Fraunhofer FOKUS

NQDM

FRAUNHOFER-INSTITUT FÜR OFFENE KOMMUNIKATIONSSYSTEME FOKUS

LEITFADEN FÜR QUALITATIV HOCHWERTIGE DATEN UND METADATEN

Lina Bruns, Benjamin Dittwald, Fritz Meiners

LEITFADEN FÜR QUALITATIV HOCHWERTIGE DATEN UND METADATEN

Gefördert durch:

Bundesministerium für Wirtschaft und Energie

aufgrund eines Beschlusses des Deutschen Bundestages

Impressum

Verantwortliche Autoren

Lina Bruns, Benjamin Dittwald, Fritz Meiners
(alle Fraunhofer-Institut für Offene Kommunikationssysteme FOKUS, Berlin)

Weitere Autoren

Mirco Schwarz, Thomas Woge (beide con terra GmbH)

Reviewer und Mitwirkende

Sebastian Askar (Senatsverwaltung für Wirtschaft, Energie und Betriebe)
Michael Binzen (DB Systel GmbH & Bitkom AK Open Data)
Manuel Blaser (Bezirksamt Pankow von Berlin)
Martin Dames (Pumacy Technologies AG)
Rüdiger Dölle (Robert Koch Institut)
Karsten Gartner (Bezirksamt Pankow von Berlin)
Marc Groß (KGSt – Kommunale Gemeinschaftsstelle für Verwaltungsmanagement)
Christian Horn (Geschäfts- und Koordinierungsstelle GovData)
Christian Jacob (Stromnetz Berlin)
Dr. Anna Kasprzik (ZBW – Leibniz-Informationszentrum Wirtschaft)
Fabian Kirstein (Fraunhofer-Institut für Offene Kommunikationssysteme FOKUS)
Marc Kleemann (ISB AG)
Dr. Jens Klessmann (Fraunhofer-Institut für Offene Kommunikationssysteme FOKUS)
Karen König (Senatsverwaltung für Bildung, Jugend und Familie)
Daniel Korp (Bike Citizens Mobile Solutions GmbH)
Miriam Felicia Cathérine Krüger (Technische Universität Berlin)
Dr. Daniela Leitner (Senatsverwaltung für Umwelt, Verkehr und Klimaschutz)
Anna Mareike Matthis (Senatsverwaltung für Umwelt, Verkehr und Klimaschutz)
Rainer Pätzold (Pumacy Technologies AG)
Daniel Reimann (Bezirksamt Pankow von Berlin)
Thomas Tursics (Code for Germany – Community Rat)

Herausgeber

Fraunhofer-Institut für Offene Kommunikationssysteme FOKUS
Geschäftsbereich Digital Public Services DPS
Kaiserin-Augusta-Allee 31, 10589 Berlin
Telefon: +49-30-3463-7200
info@fokus.fraunhofer.de
www.fokus.fraunhofer.de

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Wirtschaft und Energie unter dem Förderkennzeichen 03TNG003A gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

NQDM – Leitfaden für qualitativ hochwertige Daten und Metadaten

Dieser Leitfaden bietet praktische Hilfestellungen und Empfehlungen zur Erreichung einer hohen Daten- und Metadatenqualität. Die enthaltenen Empfehlungen können grundsätzlich auf jegliche Art von Daten angewendet werden, unabhängig von Zugänglichkeit, Herkunft und dem sektoralen Bezug. Besonders ist der Leitfaden für Datenbereitsteller aus der öffentlichen Verwaltung empfehlenswert, die ihre Daten als Open Data veröffentlichen.

Im Leitfaden werden unterschiedliche Qualitätsdimensionen, Datenstrukturtypen und Bewertungsschemata für die Qualität von Daten und Metadaten aufgezeigt. Gängige maschinenlesbare und offene Daten- und Schnittstellenformate werden vorgestellt und anhand anschaulicher Beispiele wird aufgezeigt, wie eine hohe Datenqualität erreicht werden kann.

Der Leitfaden wurde im Rahmen des Projektes NQDM – Normentwurf für qualitativ hochwertige Daten und Metadaten – von Fraunhofer FOKUS im Zeitraum von September 2017 bis August 2019 erstellt. Weiterführende Informationen zu dem Projekt können unter https://www.nqdm-projekt.de/ eingesehen werden.

INHALTSVERZEICHNIS

ABBILDUNGSVERZEICHNIS

TABELLENVERZEICHNIS

1 Einleitung

1.1 Hintergrund & Motivation

Daten nehmen in der modernen Gesellschaft eine zentrale Rolle in allen Lebensbereichen ein. Sie sind die essentielle Grundlage moderner Technologien wie z.B. das Internet der Dinge oder autonomer vernetzter Systeme. Daten liefern jedoch nicht nur die Ressourcen für diese Technologien, sie sind gleichzeitig auch Grundlage zahlreicher Dienstleistungen und Geschäftsmodelle oder werden selbst als Produkt gehandelt. Unabhängig davon, wofür die Daten eingesetzt werden, ist ihre Qualität stets von zentraler Bedeutung.

Qualitativ hochwertige Daten und Metadaten sind die essentielle Grundlage für nutzenbringende Ergebnisse, funktionierende Geschäftsprozesse und informierte Entscheidungsfindungen. Daten verbrauchen sich nicht und können unendlich weiterverwendet werden. Liegen die Daten jedoch in geringer Qualität vor, so kann dies die Weiterverwendung erschweren oder im schlimmsten Fall sogar verhindern. In der Folge können »schlechte« Daten zu finanziellen Verlusten und Effizienzeinbußen führen. Laut Schätzungen von IBM kosteten qualitativ mindere Daten beispielsweise die USA im Jahr 2016 3,1 Billionen US-Dollar, das entsprach etwa 16% des US-amerikanischen BIP[1].

Diese beeindruckenden Zahlen verdeutlichen, dass beim Umgang mit Daten und Metadaten ein besonderes Augenmerk auf die Qualität gelegt werden sollte. Dabei gilt jedoch, dass die Qualität von Daten und Metadaten ein facettenreiches Konzept darstellt, welches je nach Nutzersicht, Bedürfnissen oder auch Prioritäten der Anwenderinnen und Anwender unterschiedlich bewertet wird.

Der vorliegende Leitfaden betrachtet verschiedene Aspekte der Daten- und Metadatenqualität. So werden unterschiedliche Qualitätsdimensionen und Bewertungsschemata vorgestellt und mit praktischen Anleitungen untermauert. Des Weiteren werden häufig genutzte Datenformate vorgestellt und aufgezeigt, wie diese verwendet und strukturiert werden sollten, um eine möglichst hohe Qualität der Daten zu erlangen.

Die im Leitfaden enthaltenen Empfehlungen können grundsätzlich auf alle Arten von Daten angewendet werden, unabhängig von ihrem sektoralen Bezug (Verkehr, Politik, Umwelt etc.), ihrer Herkunft (Unternehmen, Verwaltungen etc.) und ihrer Zugänglichkeit (intern, kommerziell, frei verfügbar). Ein besonderes Augenmerk wird jedoch auf frei verfügbare Daten und hier im Speziellen auf Open Data gelegt: Behörden der Bundesverwaltung und einiger Landesverwaltungen unterliegen der gesetzlichen Verpflichtung, ihre Daten der Öffentlichkeit frei zur Verfügung zu stellen. Open Data wird ein großes Potenzial nachgesagt: Es kann Treiber für Innovation, Zusammenarbeit, Teilhabe und Transparenz sein. Die tatsächliche Wertschöpfung

[1] IBM (o.D): The Four V's of Big Data. Zuletzt aufgerufen im August 2019 unter https://www.ibmbigdatahub.com/infographic/four-vs-big-data.

dieser offenen Daten wird jedoch oftmals durch eine niedrige Datenqualität gehemmt. Datenbereitsteller aus der öffentlichen Verwaltung finden daher im vorliegenden Leitfaden zusätzlich einen gesonderten Exkurs zu Open Data.

1.2 Methodischer Aufbau

Der Leitfaden folgt einem modularen Aufbau. Für diejenigen Leserinnen und Leser, die bisher noch keine oder nur wenig Berührungspunkte mit den Themen Daten- und Metadatenqualität hatten, empfiehlt sich das Lesen in chronologischer Reihenfolge, da insbesondere in den vorderen Kapiteln Grundlagen erläutert werden. Kapitel 3 enthält einen Exkurs zu Open Data. In Kapitel 4 werden zwei gängige Bewertungsschemata für die Qualität von Daten vorgestellt. Kapitel 5 befasst sich mit allgemeinen Qualitätsmerkmalen für Daten und Metadaten. Kapitel 6 erläutert beispielhaft den Unterschied zwischen strukturierten und nicht-strukturierten Daten. Leserinnen und Leser, die Handlungsempfehlungen zu konkreten Dateiformaten suchen, werden in den Unterabschnitten zu Kapitel 7 fündig. Das achte Kapitel setzt sich mit der Qualität von Metadaten auseinander.

2 Zentrale Begriffe

Nachfolgend werden einige für den Leitfaden zentrale Begriffe definiert. Weitere Kurzdefinitionen befinden sich im Glossar am Ende des Leitfadens ab Seite 64. Die Begriffe sind im Text entsprechend blau hervorgehoben.

Daten

Daten bestehen aus Angaben, Zeichen oder Symbolen, die durch Interpretation zu Informationen werden. Daten können erzeugt, gesammelt, aufbereitet, visualisiert, analysiert und zu Information und Wissen angereichert werden. Gespeichert werden sie in Dateien oder Datenbanken als Klartext, Grafik oder in Listen- bzw. Tabellenform. In ihrer unbearbeiteten Form bezeichnet man Daten als Basis- oder **Rohdaten**. Werden Daten weiterverarbeitet, nennt man sie häufig Mehrwertdaten. Daten, die Informationen und Beschreibungen über Rohdaten oder Mehrwertdaten bereitstellen, werden als **Metadaten** bezeichnet.

