- ENVIRONMENTAL SYSTEMS RESEARCH INSTITUTE (ESRI) (2016): Grundlagen der netCDF-Datenspeicherung. Aufgerufen: 20.02.2019. URL: <http://desktop.arcgis.com/de/arcmap/10.3/manage-data/netcdf/fundamentals-of-netcdf-data-storage.htm>.
- EUROPÄISCHE KOMMISSION (2003): What changes does the revised PSI Directive bring? Aufgerufen: 10.03.2019. URL: <https://ec.europa.eu/digital-single-market/en/news/what-changes-does-revised-psi-directive-bring>.
- (2014): „Guidelines on recommended standard licences, datasets and charging for the reuse of documents“. In: *Official Journal of the European Union*.
  - (2017): Re-using Open Data. A study on companies transforming Open Data into economic & societal value. Aufgerufen: 05.01.2019. URL: [https://www.europeandataportal.eu/sites/default/files/re-using\\_open\\_data.pdf](https://www.europeandataportal.eu/sites/default/files/re-using_open_data.pdf).
  - (2019): Proposal for a revision of the Public Sector Information (PSI) Directive. Aufgerufen: 10.03.2019. URL: <https://ec.europa.eu/digital-single-market/en/proposal-revision-public-sector-information-psi-directive>.
- EUROPÄISCHES PARLAMENT UND RAT (2007): Richtlinie 2007/2/EG des Europäischen Parlaments und des Rates vom 14. März 2007 zur Schaffung einer Geodateninfrastruktur in der Europäischen Gemeinschaft (INSPIRE). Aufgerufen: 12.02.2019. URL: <http://eur-lex.europa.eu/JOHtml.do?uri=OJ:L:2007:108:SOM:DE:HTML>.
- FALLENBÖCK, MARKUS; GALLA, FRANZ und STOCKINGER, STEFAN (2005): Urheberrecht in der digitalen Wirtschaft. Wien: Manzsche Verlags- und Universitätsbuchhandlung.
- FELDMANN, PHILIPP et al. (2016): „Immobilienanalyse“. In: *Immobilienökonomie. Band I. Betriebswirtschaftliche Grundlagen*. Hrsg. von SCHULTE, KARL-WERNER; BONE-WINKEL, STEPHAN und SCHÄFERS, WOLFGANG. 5. grundlegend überarbeitete Auflage. Berlin: Walter de Gruyter GmbH, S. 363–424.
- FIELDING, ROY et al. (1999): RFC-2616: Hypertext Transfer Protocol – HTTP/1.1. URL: <https://www.w3.org/Protocols/rfc2616/rfc2616.html>.
- FRIEDL, JEFFREY E. F. (2007): Reguläre Ausdrücke. 3. Auflage. Köln: O'Reilly.
- FRIEDT, MICHAEL und LUCKHARDT, THOMAS (2014): „Open Data: Zukunftsorientierte Bereitstellung von amtlichen Geodaten im Land Berlin“. In: *Zeitschrift für Geodäsie, Geoinformation und Landmanagement* 139.5, S. 269–277.
- GEOINFORMATION UND LANDENTWICKLUNG NIEDERSACHSEN, LANDESAMT FÜR (2013): Verwenden von lokalen Geodaten oder Benutzen von Geodaten aus OGC-Diensten? Aufgerufen: 02.03.2019. URL: <https://www.geodaten.niedersachsen.de/download/70286>.
- GLAUSER, THOMAS et al. (Feb. 2009): Was wohnen in Zürich kostet. Das hedonische Mietpreismodell erlaubt differenzierte Aussagen.
- GONDRING, HANSPETER (2012): Zukunft der Immobilie: Megatrends des 21. Jahrhunderts – Auswirkungen auf die Immobilienwirtschaft; Handbuch für Studium und Praxis. Immobilien-Manager-Verlag.
- GROOT, RICHARD und McLAUGHLIN, JOHN (2000): Geospatial Data Infrastructure: Concepts, Cases, and Good Practice. New York: Oxford University Press. 320 S.