Metadaten

Metadaten werden für die Erfassung und Beschreibung eines **Datensatzes** in strukturierter Form verwendet. Sie enthalten bspw. Informationen über den Inhalt, den Titel oder das Format eines Datensatzes. Kurz gesagt sind Metadaten Daten über Daten bzw. Verweise auf die eigentlichen Daten. Dabei folgen Metadaten meist einem bestimmten Schema, welches obligatorische und optionale Informationen über den Datensatz vorgibt.

Datenqualität

Im Kontext des europäischen Datenportals werden Daten als qualitativ hochwertig angesehen, »wenn sie für ihren vorgesehenen Gebrauch im operativen Geschäft, bei Entscheidungen und bei der Planung geeignet sind«.[2] Nach dieser Definition ist die Qualität von Daten abhängig von ihrer Nutzbarkeit. Je nach Domäne, Verwendungskontext, Struktur, Inhalt und Vorwissen der **Datennutzer** können Daten somit verschiedene Merkmale aufweisen, die ihre Qualität beeinflussen.

Allgemein sollten **offene Daten** in einem offenen und maschinenlesbaren Format frei zugänglich sein. Auf diesen Prinzipien bauen auch gängige Bewertungsschemata für Datenqualität auf, indem für die Messung der Qualität u.a. die Offenheit der Daten herangezogen wird (siehe Abschnitt 3.1).

Des Weiteren gibt es eine Vielzahl von Dimensionen, die zur Beurteilung der Datenqualität herangezogen werden können: Vollständigkeit, Glaubwürdigkeit, Aktualität, Relevanz etc. (siehe Abschnitt 5). Da Daten oftmals domänenspezifischen Anforderungen folgen, lassen sich die Qualitätsmerkmale jedoch kaum abstrakt

[2] Europäische Kommission (2013): Die Qualität von offenen Daten und Metadaten. Zuletzt aufgerufen im August 2019 unter https://www.europeandataportal.eu/sites/default/files/d2.1.2_training_module_2.2_open_data_quality_de_edp.pdf.

quantifizieren. Ein Beispiel für domänenspezifische Datenschemata der öffentlichen Verwaltung sind die XÖV-Standards[3].

Metadatenqualität

Qualitativ hochwertige Metadaten erleichtern das Auffinden und die Nutzung von Daten. Da Metadaten ebenfalls Daten (über Daten) darstellen, können ähnliche Merkmale zur Bewertung ihrer Qualität herangezogen werden wie für Daten (Abschnitt 5).

Darüber hinaus legen Metadatenschemata in der Regel fest, welche Angaben obligatorisch und welche optional sind. Teilweise ist es jedoch sinnvoll, auch als optional eingestufte Felder auszufüllen, wenn diese die Qualität der Metadaten steigern. Ein Punktesystem hilft **Datenbereitstellern** bei der Erkennung der für die **Metadatenqualität** relevanten Felder (Abschnitt 8.5).

[3] Weitere Informationen unter https://www.xoev.de/.

3 Exkurs Open Data

Die öffentliche Verwaltung erhebt eine Vielzahl unterschiedlicher Daten und verfügt damit über einen enormen Datenbestand. Viele dieser Daten sind von öffentlichem Interesse und können helfen, mehr Transparenz, Teilhabe oder auch neuartige Geschäftsmodelle zu schaffen. Damit das Potenzial genutzt werden kann, müssen die Daten der Allgemeinheit zur Verfügung gestellt werden. Viele Bundesländer (bspw. Nordrhein-Westfalen, Berlin, Thüringen, Baden-Württemberg oder Hamburg) planen oder haben bereits Gesetze erlassen, die die Veröffentlichung von Daten für Landesbehörden vorschreiben. Auch auf kommunaler Ebene nimmt die Anzahl an Open-Data-Portalen stetig zu. Um Open Data auch auf Bundesebene zu befördern, wurde 2017 das **E-Government-Gesetz** (EGovG) um einen Open-Data-Paragrafen erweitert, der Bundesverwaltungen die Öffnung ihrer Daten in maschinenlesbarer Form gesetzlich vorschreibt (§12a EGovG).

3.1 Gründe für Open Data

Laut einer Studie der Konrad-Adenauer-Stiftung[4] liegt in Deutschland das wirtschaftliche Potenzial von **Open Data** in den nächsten zehn Jahren zwischen 12,1 und 131 Milliarden Euro. Die Nutzergruppen von Open Data reichen dabei von Startups über Unternehmen, Datenjournalisten, gemeinnützige Organisationen und Hobby-Entwickler. Doch letztlich profitiert vor allem die Verwaltung selbst von der Öffnung ihrer eigenen Daten:

Werden **Verwaltungsdaten** an einer zentralen Stelle veröffentlicht, vereinfacht sich der Zugang zu den Daten auch für Verwaltungen. So ermöglicht es Open Data, Daten effizienter unter Behörden auszutauschen, ohne dass hierfür ein bürokratischer Akt notwendig ist. Die Zusammenarbeit kann sich hierdurch nicht nur über Behördengrenzen hinweg verbessern. Bedingt durch Aufgabentrennung und Ressort- oder Abteilungsgrenzen liegen Daten oftmals sogar innerhalb einer Organisation in geschlossenen Silos. Werden die Daten als Open Data veröffentlicht, stehen diese Informationen der gesamten Behörde und allen weiteren Interessierten offen, wodurch eine bessere Nachnutzung der Daten erfolgen und doppelte Datenerhebungen vermieden werden können.

Mit der Veröffentlichung von Daten als offene Daten bieten Verwaltungen Informationen proaktiv an. Hiervon profitieren nicht nur **Datennutzer**, sondern letztlich auch die bereitstellende Behörde selbst: Laut einer Studie der **Sunlight Foundation**[5] verzeichnen Behörden, die ihre Daten auf Open-Data-Portalen veröffentlichen, signifikant weniger Anfragen und Anträge auf Zugang zu öffentlichen Akten als Behörden, die ihre Daten nicht veröffentlichen. Somit können Behörden, die regelmäßig Daten veröffentlichen, Zeit einsparen. Daneben wächst mit der Veröffentlichung von Daten und – im besten Fall – auch der Vernetzung von Daten

[4] Kuzev, P. (Hrsg.) (2016): Open Data. The Benefits. Das volkwirtschaftliche Potential für Deutschland. Zuletzt aufgerufen im August 2019 unter http://www.kas.de/wf/de/33.44906/.

[5] Stern, A. (2018): Open Data Policy and Freedom of Information Law. Understanding the relationship between the twin pillars of access to information. Zuletzt abgerufen im August 2019 unter http://sunlightfoundation.com/wp-content/uploads/2018/10/alena-white-paper-PDF.pdf.

untereinander (**Linked Open Data**) die Wissensbasis der Verwaltung kontinuierlich. Open Data sollte daher von Verwaltungen als Chance betrachtet werden, den eigenen Arbeitsalltag zu erleichtern und interne Prozesse zu optimieren.

Damit sich das volle Potenzial von Open Data entfalten kann, ist es wichtig, dass die Daten und Metadaten in ausreichender Qualität vorliegen. Angefangen bei der Wahl der »richtigen«, d.h. offenen und maschinenlesbaren Datenformate, erstreckt sich die Qualität von Daten u.a. auch auf ihre semantische und strukturelle Gestaltung. Ebenso wichtig wie die Qualität der eigentlichen Daten ist hierbei die Qualität der Metadaten. Metadaten sind Daten über Daten, sie liefern zusätzliche Informationen und beschreiben die zu veröffentlichenden Daten. Liegen die Metadaten in mangelnder Qualität vor, können Interessierte die eigentlichen Daten womöglich nicht auffinden und ihr Potenzial bleibt ungenutzt.

3.2 Definition von Open Data

Open Data

»Open Data sind ungefilterte und maschinenlesbare elektronische Daten, die jedem öffentlich, zweckfrei und unverbindlich zur Verfügung gestellt werden. Der Zugriff ist jederzeit, ohne verpflichtende Registrierung und ohne Begründung möglich. Sie werden unverzüglich und entgeltfrei zur uneingeschränkten Weiterverwendung für jedermann einfach angeboten«.[6]

Das Prinzip der Offenheit von Open Data bezieht sich auf zwei Aspekte: Aus rechtlicher Sicht werden die Daten unter einer offenen **Lizenz** mit der Möglichkeit der Weiterverwendung und Weitergabe veröffentlicht. Dies umfasst auch die Veränderung durch Dritte unter kommerziellen Gesichtspunkten. Der Aspekt der technischen Offenheit bezieht sich auf die Bereitstellung der Daten in einem maschinenlesbaren und nicht-proprietären Dateiformat, sodass die Daten universell genutzt und verarbeitet werden können.

3.3 Grundprinzipien von Open Data

Bereits im Jahr 2007 wurden acht Prinzipien von einer Expertengruppe für die Veröffentlichung von Open Data[7] aufgestellt, die zu einem späteren Zeitpunkt durch die Sunlight Foundation[8] ergänzt wurden. In Ergänzung zu den zehn Prinzipien fasst die Sunlight Foundation in einem »lebenden« Dokument[9] weitere Prinzipien

[6] Bitkom (2017): Open Data Manifest. Zuletzt aufgerufen im August 2019 unter https://www.bitkom.org/sites/default/files/file/import/170420-Open-Data-Manifest-finaler-Entwurf.pdf.

[7] Sebastopol Group (2007): The eight principles of Open Government Data. Zuletzt aufgerufen im August 2019 unter https://opengovdata.org/.

[8] Sunlight Foundation (2010): Ten Principles for Opening Up Government Information. Zuletzt aufgerufen im August 2019 unter https://sunlightfoundation.com/policy/documents/ten-open-data-principles/.

[9] Das Dokument kann unter folgendem Link eingesehen werden: https://opendatapolicyhub.sunlightfoundation.com/guidelines/.

zusammen, die bei der Veröffentlichung von Daten hilfreich sind. Nachfolgend werden die originären zehn Prinzipien kurz vorgestellt.

1. Vollständigkeit

Daten sollten so vollständig wie möglich veröffentlicht werden. Hierzu gehört auch die Veröffentlichung der zu den Daten gehörigen Metadaten. Zusammenhängende Daten, wie etwa alle Messwerte eines Tages, sollten auf einmal herunterladbar sein (**Bulk-Download**).

2. Primärquelle

Daten sollen mit dem höchstmöglichen Grad an Granularität als Rohdaten veröffentlicht, und nicht aggregiert oder modifiziert werden. Liegt beispielsweise eine auf Daten erhobene Statistik vor, so ist nicht die Statistik selbst, sondern die ihr zugrundeliegenden Rohdaten zu veröffentlichen. Die Statistik kann zusätzlich zu den Rohdaten veröffentlicht werden.

3. Aktualität der Daten

Um den Wert der Daten zu erhalten, sollten diese so schnell wie möglich nach der Datenerhebung zur Verfügung gestellt werden.

4. Zugänglichkeit

Die Daten werden barrierefrei veröffentlicht, so dass sie allen Interessierten zur Verfügung stehen. Eine leichte Auffindbarkeit und eine einfache Suche erhöhen die Zugänglichkeit der Daten. Auch sollte es möglich sein, mit den Daten über eine **API** (Programmierschnittstelle) zu interagieren.

5. Maschineninterpretierbarkeit

Um eine automatisierte Verarbeitung zu ermöglichen, müssen die Daten angemessen strukturiert und in maschineninterpretierbaren Formaten zur Verfügung gestellt werden. Die Maschineninterpretierbarkeit beschreibt, ob ein Datensatz von einer Software interpretiert und weiterverarbeitet werden kann. Vollständig maschineninterpretierbare Formate sind u.a. **JSON**, **XML** und **RDF**. Eine Übersicht über die Maschineninterpretierbarkeit von Formaten findet sich in Tabelle 1 (Kapitel 4.1).

6. Diskriminierungsfreiheit

Jede Person hat jederzeit Zugriff auf die Daten. Eine Identifizierung oder Registrierung der Person ist nicht notwendig, um auf die Daten zugreifen zu können.

7. Offene Standards

Daten sollen in offenen Formaten bereitgestellt werden. Ein Format ist dann offen, wenn kein Spezialprogramm für die Verarbeitung der Daten benötigt wird. Auch wenn einige proprietäre Formate nahezu überall verbreitet sind, sollten dennoch offene Formate verwendet werden, um sicherzustellen, dass jeder Person der Zugriff auf die Daten möglich ist. Zudem ist es sinnvoll, die Daten in verschiedenen Formaten zugänglich zu machen. Eine Übersicht über die Offenheit von Formaten findet sich in Tabelle 2 (Kapitel 4.1).

8. Lizenzierung

Lizenzen regeln die Weiterverwendung der veröffentlichten Daten. Im Sinne von Open Data sollten Daten aus rechtlicher Perspektive so offen wie möglich verfügbar gemacht werden. Hierbei ist zu beachten, dass es aus rechtlicher Sicht Mindestanforderungen an die Offenheit von Daten gibt, sofern sie als Open Data zur Verfügung gestellt werden sollen. Wird lediglich eine Weiterverwendung der Daten in nicht kommerzieller Hinsicht erlaubt oder werden andere Restriktionen bzw. einschränkende Nutzungsbestimmungen genannt, handelt es sich nicht um Open Data.

Um den Aufwand zu reduzieren, sollte auf vorgefertigte Lizenzmodelle wie beispielsweise **Datenlizenz Deutschland 2.0** oder **Creative Commons** zurückgegriffen werden.

9. Beständigkeit

Daten, die veröffentlicht werden, sollten auf Dauer online bleiben und langfristig auffindbar sein. Bei Aktualisierungen oder Veränderungen sollte eine entsprechende **Versionierung** der Daten gepflegt werden.

10. Nutzungskosten

Das Erheben von Gebühren für den Zugriff auf Daten ist ein Hindernis, welches den Zugang zu den Daten einschränkt. Daher sollten die Daten im Sinne von Open Data kostenfrei zur Verfügung gestellt werden.

4 Bewertungsschemata für die Qualität von Daten

Im Folgenden werden zwei Bewertungsschemata für die Qualität von Daten vorgestellt: Das »5-Sterne-Modell« von Tim Berners-Lee, welches Daten allgemein in Hinblick auf ihre Offenheit und Verwendbarkeit bewertet und der »Global Open Data Index«, der speziell Qualitätsaspekte von Open Data betrachtet.

4.1 5-Sterne-Modell

Tim Berners-Lee, Direktor des World Wide Web Consortiums (W3C), entwickelte das 5-Sterne Modell im Jahr 2001.[10] Das Modell bewertet die technische Verwendbarkeit und Zugänglichkeit von Datensätzen. Die einzelnen Stufen des 5-Sterne-Modells werden nachfolgend erläutert:

☆	Offene Lizenz
☆ ☆	Wiederverwendbares Format
☆ ☆ ☆	Offenes Format
☆ ☆ ☆ ☆	Eindeutige Identifizierung (durch URIs)
☆ ☆ ☆ ☆ ☆	Vernetzung mit anderen Daten

Die Stufen sind kaskadierend: Damit ein Datensatz beispielsweise eine Drei-Sterne-Bewertung erhalten kann, müssen die Stufen eins bis drei vollständig erfüllt sein. Nachfolgend werden die einzelnen Stufen inklusive der Mehrwerte für Nutzerinnen und Nutzer sowie Datenanwenderinnen und Datenanwender vorgestellt. Weiterführende Informationen und Beispiele für die Umsetzung der einzelnen Stufen finden sich unter: https://5stardata.info/de/.

☆ Offene Lizenz

Der erste Stern wird – unabhängig von der eigentlichen Datenqualität – an Datensätze verliehen, welche unter einer offenen Lizenz zur Verfügung gestellt werden. Dies bedeutet, dass der Datensatz für kommerzielle und nicht-kommerzielle Zwecke verwendet werden darf.

Mehrwerte:

- Nutzerinnen und Nutzer können den Datensatz lokal speichern, ändern und teilen.
- Datenherausgeber können mit geringem Aufwand einmalig festlegen, dass die Daten weiterverwendet werden dürfen.

[10] Weitere Informationen unter http://5stardata.info/en/.

☆☆ Wiederverwendbares Format

Ist ein Datensatz neben der offenen Lizenz zusätzlich leicht wiederverwendbar, erhält er zwei Sterne. Eine einfache Wiederverwendbarkeit wird durch die Nutzung von maschinenlesbaren Formaten, wie z.B. Microsoft Excel, erreicht.

Mehrwerte:
Nutzerinnen und Nutzer können:
- den Datensatz direkt in der proprietären Software verarbeiten.
- die Daten in ein anderes Format exportieren.

☆☆☆ Offenes Format

Zur Erreichung der dritten Stufe muss der Datensatz unter einer offenen Lizenz in einem wiederverwendbaren, nicht proprietären Format bereitgestellt werden. Die Verwendung eines nicht proprietären Formates bedeutet, dass der Datensatz unabhängig von einer spezifischen Software verwendet werden kann.

Mehrwerte:
Nutzerinnen und Nutzer können den Datensatz verarbeiten, ohne eine proprietäre Software verwenden zu müssen.

☆☆☆☆ Eindeutige Identifizierung

Die vierte Stufe erreicht ein Datensatz, sofern er unter einer offenen Lizenz in einem wiederverwendbaren, offenen Format zur Verfügung gestellt und zusätzlich eindeutig identifiziert wird, z.B. durch die Verwendung von **URIs** (Uniform Resource Identifiers).

Mehrwerte:
Nutzerinnen und Nutzer können:
- den Datensatz im Web oder lokal verlinken.
- Lesezeichen erstellen.
- die Daten mit anderen Daten kombinieren.

Datenherausgeber können:
- den Zugang zu den Daten optimieren.
- Verlinkungen von anderen Datenherausgebern zu den Daten erhalten.

☆☆☆☆☆ Vernetzung mit anderen Daten

Ein Datensatz erreicht die letzte Stufe, wenn alle vorherigen Stufen erreicht wurden und zusätzlich eine Verlinkung der Daten mit anderen Daten erfolgt. Diese Verlinkung ermöglicht es, den Datensatz in einen Kontext zu stellen und zwischen verschiedenen Datenpunkten zu navigieren.

Mehrwerte:
Nutzerinnen und Nutzer können:
- während der Nutzung ähnliche Daten entdecken.
- Informationen über das Datenschema erhalten.

Datenherausgeber:

- steigern die Auffindbarkeit ihrer Daten und erhöhen ihren Wert.