- GURIN, JOEL (2014): *Open Data Now: The Secret to Hot Startups, Smart Investing, Savvy Marketing, and Fast Innovation*. Madison: McGraw Hill Professional. 272 S.
- HAASE, RONNY (2011): „Ertragspotenziale: Hedonische Mietpreismodellierungen am Beispiel von Büroimmobilien“. Diss. ETH Zürich.
- HAMMEL, RICHARD et al. (2012): *Open data. Driving growth, ingenuity and innovation*. London: Deloitte LPP.
- HENCKEL, DIETRICH et al. (15. März 2010): *Planen – Bauen – Umwelt – Ein Handbuch*. Wiesbaden: VS Verlag für Sozialwissenschaften. 601 S. URL: [https://www.ebook.de/de/product/8778084/planen\\_bauen\\_umwelt.html](https://www.ebook.de/de/product/8778084/planen_bauen_umwelt.html).
- HENDRIX, STEFAN (2011): *Vergleich der Wertermittlung nach ImmoWertV und BelWertV: Umfangreicher Vergleich mit Praxisbeispielen (Beiträge zur Wirtschaftswissenschaft)*. München: Akademische Verlagsgemeinschaft München.
- HÜRZELER, CHRISTIAN (2012): „Der Preis der Aussicht. Einsatz von 3D-Stadtmodellen und 3D-Sichtbarkeitsanalysen in einem hedonischen Mietpreismodell“. Master Thesis. Paris Lodron-Universität Salzburg.
- ILLERT, ANDREAS (2012): „Ziel erkannt. Ziel erreicht? Eine Bestandsaufnahme zu INSPIRE“. In: *Geoinformationssysteme, Beiträge zum 17. Münchner Fortbildungsseminar*, S. 41–48.
- INTERNATIONAL ASSOCIATION OF OIL & GAS PRODUCERS (2019): *About the EPSG Dataset*. Aufgerufen: 15.03.2019. URL: <http://www.epsg.org>.
- JANSSEN, KATLEEN und CROMPVOETS, JOEP (2013): *Geographic Data and the Law: Defining New Challenges*. New York: Leuven University Press. 206 S.
- JANSSEN, MARIJN; CHARALABIDIS, YANNIS und ZUIDERWIJK, ANNEKE (2012): „Benefits, Adoption Barriers and Myths of Open Data and Open Government“. In: *Information System Management* 29.4, S. 258–268.
- KALASEK, ROBERT und WENINGER, KURT (2015): „Open GeoData“. In: *Der öffentliche Sektor* 41.1, S. 17–28.
- KINDLER, ANNEGRET und FRANCK, ULRICH (2015): „Kernindikator 2: Sozialräumliche Verteilung der Luftbelastung in Berlin.“ In: *Umweltgerechtigkeit im Land Berlin - Grundlagen für die handlungsorientierte sozialräumliche Umweltpolitik in Berlin*.
- KOORDINIERUNGSSTELLE GDI-DE IM BUNDESAMT FÜR KARTOGRAPHIE UND GEODÄSIE (2018): *INSPIRE Monitoring-DE 2017*. Aufgerufen: 27.02.2019. URL: <http://www.gdi-de.org/monitoring2017/>.
- KOWARSCHICK, WOLFGANG (2017): *Model-View-Controller-Service-Paradigma*. Aufgerufen: 04.03.2019. URL: <https://glossar.hs-augsburg.de/Model-View-Controller-Service-Paradigma>.
- KRALIDIS, ATHANASIOS TOM (2008): „Geospatial Open Source and Open Standards Convergences“. In: *Open Source Approaches in Spatial Data Handling*. Hrsg. von HALL, BRENT und LEAHY, MICHAEL G. Berlin, Heidelberg: Springer, S. 1–20.
- KUMARI, SMITA und RATH, SANTANU KUMAR (Aug. 2015): „Performance comparison of SOAP and REST based Web Services for Enterprise Application Integration“. In: *2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, S. 1656–1660.