Das 5-Sterne-Modell trägt zur Weiterverwendbarkeit von offenen Daten bei. Die ersten drei Stufen können Datenbereitsteller in der Regel ohne größere Aufwände umsetzen. Die Erreichung der vierten und fünften Stufe des Modells ist hingegen zeitintensiver und erfordert vertiefte Kenntnisse. Daher sollten unerfahrene Datenbereitsteller bei der erstmaligen Veröffentlichung von Daten nicht sofort die Erreichung der fünften Stufe des Modells anvisieren. **Eine konsequente Umsetzung der ersten drei Stufen liefert bereits einen großen Mehrwert für Datennutzer**. Abbildung 1 verdeutlicht die einzelnen Stufen des 5-Sterne-Modells.

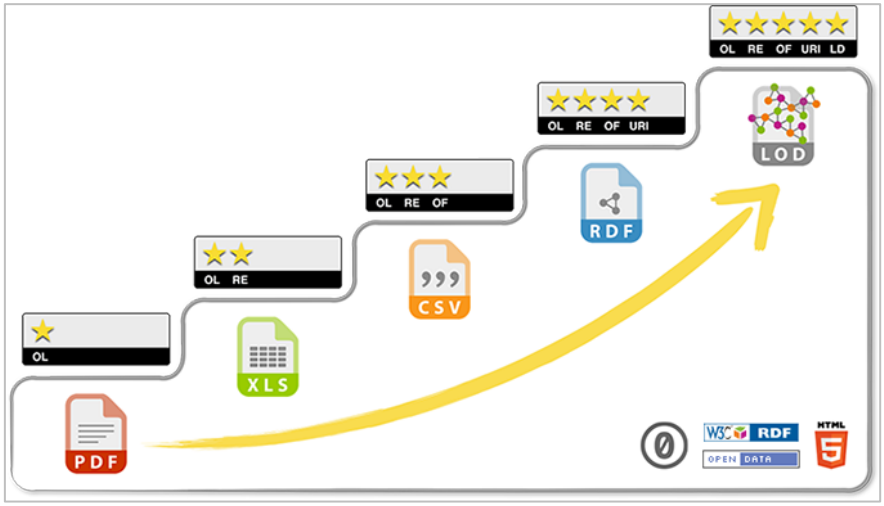


Abbildung 1: 5-Sterne-Modell von Tim Berners-Lee[11].

[11] Hausenblas, M. (2012): 5 ★ Open Data. Zuletzt aufgerufen im August 2019 unter https://5stardata.info/en/.

Eine Übersicht über maschineninterpretierbare und offene Formate liefern die folgenden Tabellen (keine abschließende Aufzählung).

Tabelle 1: Maschineninterpretierbarkeit.[12]

MASCHINENINTERPRETIERBARKEIT	
Gar nicht bis gering	PDF, DOC, DOCX, GIF, JPG, JPEG, PNG, TIFF, GeoTIFF, ODT
Überwiegend	TXT, RTF, ODS, XLS, CSV, HTML, XLSX
Vollständig	XML, RDF, RSS, KMZ, DXF, GPX, GML

Tabelle 2: Offene Standards[13].

OFFENE STANDARDS	
Proprietäre Formate	XLS, DOC, PPT
Standardisierte, aber nicht offene Formate	RTF, GIF, JPG/JPEG, TIFF, GeoTIFF, DXF, GPX
Standardisierte, offene Formate	TXT, CSV, HTML, XML, RDF, ODT, ODS, RSS, XLSX, PDF, PNG, DOCX, KMZ, GML

4.2 Global Open Data Index

Der Global Open Data Index wurde von 2013 bis 2017 von der Open Knowledge Foundation jährlich erhoben. Der Index misst das Maß an Offenheit staatlicher bzw. öffentlicher (Verwaltungs-)Daten auf nationaler Ebene. Bezüglich der Definition offener Daten lehnt sich der Index an die Open Definition[14] an, die bspw. Prinzipien der offenen Lizenzierung, der Zugänglichkeit, dem Format sowie der **Maschinenlesbarkeit** formuliert.

Der Global Open Data Index bewertet anhand verschiedener Fragen (Tabelle 3) die Qualität von ausgewählten nationalen Regierungsdaten diverser Länder. Als Ergebnis der Bewertung erhalten die Länder einen »Score« für die Daten (höchstens 100%), sodass ein Ranking bezüglich der Qualität der untersuchten Datensätze entsteht. 40 Punkte können in dem Bereich der rechtlichen und technischen Offenheit der Datensätze erzielt werden, während Aspekte wie rechtzeitige Veröffentlichung, Verfügbarkeit und Zugänglichkeit mit insgesamt bis zu 60 Punkten bewertet werden.

[12] Siehe Klessmann, J.; Denker, P.; Schulz, S. E.; u. a.; Bundesministerium des Innern (Hrsg.) (2012): Open Government Data Deutschland. Zuletzt aufgerufen im August 2019 unter https://www.verwaltung-innovativ.de/SharedDocs/Publikationen/eGovernment/open_government_data_deutschland_langfassung.pdf?__blob=publicationFile&v=5.

[13] Siehe ebd.

[14] Weitere Informationen unter https://opendefinition.org/od/2.1/en/.

Auch wenn der Global Open Data Index primär für die Bewertung ausgewählter Regierungsdaten auf nationaler Ebene verwendet wird, kann dieses Bewertungsschema von jeder veröffentlichenden Stelle genutzt werden, um die eigenen Daten hinsichtlich ihrer Qualität zu überprüfen.

Tabelle 3 zeigt die nach dem Global Open Data Index zur Bewertung der Datenqualität herangezogenen Fragen, ihre Gewichtung und eine Erläuterung[15].

Tabelle 3: Kriterien zur Bewertung der Datenqualität nach dem Global Open Data Index.

FRAGE	GEW.	ERLÄUTERUNG
Sind die Daten online verfügbar, ohne dass hierfür eine Registrierung oder Anfrage notwendig ist?	15	Online-Verfügbarkeit ist die Voraussetzung für Offenheit. Auf Registrierungspflichten sollte verzichtet werden, da sie potenzielle Anwenderinnen und Anwender von der Nutzung der Daten abhalten kann.
Stehen die Daten kostenfrei zur Verfügung?	15	Um die Zugänglichkeit zu Daten für jedermann zu ermöglichen, sollten sie möglichst kostenfrei zur Verfügung gestellt werden.
Können die Daten gleichzeitig heruntergeladen werden (Bulk-Download)?	15	Daten sollten auf einfache Weise zeitgleich abgerufen werden können. **Captchas**, Anfragen oder Schnittstellen, die lediglich den Zugriff auf einen Teil der Daten ermöglichen, können beschränkend wirken und entsprechen nicht dem Prinzip der Offenheit.
Werden die Daten zeitnah und aktuell bereitgestellt?	15	Einige Daten sind nach ihrer Veröffentlichung am wertvollsten (z.B. Wettervorhersagen, Wahlergebnisse). Dementsprechend ist ihre rechtzeitige Bereitstellung von großer Bedeutung.
Werden die Daten unter einer offenen Lizenz verfügbar gemacht?	20	Die freie Verwendbarkeit der Daten aus rechtlicher Sicht ist eine Kernanforderung an Open Data.
Werden die Daten in offenem und maschinenlesbarem Format zur Verfügung gestellt?	20	Die Maschinenlesbarkeit und Offenheit der Formate ist eine Kernanforderung an Open Data.

Anhand der Gewichtung ist bereits erkennbar, dass für qualitativ hochwertige offene Daten die Bereitstellung unter **offener Lizenz** in einem **maschinenlesbaren Format** von besonderer Bedeutung ist.

[15] Die Methodik des Global Open Data Index beinhaltet weitere Fragen, die jedoch nicht in den Score einfließen. Eine Auflistung aller Fragen findet sich unter https://index.okfn.org/methodology/.

5 Qualitätsmerkmale für Daten und Metadaten

Auch das Themenfeld der Datenqualität liefert Ansätze zur Untersuchung der Qualität von Daten und Metadaten. In der Fachliteratur findet man zahlreiche Auflistungen von Qualitätsparametern (sogenannten Dimensionen). Hierbei steht im Gegensatz zu den in Kapitel 4 vorgestellten Schemata nicht die Offenheit und Weiterverwendbarkeit der Daten im Fokus. Vielmehr wird die inhaltliche und kontextuelle Qualität der Daten adressiert. Die Qualitätsdimensionen wurden hauptsächlich geprägt durch die Arbeiten von Wang und Strong zur Reduktion von fehlerhaften Daten in Datenbanken Mitte der 1990er Jahre[16].

Eine allgemeingültige und umfassende Auffassung, welche Dimensionen für Daten und Metadaten von Bedeutung sind, herrscht in der Fachliteratur jedoch nicht. Dies liegt unter anderem daran, dass die Relevanz der einzelnen Qualitätsdimensionen in Abhängigkeit vom jeweiligen Verwendungskontext differiert. So können beispielsweise Daten, die mehrere Jahre alt sind und historische Begebenheiten beschreiben, durchaus aktuell genug sein, während Fahrplandaten oft schon viel früher als überholt gelten. Dieses Beispiel verdeutlicht auch, dass nicht für jede Dimension objektive und allgemein gültige Metriken angewendet werden können. Zu diesem Schluss kommen auch die Autoren Neumaier, Umbrich und Polleres in ihrer Arbeit *Automated Quality Assessment of Metadata across Open Data Portals*[17]. Auch ohne die Nennung von konkreten Metriken zur Überprüfung der Qualität liefern die Dimensionen einen Einblick in die Komplexität der Thematik und können Datenbereitsteller in dieser Hinsicht sensibilisieren. Daher wird nachfolgend ein Auszug der meistgenannten Qualitätsdimensionen geliefert.

Aktualität

Die Daten sollten in regelmäßigen Intervallen in Hinblick auf ihre Aktualität überprüft werden. Das Aktualisierungsintervall sollte auch in den Metadaten Erwähnung finden, damit Datennutzerinnen und -nutzer die Aktualität der Daten einschätzen können.

Es ist darauf zu achten, dass nicht nur die Daten selbst aktuell sind, sondern auch die entsprechenden Metadaten. Beispielsweise ist es für Datennutzerinnen und -nutzer schwierig, Fragen oder Feedback zu den Daten zu stellen bzw. zu geben, wenn die Kontaktangaben in den Metadaten veraltet sind.

Damit ersichtlich wird, welche Daten nach einer Aktualisierung die neuesten sind, sollte der Ressourcenname einen Zeitstempel und ggf. eine Versionsnummer enthalten.

[16] Wang, R. Y.; Strong, D. M. (1996): Beyond Accuracy: What Data Quality Means to Data Consumers. In: *Journal of Management Information Systems* 12 (4), S. 5-33. DOI: 10.1080/07421222.1996.11518099.
[17] Neumaier, S.; Umbrich, J.; Polleres, A. (2016): Automated Quality Assessment of Metadata across Open Data Portals. In: *Journal of Data and Information Quality* 8 (1), S. 1-29. DOI: 10.1145/2964909.

Fehlerfreiheit

Die Daten und Metadaten sollten korrekte Werte beinhalten und möglichst fehlerfrei sein.

Hinweis: Beinhalten die Daten kleinere Fehler, ist dies zunächst kein Hinderungsgrund für die Veröffentlichung. Jedoch sollte auf die entsprechenden Fehler in den Metadaten hingewiesen werden.

Genauigkeit

Je nach Verwendungskontext kann es von hoher Relevanz für die Datennutzerin oder den Datennutzer sein, dass Daten, wie z.B. Messwerte, so genau wie möglich angegeben und nicht gerundet werden. Auch Metadaten sollten eine hohe Genauigkeit aufweisen. So sollte beispielsweise die inhaltliche Beschreibung der Daten möglichst präzise erfolgen, sodass potenzielle Datennutzerinnen und -nutzer eine realistische Vorstellung über die Daten erhalten und die Relevanz der Daten für ihren eigenen Kontext schnell einschätzen können.

Konformität

Bei der Bereitstellung von Daten und Metadaten sollten jeweils relevante Standards berücksichtigt werden (z.B. Datumsangabe nach ISO 8601).

Auch sollte auf Erwartungskonformität geachtet werden. Damit ist gemeint, dass das bereitgestellte Material dem entspricht, was von den Nutzerinnen und Nutzern erwartet wird, beispielsweise bei der Benennung von Attributen und Vokabeln.

Hinweis: Es existieren oftmals domänenspezifische Standards (z.B. SDMX[18] für Statistikdaten). Vor einer Veröffentlichung der Daten sollte daher geprüft werden, welche Standards relevant sind und ob diese eingehalten werden. Auch die Verwendung von kontrollierten Vokabularen sollte berücksichtigt werden.

Konsistenz

Daten und Metadaten sollten widerspruchsfrei sein, sowohl in sich selbst als auch Datensatz-übergreifend.

Ein Datensatz ist dann beispielsweise in sich konsistent, wenn das Erstellungsdatum vor der letzten Änderung liegt (zeitliche Konsistenz).

Hinweis: Werden Informationen aus verschiedenen Quellen zusammengefügt, ist besonders genau auf die Konsistenz der Daten und Metadaten zu achten, beispielsweise in Hinblick auf die Lizenz.

Transparenz und Vertrauenswürdigkeit

Ursprung, Originalität und Veränderungen der Daten sollten nachvollziehbar gemacht werden, damit die Transparenz und Glaubwürdigkeit der Daten gestärkt und somit das Vertrauen der Nutzerinnen und Nutzer gewonnen werden kann.

[18] Statistical Data and Metadata Exchange; Standard für den Austausch statistischer Daten.

Hinweis: Um Änderungen klar kenntlich zu machen, sollten Ressourcen immer eine Versionsnummer erhalten.

Verlässlichkeit

Um den Nutzerinnen und Nutzern eine Vorstellung davon zu geben, wie verlässlich die Daten sind, empfiehlt es sich, jeder Ressource einen Status zuzuweisen. So wird z.B. ersichtlich, ob die Ressourcen in einer Entwurfsfassung vorliegen und ggf. die Verlässlichkeit geringer einzuschätzen ist.

Hinweis: DCAT-AP.de (siehe Kapitel 8.4) gibt beispielsweise für die Angabe des Reifegrads einer Ressource ein Vokabular vor. Die vorgesehenen Statusangaben können der DCAT-AP.de-Spezifikation entnommen werden.

Verständlichkeit

Die Struktur der Daten und die Benennung von Attributen sollten so gewählt sein, dass Außenstehende diese leicht verstehen können.

Metadaten sollten so befüllt werden, dass Nutzerinnen und Nutzern eine konkrete Vorstellung über den Datensatz erhalten.

Vollständigkeit

Ein Datensatz sollte vollständig sein: Attribute, die zwingend für die Weiternutzung des Datensatzes erforderlich sind, müssen demnach einen Wert enthalten.

Vollständige Metadaten erleichtern die Auffindbarkeit und erleichtern es Nutzerinnen und Nutzern, sich schon früh ein detailliertes Bild bezüglich des bereitgestellten Materials zu bilden.

Hinweis: Die Unvollständigkeit eines Datensatzes stellt kein Hinderungsgrund für die Veröffentlichung dar. Beispielsweise können Daten, die einen Personenbezug oder Betriebs- oder Geschäftsgeheimnisse enthalten, hiervon bereinigt und anschließend veröffentlicht werden. Grundsätzlich gilt: Lieber unvollständige Daten veröffentlichen als gar keine. Hierbei ist jedoch zu beachten, dass die Unvollständigkeit der Daten gekennzeichnet werden muss.

Zugänglichkeit und Verfügbarkeit

Die Ressourcen sollten leicht zugänglich sein. Dazu gehören eine einfache Auffindbarkeit, langlebige Verlinkungen und Referenzen sowie verständliche Beschreibungen des angebotenen Materials.

Hinweis: Um sogenannte „Broken Links" zu vermeiden – also Links, die auf ein nicht mehr erreichbares Ziel verweisen (oftmals erscheint die Fehlermeldung 404 page not found) – empfiehlt es sich, einmal gesetzte **URIs** *nicht mehr zu verändern. Sollte dies unmöglich sein, hilft ein sogenannter »Redirect«, Anfragen zur neuen Adresse weiterzuleiten.*

6 Datenstrukturtypen

Daten können in unterschiedlichen Formaten und Strukturen auftreten. Besonders für die maschinelle Weiterverarbeitung spielt diese »Anordnung« der gespeicherten Werte eine große Rolle. Häufig gibt die Natur der Daten diese Anordnung schon vor. Dennoch ist es sinnvoll, nach Möglichkeit vorher festzulegen, wie Daten idealerweise abgelegt werden sollen. Dabei gibt es nicht immer »richtig« und »falsch«. Die in Tabelle 4 dargestellten Beispiele sollen zur Bestimmung der geeignetsten Struktur als Entscheidungshilfe dienen.

Am einfachsten maschinell zu verarbeiten sind strukturierte Daten, denen ein festes Datenbankschema zugrunde liegt. Beziehungen zwischen den Daten sind festgelegt und sofort ersichtlich. Als Spezialfall von strukturierten Daten eignen sich tabellarisch strukturierte Daten besonders bei Listen. Hierfür bietet sich konkret das in Abschnitt 7.4 behandelte Format JavaScript Object Notation (JSON) an, das durch den Einsatz von JSON-Schema in ein strukturiertes Datenmodell überführt werden kann.

Semistrukturierte Daten enthalten selber Teile der Struktur. Dabei gibt es im Gegensatz zu strukturierten Daten kein den Daten zugrundeliegendes Modell. Ein weit verbreitetes Format zur Darstellung semi-strukturierter Daten ist das in Abschnitt 7.2 vorgestellte Comma Separated Values (CSV).

Das folgende Beispiel soll den Unterschied zwischen strukturierten und semi-strukturierten Daten erläutern. Abbildung 2 zeigt exemplarisch, wie eine Person mit JSON beschrieben werden kann. Aus dem Beispiel ist noch keine semantische Struktur ablesbar. Es ist beispielsweise nicht zu erkennen, dass die Postleitzahl einen eingeschränkten Wertebereich hat. Dies wird erst durch die Betrachtung des Schemas deutlich, siehe Abbildung 3. Darin ist der erlaubte Wertebereich als Integer definiert. Durch diese Information wird die semantische Struktur festgelegt, die strukturierte Daten von semi-strukturierten Daten unterscheidet.

```
{
  "person" : {
    "name" : "Max Mustermann",
    "adresse" : "Krummestrasse 12b",
    "postleitzahl" : 12345,
    "land" : "Deutschland"
  }
}
```

Abbildung 2: Beschreibung einer Person mit JSON.

```
{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "type": "object",
    "properties": {
        "person": {
            "type": "object",
            "properties": {
                "name": {
                    "type": "string"
                },
                "adresse": {
                    "type": "string"
                },
                "postleitzahl": {
                    "type": "integer",
                    "pattern": "/^([0]{1}[1-9]{1}|[1-9]{1}[0-9]{1})[0-9]{3}$/"
                },
                "land": {
                    "type": "string"
                }
            },
            "required": [
                "name",
                "adresse",
                "postleitzahl",
                "land"
            ]
        }
    },
    "required": [
        "person"
    ]
}
```

Abbildung 3: Strukturierung einer Personenbeschreibung mit JSON-Schema.