- KUZEV, PENCHO (2016): Open Data. The Benefits. Das volkswirtschaftliche Potential für Deutschland.
- LEOPOLDSBERGER, GERRIT; THOMAS, MATTHIAS und NAUBEREIT, PHILIPP (2016): „Immobilienbewertung“. In: *Immobilienökonomie. Band I. Betriebswirtschaftliche Grundlagen*. Hrsg. von SCHULTE, KARL-WERNER; BONE-WINKEL, STEPHAN und SCHÄFERS, WOLFGANG. 5. grundlegend überarbeitete Auflage. Berlin: Walter de Gruyter GmbH, S. 425–480.
- LOELIGER, JON (2012): Versionskontrolle mit Git. Projekte effizient und flexibel verwalten. Sebastopol: O’Reilly.
- MASSÉ, MARK (11. Nov. 2011): REST API Design Rulebook. Designing Consistent RESTful Web Service Interfaces. Sebastopol: O’Reilly. 114 S. URL: [https://www.ebook.de/de/product/16725812/mark\\_masse\\_rest\\_api\\_design\\_rulebook.html](https://www.ebook.de/de/product/16725812/mark_masse_rest_api_design_rulebook.html).
- MCKINSEY GLOBAL INSTITUTE (2011): Big data: The next frontier for innovation, competition, and productivity. Aufgerufen: 22.02.2019. URL: [https://www.mckinsey.com/~media/McKinsey/Business%20Functions/McKinsey%20Digital/Our%20Insights/Big%20data%20The%20next%20frontier%20for%20innovation/MGI\\_big\\_data\\_full\\_report.aspx](https://www.mckinsey.com/~media/McKinsey/Business%20Functions/McKinsey%20Digital/Our%20Insights/Big%20data%20The%20next%20frontier%20for%20innovation/MGI_big_data_full_report.aspx).
- MICHAEL, CHRISTOPHER und AMES, DANIEL P. (Mai 2007): „Evaluation of the OGC Web Processing Service for Use in a Client-Side GIS“. In: *OSGeo Journal* 1, S. 1–9.
- MILES, RUSS und HAMILTON, KIM (2006): Learning UML 2.0 – A Pragmatic Introduction to UML. Sebastopol: O’Reilly.
- MÜLLER, MICHAEL (2016): Java Lambdas and Parallel Streams. New York: Apress.
- MÜLLER, VIKTOR-HERMANN (2009): „Die Bewertung von Immobilien: Verfahrensweisen – quantitative und qualitative Methoden“. In: *Kapitalanlagen mit Immobilien – Produkte – Märkte – Strategien zitieren*. Hrsg. von BRUNNER, MARLIES. Wiesbaden: Gabler, S. 271–293.
- MÜRI LEUPP, RUTH et al. (2011): „Ruhe bitte! Wie Lage und Umweltqualität die Schweizer Mieten bestimmen.“ In: *Zürich: Zürcher Kantonalbank*, S. 57.
- NETWORKNT (2018): Github: json-schema-validator-perftest. Aufgerufen: 27.02.2019. URL: <https://github.com/networknt/json-schema-validator-perftest>.
- NEUMANN, KARL und ECKSTEIN, SILKE (2003): „Geography Markup Language (GML) - Eine Einführung aus Informatiksicht“. In: *Mitteilungen des Bundesamtes für Kartographie und Geodäsie* 24, S. 103–111.
- NIEMANN, HILDEGARD; MASCHKE, CHRISTIAN und HECHT, KARL (2005): „Lärmbedingte Belästigung und Erkrankungsrisiko. Ergebnisse des paneuropäischen LARES-Survey“. In: *Bundesgesundheitsblatt - Gesundheitsforschung - Gesundheitsschutz* 48, S. 315–328.
- NOTTINGHAM, MARK und WILDE, ERIK (2016): RFC-7807: Problem Details for HTTP APIs. URL: <https://tools.ietf.org/html/rfc7807>.
- NURSEITOV, NURZHAN et al. (Jan. 2009): „Comparison of JSON and XML data interchange formats: A case study“. In: *22nd International Conference on Computer Applications in Industry and Engineering 2009, Caine*, S. 157–162.
- OESTREICH, MARKUS und ROMBERG, OLIVER (2018): Keine Panik vor Statistik! Erfolg und Spaß im Horrorfach nichttechnischer Studiengänge. 6. Auflage. Berlin, Heidelberg, New York: Springer.

- OPEN GEOSPATIAL CONSORTIUM (2006): OpenGIS®Web Map Server Implementation Specification. Aufgerufen: 04.01.2019. URL: [http://portal.opengeospatial.org/files/?artifact\\_id=14416](http://portal.opengeospatial.org/files/?artifact_id=14416).
- (2010): OpenGIS Web Feature Service 2.0 Interface Standard. Aufgerufen: 10.02.2019. URL: [http://portal.opengeospatial.org/files/?artifact\\_id=39967](http://portal.opengeospatial.org/files/?artifact_id=39967).
  - (2018): OGC WPS 2.0.2 Interface Standard Corrigendum 2. Aufgerufen: 20.02.2019. URL: <https://portal.opengeospatial.org/files/14-065r2>.
  - (2019a): CityGML. Aufgerufen: 06.01.2019. URL: <http://www.opengeospatial.org/standards/citygml>.
  - (2019b): GML. Aufgerufen: 06.01.2019. URL: <http://www.opengeospatial.org/standards/gml>.
  - (2019c): KML. Aufgerufen: 06.01.2019. URL: <http://www.opengeospatial.org/standards/kml>.
- OPEN KNOWLEDGE INTERNATIONAL (2017): Open Definition 2.0. Aufgerufen: 27.01.2019. URL: <http://opendefinition.org/od/2.0/de/>.
- (2019): Global Open Data Index. Place overview. Aufgerufen: 27.01.2019. URL: <https://index.okfn.org/place/>.
- OPENSTREETMAP (2012): OpenStreetMap data license is ODbL. Aufgerufen: 08.02.2019. URL: <https://blog.openstreetmap.org/2012/09/12/openstreetmap-data-license-is-odbl/>.
- (2018a): List of OSM-based services. Aufgerufen: 13.02.2019. URL: [https://wiki.openstreetmap.org/wiki/List\\_of\\_OSM-based\\_services](https://wiki.openstreetmap.org/wiki/List_of_OSM-based_services).
  - (2018b): Stats. Aufgerufen: 13.02.2019. URL: <https://wiki.openstreetmap.org/wiki/Stats>.
  - (2018c): Urheberrecht und Lizenz. Aufgerufen: 10.02.2019. URL: <https://www.openstreetmap.org/copyright>.
  - (2019): Overpass API. Aufgerufen: 03.03.2019. URL: [https://wiki.openstreetmap.org/wiki/Overpass\\_API](https://wiki.openstreetmap.org/wiki/Overpass_API).
- ORACLE CORPORATION (2019): Jersey. RESTful Web Services in Java. Aufgerufen: 01.03.2019. URL: <https://jersey.github.io>.
- PADERTA, DAMIAN (2012): Open Data – Raumbezogene Daten. URL: <https://nbn-resolving.org/urn:nbn:de:0168-ssoar-364743>.
- PERCIVALL, GEORGE (2017): „OGC’s Open Standards for Geospatial Interoperability“. In: *Encyclopedia of GIS*. Hrsg. von SHEKHAR, SHASHI; XIONG, HUI und ZHOU, XUN. Cham: Springer, S. 1466–1473.
- PLATZ, JÜRGEN (2013): Immobilien-Management: Prüfkriterien zu Lage, Substanz, Rendite. 3. überarbeitete Auflage. Wiesbaden: Springer.
- RAMM, FREDERIK und TOPF, JOCHEN (2010): OpenStreetMap: Die freie Weltkarte nutzen und mitgestalten. Berlin: Lehmanns Media.
- REIMER, HELMUT (Apr. 2012): „Geoportal.DE freigeschaltet“. In: *Datenschutz und Datensicherheit - DuD* 36.4, S. 287–287.
- RICHARDSON, LEONARD und RUBY, SAM (2007): RESTful web services. Sebastopol: O’Reilly Media.
- SCHMOLL, FRITZ et al. (2015): Basiswissen Immobilienwirtschaft. 3. Auflage. München: Vahlen.