Daten, die nicht in einer formalisierten Struktur vorliegen, werden unstrukturierte Daten genannt. Aus ihnen lassen sich Modelle zur maschinellen Weiterverarbeitung häufig nur mit unangemessen hohem Aufwand ableiten. Ein Beispiel für unstrukturierte Daten sind elektronische Dokumente mit Fließtext in natürlicher Sprache.

Tabelle 4: Beispiele für unterschiedlich strukturierte Daten.

HERKUNFT	EMPFOHLENE STRUKTUR	GEEIGNETES FORMAT	BEMERKUNG
Wetterdaten	Strukturiert	Gespeichert in relationaler Datenbank	Messergebnisse werden gemäß dem Datenbankschema auf die entsprechenden Tabellen verteilt.
Metadaten	Strukturiert	RDF/XML (mit Schema)	Informationen werden in Elementen gespeichert. Verschachtelung bei mehreren **Distributionen** pro Metadatensatz.
Gesprächsprotokoll	Semistrukturiert	CSV	Eine Aussage pro Zeile. Kommaseparierte Information für Aussage, Person, Thema, Aufgabe etc.
Adressbuch	Semistrukturiert	CSV	Ein Kontakt pro Zeile. Kommaseparierte Information für Name, Adresse, Telefonnummer etc.
Filmmitschnitt	Unstrukturiert	MP4	Der Inhalt (Semantik) von Multimedia-Dateien kann sehr schlecht extrahiert werden.

7 Datenformatspezifische Qualitätsmerkmale und Handlungsempfehlungen

In den vorigen Abschnitten wurden allgemeine Hinweise und Empfehlungen für die Veröffentlichung von qualitativ hochwertigen Daten und Metadaten gegeben. Nachfolgend wird nun eine Auswahl konkreter Datenformate beleuchtet, die u.a. für die Veröffentlichung von Open Data geeignet sind: CSV, JSON, RDF, XML und GeoJSON[19]. Des Weiteren werden REST-Schnittstellen sowie WFS als Möglichkeit des Zugriffs auf Daten behandelt. Jedes dieser Formate wird in einem eigenen Unterkapitel behandelt. Dabei wird auf den Aufbau bzw. die Struktur des Formats eingegangen, konkrete Empfehlungen für die Verwendung gegeben und Links zu hilfreichen Tools und Werkzeugen präsentiert. In einem einführenden Grundlagenkapitel werden zunächst Datenformat-übergreifende Handlungsempfehlungen gegeben.

7.1 Grundlagen

7.1.1 Umgang mit personenbezogenen Daten

Die allgemeinen Datenschutzbedingungen sind bei der Veröffentlichung von Daten stets einzuhalten.

Nicht alle Daten sind für eine Veröffentlichung oder Weitergabe geeignet. Hierzu zählen u.a. personenbezogene Daten oder auch solche, aus denen in Verbindung mit weiteren Daten ein Personenbezug entstehen kann.

Anonymisierungstechniken können dabei helfen, die Daten für eine Veröffentlichung aufzubereiten. Bei solchen Verfahren muss jedoch eine hohe Anonymisierungsqualität gewährleistet werden. Daher empfiehlt es sich, diese vorab von Experten evaluieren zu lassen.

7.1.2 Veröffentlichung von Rohdaten

Es sollten stets **Rohdaten** in einem höchstmöglichen Feinheitsgrad veröffentlicht werden. Hiermit werden die Nachprüfbarkeit der Daten sowie etwaige Möglichkeiten zur Weiterverarbeitung durch Datenanwenderinnen und -anwender gewährleistet.

Zusätzlich zu den Rohdaten können ergänzend auch bearbeitete Daten bereitgestellt werden. Dies gilt beispielsweise für Statistiken oder Berichte, die auf Basis der Rohdaten erstellt wurden, oder für aggregierte Daten. Sie sollten jedoch immer zusammen mit den zugrundeliegenden Rohdaten veröffentlicht werden.

Ausgenommen von der Veröffentlichung als Rohdaten sind selbstverständlich aufgrund von Datenschutz anonymisierte Daten.

[19] Neben der aufgeführten Auswahl können selbstverständlich in Abhängigkeit des Kontexts auch weitere Formate für die Veröffentlichung von Open Data geeignet sein. An dieser Stelle kann jedoch nur eine begrenzte Auswahl an Formaten genauer beleuchtet werden.

Eine Bearbeitung der Daten vor ihrer Veröffentlichung ist auch dann sinnvoll, wenn in den Rohdaten offensichtliche Fehler vorhanden sind, die dem Datenbereitsteller vor der Veröffentlichung auffallen. Auf die Bereinigung der Fehler ist bei der anschließenden Veröffentlichung in den Metadaten hinzuweisen.

7.1.3 Datums- und Zeitangaben

Unabhängig vom Dateiformat sollten Datums- und Zeitangaben stets im ISO-8601-Format[20] (JJJJ-MM-TT) angegeben werden, eventuell unter Angabe der Zeitzone. Als Trenner zwischen Datums- und Zeit wird ein großgeschriebenes T verwendet. Die Zeitzone wird immer von der **Coordinated Universal Time** (UTC) abgeleitet. Handelt es sich direkt um UTC, folgt der Zeitangabe ein großgeschriebenes »Z«. Bei Abweichungen von der UTC wird auf diesen Buchstaben verzichtet und stattdessen die Verschiebung im Format ±hh:mm angegeben. Nachstehend sind dazu zwei Beispiele aufgeführt:

- UTC: 2018-04-25T10:35:00Z
- Abweichung von UTC: 10:35:00-01:00

[20] International Organization for Standardization (ISO): ISO 8601 Date and Time Format. Zuletzt aufgerufen im August 2019 unter https://www.iso.org/iso-8601-date-and-time-format.html.

7.2 CSV-Dateien

7.2.1 Aufbau und Struktur

CSV[21] (Comma Separated Values) ist ein Dateiformat, in dem Werte tabellarisch strukturiert in Form von Textdateien gespeichert werden. Die einzelnen Werte werden durch Trennzeichen voneinander separiert und bilden somit die Spalten der Tabelle. In der obersten Zeile sind die Spaltenüberschriften enthalten. Als Trennzeichen zwischen den einzelnen Spalten und Werten können Kommata, Semikola, Leerzeichen oder auch andere Zeichen dargestellt werden. Der Name CSV legt zwar die Verwendung von Kommata als Trennzeichen nahe, jedoch existiert kein einheitlicher Standard hierfür.

Abbildung 4 visualisiert den Aufbau einer CSV-Datei. Die oberste Zeile enthält die Spaltenüberschriften, die nachfolgenden Zeilen entsprechend der Reihenfolge der Spaltenüberschriften die zugehörigen Werte für die Jahre 2014 bis 2009.

```
Jahr;Besucher;Ansichtszeit pro Besucher;Ansichtszeit pro Seite
2014;768954;00:03:18;00:00:45
2013;822101;00:02:59;00:00:44
2012;792967;00:02:52;00:00:42
2011;721519;00:03:44;00:00:47
2010;707302;00:03:50;00:00:43
2009;429430;00:03:16;00:00:41
```

Abbildung 4: Abrufstatistiken als CSV-Datei mit Spaltenüberschriften, geöffnet in einem Texteditor.

Der tabellarische Charakter der Daten wird stärker erkennbar, sobald die Datei mit einem Tabellenkalkulationsprogramm geöffnet wird (siehe Abbildung 5).

	A	B	C	D	E
1	Jahr	Besucher	Ansichtszeit	Ansichtszeit pro Seite	
2	2014	768954	00:03:18	00:00:45	
3	2013	822101	00:02:59	00:00:44	
4	2012	792967	00:02:52	00:00:42	
5	2011	721519	00:03:44	00:00:47	
6	2010	707302	00:03:50	00:00:43	
7	2009	429430	00:03:16	00:00:41	
8					

Abbildung 5: CSV-Datei geöffnet in einem Tabellenkalkulationsprogramm.

[21] Weitere Informationen unter https://tools.ietf.org/html/rfc4180.

7.2.2 Empfehlungen

Trennzeichen

Bereits im Namen »CSV« ist enthalten, dass Kommata als Trennzeichen für die einzelnen Werte genutzt werden sollten. Dies besagt auch die Spezifikation zum CSV-Format[22]. Jedoch gibt es – wie eingangs erwähnt – keinen einheitlichen Standard, der die Verwendung des Kommas als Trennzeichen vorschreibt. Daher können auch Semikola, Leerzeichen oder andere Zeichen verwendet werden.

Da Kommata und Leerzeichen oftmals Inhalt eines Tabellenwertes sind, wird an dieser Stelle die Verwendung des Semikolons als Trennzeichen empfohlen. Zu beachten ist, dass hinter dem letzten Wert in jeder Zeile kein weiteres Trennzeichen folgt (siehe Abbildung 6).

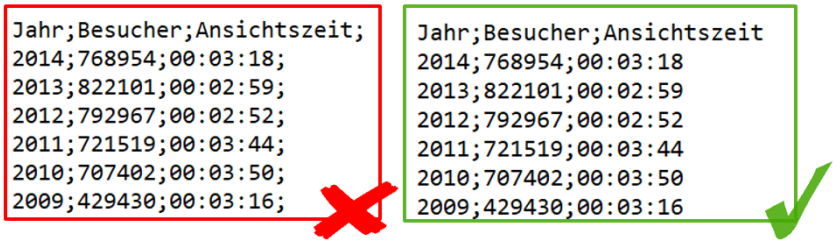


Abbildung 6: Verwendung von Trennzeichen.