- SELFHTML (2018): HTTP/Statuscodes. Aufgerufen: 17.03.2019. URL: <https://wiki.selfhtml.org/wiki/HTTP/Statuscodes>.
- SENATSKANZLEI, GESCHÄFTS- UND KOORDINIERUNGSSTELLE GOVDATA (2016): Datenlizenz Deutschland. Aufgerufen: 28.02.2019. URL: <https://www.govdata.de/lizenzen>.
- SENATSVERWALTUNG FÜR STADTENTWICKLUNG UND WOHNEN / LAND BERLIN (2013): Nutzungsbestimmungen für die Bereitstellung von Geodaten des Landes Berlin. GeoNutz-BE. Aufgerufen: 10.02.2019. URL: <http://www.stadtentwicklung.berlin.de/geoinformation/download/nutzIII.pdf>.
- (2015a): Umweltatlas Berlin. 09.01 Umweltgerechtigkeit (Ausgabe 2015) – Kernindikator 1: Lärmbelastung. Aufgerufen: 29.03.2019. URL: [http://www.stadtentwicklung.berlin.de/umwelt/umweltatlas/d901\\_04.htm](http://www.stadtentwicklung.berlin.de/umwelt/umweltatlas/d901_04.htm).
  - (2015b): Umweltatlas Berlin. 09.01 Umweltgerechtigkeit (Ausgabe 2015) – Kernindikator 3: Grünflächenversorgung. Aufgerufen: 29.03.2019. URL: [http://www.stadtentwicklung.berlin.de/umwelt/umweltatlas/d901\\_06.htm](http://www.stadtentwicklung.berlin.de/umwelt/umweltatlas/d901_06.htm).
  - (2019): Umweltatlas Berlin. 06.06 Einwohnerdichte (Ausgabe 2018). Aufgerufen: 13.01.2019. URL: [https://www.stadtentwicklung.berlin.de/umwelt/umweltatlas/dn606\\_02.htm](https://www.stadtentwicklung.berlin.de/umwelt/umweltatlas/dn606_02.htm).
- SENATSVERWALTUNG FÜR WIRTSCHAFT, ENERGIE UND BETRIEBE (2019): Berlin Open Data. Aufgerufen: 26.03.2019. URL: <https://daten.berlin.de>.
- SENATSVERWALTUNG FÜR WIRTSCHAFT, ENERGIE UND BETRIEBE (2015): Schritt für Schritt – Nutzungsbedingungen. Aufgerufen: 12.02.2019. URL: <https://daten.berlin.de/datenbereinsteller/schritt-fuer-schritt/nutzungsbedingungen>.
- SEUSS, ROBERT (2015): „Open Geo Data – grenzenlos nutzbar?“ In: *Zeitschrift für Geodäsie, Geoinformation und Landmanagement* 140.2, S. 63–69.
- SMARTBEAR SOFTWARE (2019): API Design. Aufgerufen: 15.02.2019. URL: <https://swagger.io/solutions/api-design/>.
- SPICHALE, KAI (2017): API-Design – Praxishandbuch für Java- und Webservice-Entwickler. Heidelberg: dpunkt.verlag.
- STAATSBETRIEB SÄCHSISCHE INFORMATIK DIENSTE (2018): Open Data. Standard-Lizenzen. URL: <https://www.opendata.sachsen.de/634.htm>.
- STATISTISCHES BUNDESAMT (DESTATIS) (2018): Bautätigkeit und Wohnungen – Bestand an Wohnungen. Fachserie 5, Reihe 3. Aufgerufen: 18.02.2019. URL: [https://www.destatis.de/DE/Publikationen/Thematisch/Bauen/Wohnsituation/BestandWohnungen2050300177004.pdf?\\_\\_blob=publicationFile](https://www.destatis.de/DE/Publikationen/Thematisch/Bauen/Wohnsituation/BestandWohnungen2050300177004.pdf?__blob=publicationFile).
- STINGLWAGNER, CARL O. und NEUNDÖRFER, MATTHIAS (2007): „Einsatz von Geoinformationssystemen in der Immobilienwirtschaft“. In: *Zeitschrift für Geodäsie, Geoinformation und Landmanagement* 132.5, S. 331–335.
- STUBBLEBINE, TONY (2012): Reguläre Ausdrücke kurz & gut. 2. Auflage. Köln: O'Reilly.
- THE APACHE SOFTWARE FOUNDATION (2018): What is Maven? Aufgerufen: 07.02.2019. URL: <http://maven.apache.org/what-is-maven.html>.
- THE EUROPEAN GROUP OF VALUERS' ASSOCIATIONS (2003): Europäisches Objekt- und Marktrating: Ein Leitfaden für Gutachter.
- (2016): European Valuation Standards - EVS 2016.
- TICKOO, SHAM (2008): Learning Java Programming. Schereville: CADCIM Technologies.