Inhalte der Datei

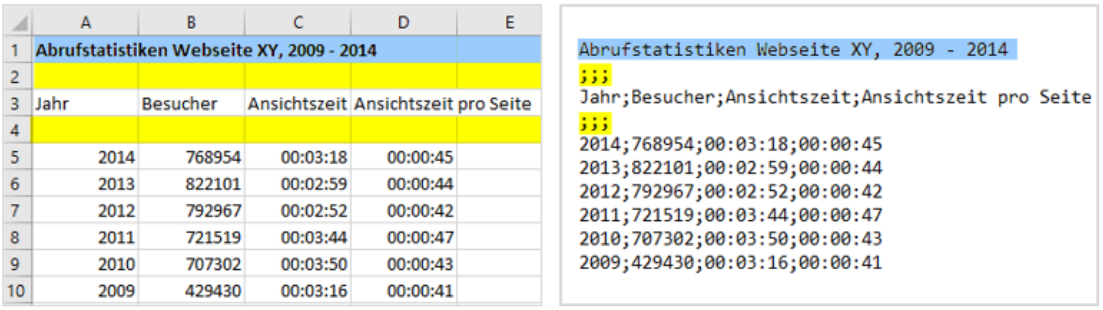
Jede CSV-Datei sollte lediglich eine Tabelle beinhalten. Besteht die zu veröffentlichende Tabelle aus mehreren Blättern, so sollte pro Tabellenblatt eine eigene CSV-Datei erzeugt werden. Andere Strukturierungen würden zwangsläufig zu einem Bruch der Tabellenstruktur führen und die Maschineninterpretierbarkeit gefährden.

Des Weiteren ist darauf zu achten, dass die Datei nur Daten enthält, die zur eigentlichen Tabelle gehören, also Spaltenüberschriften und die Werte aus den entsprechenden Tabellenzellen.

Hintergrund: Oftmals werden tabellarische Daten zusätzlich visuell optimiert, beispielsweise durch das Hinzufügen von Überschriften oder Leerzeilen. Auch wenn die so entstandene Übersichtlichkeit der Tabelle für menschliche Betrachterinnen und Betrachter positiv zu werten ist, sollte bei der Veröffentlichung dringend darauf verzichtet werden. Dies kann zu Schwierigkeiten bei der automatisierten Weiterverarbeitung der Daten führen, denn Leerzeilen und zusätzlich zu den Spaltenüberschriften enthaltene Überschriften werden automatisch interpretiert. Abbildung 7 veranschaulicht dies: Die linke Seite zeigt eine für Menschen übersichtliche Tabelle mit Überschrift (blau markiert) und Leerzeilen (gelb markiert). Die rechte Seite zeigt, wie die gleiche Datei in einem Texteditor aussieht: Die Überschrift (blau) und die

[22] Shafranovich, Y. (2005): Common Format and MIME Type for Comma-Separated Values (CSV) Files, RFC4180. Zuletzt aufgerufen im August 2019 unter https://tools.ietf.org/html/rfc4180.

Leerzeilen (gelb) werden interpretiert. Da dies bei der Weiterverarbeitung zu Fehlern führen kann, sollten neben den Spaltenüberschriften und den eigentlichen Werten keine zusätzlichen Inhalte, wie z.B. Überschriften oder Leerzeilen, enthalten sein.

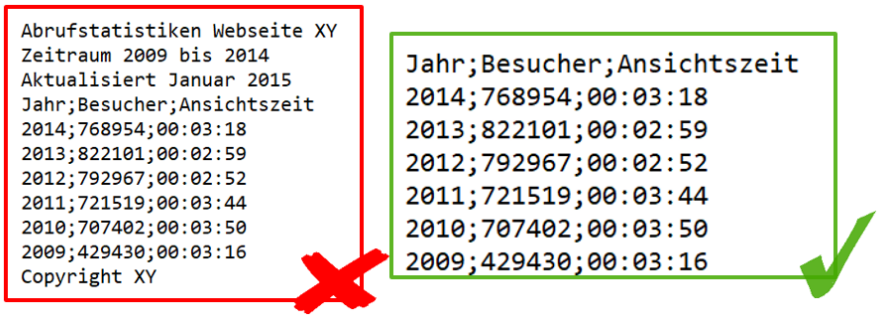


	A	B	C	D	E
1	**Abrufstatistiken Webseite XY, 2009 - 2014**				
2					
3	Jahr	Besucher	Ansichtszeit	Ansichtszeit pro Seite	
4					
5	2014	768954	00:03:18	00:00:45	
6	2013	822101	00:02:59	00:00:44	
7	2012	792967	00:02:52	00:00:42	
8	2011	721519	00:03:44	00:00:47	
9	2010	707302	00:03:50	00:00:43	
10	2009	429430	00:03:16	00:00:41	

```
Abrufstatistiken Webseite XY, 2009 - 2014
;;;
Jahr;Besucher;Ansichtszeit;Ansichtszeit pro Seite
;;;
2014;768954;00:03:18;00:00:45
2013;822101;00:02:59;00:00:44
2012;792967;00:02:52;00:00:42
2011;721519;00:03:44;00:00:47
2010;707302;00:03:50;00:00:43
2009;429430;00:03:16;00:00:41
```

Abbildung 7: Interpretation von Leerzeilen mit Überschriften in CSV-Dateien.

Erläuterungen, Aktualisierungsdatum, Tabellenüberschriften etc. gehören nicht in die CSV-Datei und können ggf. in den Metadaten des resultierenden Datensatzes aufgelistet werden (siehe Abbildung 8). *Hinweis: Nicht Tabellenüberschriften mit Spaltenüberschriften verwechseln. Letztere sollen weiterhin als oberste Zeile vorhanden sein.*



```
Abrufstatistiken Webseite XY
Zeitraum 2009 bis 2014
Aktualisiert Januar 2015
Jahr;Besucher;Ansichtszeit
2014;768954;00:03:18
2013;822101;00:02:59
2012;792967;00:02:52
2011;721519;00:03:44
2010;707402;00:03:50
2009;429430;00:03:16
Copyright XY
```

```
Jahr;Besucher;Ansichtszeit
2014;768954;00:03:18
2013;822101;00:02:59
2012;792967;00:02:52
2011;721519;00:03:44
2010;707402;00:03:50
2009;429430;00:03:16
```

Abbildung 8: Inhalte von CSV-Dateien.

Falls ein Datenexport nicht ohne umfangreiche Erläuterungen auskommt und diese Erläuterungen nicht für jeden Datenexport identisch sind, so sollte statt CSV ggf. ein anderes Format, z.B. JSON oder XML, verwendet werden.

Spaltenüberschriften

Spaltenüberschriften sollten immer in der ersten Zeile der CSV-Datei enthalten sein. Ohne Überschriften ist es für Datenanwenderinnen und -anwender schwierig, die Bedeutung der Daten zu interpretieren. Daher ist es auch wichtig, dass die Spaltenüberschriften so gewählt werden, dass die Bedeutung der zugehörigen Werte eindeutig aus ihr hervorgeht. Sind die Spaltenüberschriften nicht selbsterklärend, sollte eine entsprechende Erläuterung in den Metadaten ergänzt werden. Werkzeuge, wie

Frictionless Data von der Open Knowledge Foundation[23] oder CSV on the Web[24] vom W3C, können hier unterstützen.

Für die Benennung der Spaltenüberschriften sollten lediglich Buchstaben des Alphabets (in Groß- oder Kleinbuchstaben, auch gemischt) und die Ziffern 0 bis 9 verwendet werden. Darüber hinaus können noch Binde- und Unterstriche verwendet werden. Weitere Zeichen sollten generell im gesamten Datensatz vermieden oder ersetzt werden, auch wenn sie Teil der empfohlenen Zeichenkodierung UTF-8 sind, da dies die Abwärtskompatibilität zu älteren Systemen fördert.

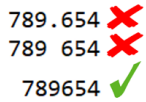
Konsistenz von Spalten und Zeilen

Es ist darauf zu achten, dass die Anzahl der Werte in jeder Zeile mit der Anzahl der Spaltenüberschriften übereinstimmt – jede Zeile sollte also die gleiche Anzahl an Feldtrennern aufweisen.

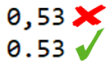
Fehlt ein Wert, so wird dieser automatisch als »null« interpretiert, was bei einer Weiterverwendung der Daten zu Fehlern und Missverständnissen führen kann.

Zahlenwerte

Bei der Speicherung von Zahlenwerten sollten keine Tausender-Stellen durch etwaige Zeichen wie Kommata oder Leerzeichen dargestellt werden.



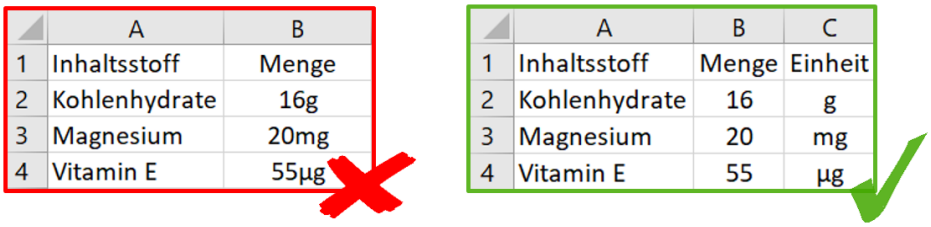
Bei Dezimalzeichen sollte als Trenner immer der amerikanischen Schreibweise mit einem Punkt gefolgt werden, insbesondere dann, wenn das Komma als Trennzeichen zwischen den einzelnen Werten genutzt wird.



Für mehr Kompaktheit und Übersichtlichkeit sorgt auch die Angabe von Zahlenwerten als Ganzzahlen ohne Dezimalstellen, wenn dies möglich ist. Hier ist darauf zu achten, dass pro Spalte die Zahlenwerte einheitlich, also entweder stets als Ganzzahlen oder immer als Dezimalzahlen, dargestellt werden.