- TIHOMIROVS, JURIS und GRABIS, JĀNIS (Jan. 2016): „Comparison of SOAP and REST Based Web Services Using Software Evaluation Metrics“. In: *Information Technology and Management Science* 19.1, S. 92–97.
- TILKOV, STEFAN et al. (2015): REST und HTTP. Heidelberg: dpunkt.verlag.
- TWITTER (2018): Response Codes – Error Codes. Aufgerufen: 13.02.2019. URL: <https://developer.twitter.com/en/docs/basics/response-codes.html>.
- VOGT, STEFFEN (2011): „WebGIS und OGC“. In: *Geographie. Physische Geographie und Human-geographie*. Hrsg. von GEBHARD, HANS et al. 2. Auflage. Heidelberg: Spektrum Akademischer Verlag, S. 205.
- VOSS, WINRICH und BAKKER, KENO (2017): „Einsatz von Geodaten für die Immobilienwertermittlung“. In: *Flächennutzungsmonitoring IX. Nachhaltigkeit der Siedlungs- und Verkehrsentwicklung?* Hrsg. von MEINEL, GOTTHARD et al. IÖR Schriften 73. Berlin: Rhombos, S. 113–122.
- WAGNER, DAVID (2002): „A Generalized Birthday Problem“. In: *Advances in Cryptology — CRYPTO 2002*. Hrsg. von YUNG, MOTI. Berlin, Heidelberg: Springer, S. 288–304.
- WORBOYS, MICHAEL F. (1995): GIS : A Computer Science Perspective. CRC Press.

---

# A Anhang

## A.1 Auswahl der implementierten Endpunkte

Operation	Pfad	Beschreibung
<b>json</b>		
GET	/json/request-schema	Returns the expected schema for /rate endpoint of the API.
POST	/json/validate-request	Validates given JSON against the expected schema for /rate endpoint.
GET	/json/error-schema	Returns schema for errors returned by the API.
<b>objects</b>		
PUT	/objects/reference-object	Saves reference object and returns internal UUID.
GET	/objects/reference-object/{uuid}	Returns reference object by its UUID.
DELETE	/objects/reference-object/{uuid}	Deletes a reference object by its UUID.
PUT	/objects/comparables	Saves comparable objects and returns internal UUID of the set.
GET	/objects/comparables/{uuid}	Returns a set of comparable objects by its UUID.
DELETE	/objects/comparables/{uuid}	Deletes a set of comparable objects by its UUID.
<b>rate</b>		
POST	/rate/by-ids/{uuid-ref}/{uuid-comp}	Rates the set of comparables (by UUID) concerning the given reference object (by UUID)
GET	/rate	Rates the set of comparables concerning the given reference object.
GET	/rate/sources	Returns indication of sources for integrated external services
<b>status</b>		
GET	/status	Verifies if the service is running.
GET	/status/external-services	Verifies if a connection to all of the external services can be established.
<b>test</b>		
GET	/test/benchmark-wfs	Fetches and parses the data of the WFS for the given number of times and returns time-logs.
GET	/test/benchmark-overpass	Fetches and parses the data of the Overpass API for the given number of times and returns time-logs.
GET	/test/rate	Creates given number of random positions in Berlin and fakes the rating process.

## A.2 JSON Schemata

Die in diesem Abschnitt gezeigten JSON Schemata wurden nicht veröffentlicht und besitzen demnach keine extern erreichbare URL, die als ID verwendet werden kann. Anstelle des URL-Basispfads wird „http://example.com“ als Platzhalter verwendet.

### A.2.1 ComparableObject

```
1 {
2   "$schema": "http://json-schema.org/draft-04/schema#",
3   "$id": "http://example.com/comparable-object.json",
4   "type": "object",
5   "definitions": {
6     "position": {
7       "type": "object",
8       "properties": {
9         "lat": {
10          "type": "number",
11          "description": "The latitude. As the API is designed to mainly work with EPSG:4326 use
12            this field as Y variable if using another EPSG code.",
13          "examples": [
14            52.46782684326172
15          ]
16        },
17        "lon": {
18          "type": "number",
19          "description": "The longitude. As the API is designed to mainly work with EPSG:4326 use
20            this field as X variable if using another EPSG code.",
21          "examples": [
22            13.340462684631348
23          ]
24        },
25        "epsg": {
26          "type": "integer",
27          "description": "The EPSG code the coordinates are referenced to.",
28          "default": 4326,
29          "examples": [
30            4326,
31            25833
32          ]
33        }
34      },
35      "required": [
36        "lat",
37        "lon"
38      ],
39      "additionalProperties": false
40    },
41    "comparableObject": {
42      "type": "object",
43      "properties": {
```

```
42     "id":{
43         "type":"string",
44         "description":"A unique identifier for the object.",
45         "examples":[
46             "A18-0001"
47         ]
48     },
49     "position":{
50         "$ref":"#/definitions/position"
51     }
52 },
53 "required":[
54     "id",
55     "position"
56 ],
57 "additionalProperties":false
58 }
59 },
60 "oneOf":[
61     {
62         "$ref":"#/definitions/comparableObject"
63     }
64 ],
65 "additionalProperties":false
66 }
```

## A.2.2 Rate-Request

```
1 {
2     "$schema":"http://json-schema.org/draft-04/schema#",
3     "$id":"http://example.com/rate-request.json",
4     "type":"object",
5     "properties":{
6         "reference":{
7             "$ref":"http://example.com/comparable-object.json"
8         },
9         "comparables":{
10            "type":"array",
11            "items":{
12                "$ref":"http://example.com/comparable-object.json"
13            },
14            "minItems":1,
15            "uniqueItems":true
16        }
17    },
18    "required":[
19        "reference",
20        "comparables"
21    ],
22    "additionalProperties":false
23 }
```

### A.2.3 Beispielinstanz eines Rate-Requests

```
1 {
2   "reference":{
3     "id": "Reference Object",
4     "position": {
5       "lat": 52.472690,
6       "lon": 13.317393
7     }
8   },
9   "comparables":[
10    {
11      "id": "Comparable 1",
12      "position": {
13        "lat": 52.4882081,
14        "lon": 13.3339177
15      }
16    },
17    {
18      "id": "Comparable 2",
19      "position": {
20        "lat": 52.516133,
21        "lon": 13.551410
22      }
23    }
24  ]
25 }
```

---

# Eidstattliche Erklärung

Ich versichere, diese Master Thesis ohne fremde Hilfe und ohne Verwendung anderer als der angeführten Quellen angefertigt zu haben. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen. Alle Ausführungen der Arbeit, die wörtlich oder sinngemäß übernommen wurden, sind gekennzeichnet.

Berlin, 31.03.2019

---

Maximilian Lohse