Viele Zahlenwerte benötigen die Angabe einer Einheit. Diese sollte stets in der Spaltenüberschrift berücksichtigt werden. Falls die Einheit der Zahlen pro Spalte variiert, sollte diese in einer zusätzlichen Spalte angegeben werden (siehe Abbildung 9).

[23] The Open Knowledge Foundation (o.D.): Frictionless Data. Zuletzt aufgerufen im August 2019 unter https://frictionlessdata.io/.
[24] W3C (2019): CSV on the Web. Zuletzt aufgerufen im August 2019 unter https://w3c.github.io/csvw/primer/.

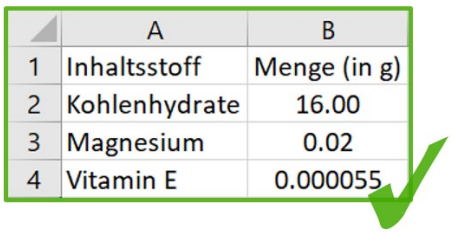


	A	B
1	Inhaltsstoff	Menge
2	Kohlenhydrate	16g
3	Magnesium	20mg
4	Vitamin E	55µg

	A	B	C
1	Inhaltsstoff	Menge	Einheit
2	Kohlenhydrate	16	g
3	Magnesium	20	mg
4	Vitamin E	55	µg

Abbildung 9: Angabe von Zahlenwerten mit verschiedenen Einheiten.

Alternativ ist auch zu prüfen, ob die Zahlenwerte nicht in eine gemeinsame Einheit umgerechnet werden können (siehe Abbildung 10).



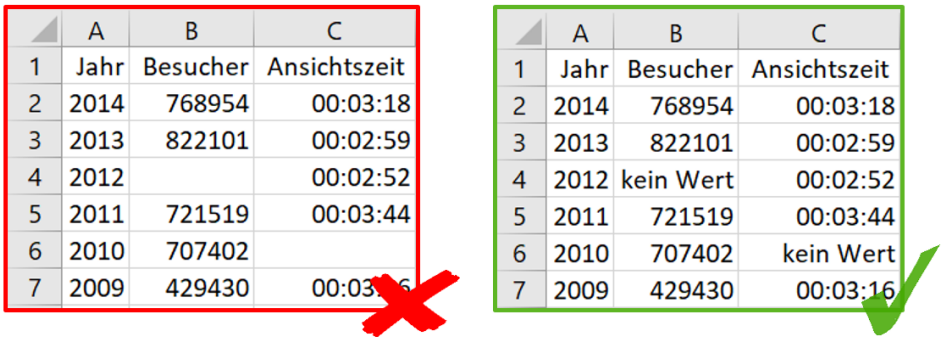
	A	B
1	Inhaltsstoff	Menge (in g)
2	Kohlenhydrate	16.00
3	Magnesium	0.02
4	Vitamin E	0.000055

Abbildung 10: Umrechnung der Zahlenwerte in eine gemeinsame Einheit.

Weiterhin empfiehlt es sich, die verwendeten Einheiten zusätzlich in den Metadaten zu vermerken. Es sollte außerdem auf Konsistenz in der Schreibweise der Einheiten geachtet werden, um die Weiterverarbeitung oder die Zusammenführung mehrerer Tabellen zu vereinfachen. Auch dabei ist eine zusätzliche Beschreibung der Einheiten sowie der verwendeten Schreibweise in den Metadaten sinnvoll.

Null-Werte

Die Abwesenheit eines Wertes, auch **Null-Wert** genannt, sollte durch eindeutige, selbst wählbare Bezeichner angezeigt werden. Dabei ist es wichtig, diese Bezeichner in den Metadaten zu hinterlegen. Beispiele sind »kein Wert« oder »null«.



	A	B	C
1	Jahr	Besucher	Ansichtszeit
2	2014	768954	00:03:18
3	2013	822101	00:02:59
4	2012		00:02:52
5	2011	721519	00:03:44
6	2010	707402	
7	2009	429430	00:03:16

	A	B	C
1	Jahr	Besucher	Ansichtszeit
2	2014	768954	00:03:18
3	2013	822101	00:02:59
4	2012	kein Wert	00:02:52
5	2011	721519	00:03:44
6	2010	707402	kein Wert
7	2009	429430	00:03:16

Abbildung 11: Kenntlichmachung von Null-Werten.

Zeichenkodierung

Schriftzeichen können von Programmen auf unterschiedliche Art und Weise abgespeichert werden. Um größtmögliche Kompatibilität zu anderen Programmen zu gewährleisten, sollten Dateien immer der **UTF-8-Zeichenkodierung** folgen. Andernfalls können Probleme bei der maschinellen Verarbeitung entstehen. Je nach Programm muss UTF-8 explizit im »Speichern unter«-Dialog aktiviert werden. Libre Office Calc beispielsweise lässt beim Speichern einer CSV-Datei neben dem Trennzeichen auch die Zeichenkodierung auswählen. Eine Anleitung für das Speichern als UTF-8 in Excel ist in Abschnitt 7.2.3 aufgeführt. Falls ein anderer Zeichensatz genutzt wird, ist dies unbedingt in den Metadaten des resultierenden **Datensatzes** mitzuteilen.

Maskierung

Enthalten Werte Zeichen, die identisch zum Feldtrenner sind (dieses Dokument empfiehlt ein Semikolon als Feldtrenner), müssen diese in doppelte Anführungszeichen gesetzt werden. Dieser Vorgang wird Maskierung genannt und verhindert die Interpretation dieser Werte. Beispielsweise erhält ein maskiertes Semikolon keine besondere Bedeutung – im Gegensatz zu einem nicht maskierten Semikolon, welches als Feldtrenner interpretiert wird und somit eine weitere Spalte erzeugen würde.

Die folgenden Zeichen müssen maskiert werden:

- Feldtrenner
- Kommata
- Semikola
- Zeilenumbrüche
- Doppelte Anführungszeichen
- Backslash ****.

Zur Maskierung einzelner Zeichen statt einer ganzen Zeichenkette wird das Symbol **** verwendet, auch Backslash genannt. Innerhalb einer Zeichenfolge vorkommende Anführungszeichen werden dementsprechend mithilfe des Konstrukts **\"** von der Interpretation eines Programmes ausgenommen. Um zu verhindern, dass ein inhaltlich relevanter Backslash wie ein Maskierungszeichen behandelt wird, muss dieser selbst maskiert werden, indem ein weiterer Backslash eingefügt wird: ****.

Abbildung 12: Korrekter Einsatz von Maskierung bei der Verwendung eines Semikolons.

Nachstehend ist derselbe Satz exemplarisch in Rohform, als Ganzes sowie teilmaskiert dargestellt:

Maskierung erlaubt Anführungszeichen ("), das Semikolon (;) sowie den Backslash (\).

"Maskierung erlaubt Anführungszeichen (\"), das Semikolon (;) sowie den Backslash (\)."

Maskierung erlaubt Anführungszeichen ("), das Semikolon (\;) sowie den Backslash (\\).

Zuletzt sollten Werte vor dem Speichern getrimmt werden. Dabei werden überflüssige Leerzeichen am Anfang und Ende einer Zeichenkette entfernt.

7.2.3 Microsoft Excel

Bearbeiten von CSV im Programm Excel

Häufig werden CSV-Daten im Programm Excel bearbeitet, das dafür eine entsprechende Unterstützung enthält. Je nach Konfiguration werden diese im »Öffnen« -Fenster nicht immer angezeigt. In diesen Fällen muss der Menüpunkt »Textdateien« ausgewählt werden (Abbildung 13).

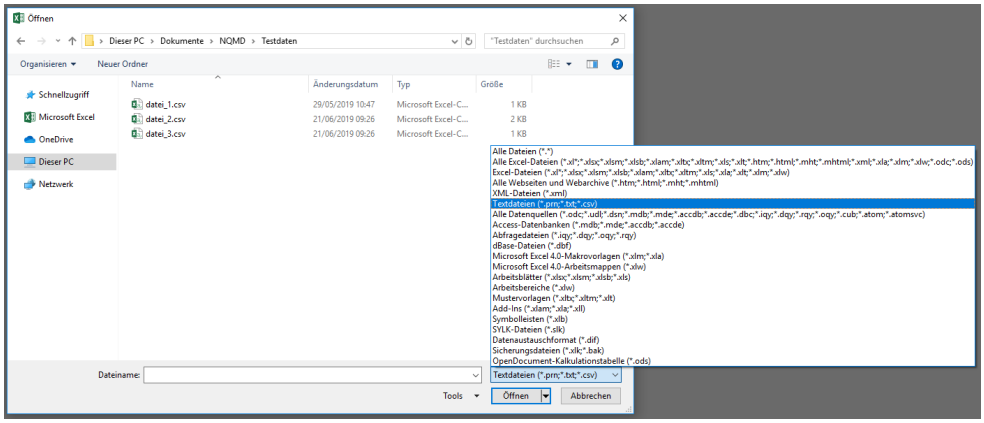


Abbildung 13: CSV-Dateien werden im »Öffnen-«Fenster von Excel nicht immer angezeigt.

Abhängig vom verwendeten Trennzeichen erscheint nach dem Öffnen einer CSV-Datei der gesamte Zeileninhalt in einer Spalte, was zu einer geringen Lesbarkeit führt. Nach Markierung der gesamten Spalte kann die Darstellung durch die Verwendung der »Text-in-Spalten« -Schaltfläche im Reiter »Daten« angepasst werden. Im anschließend erscheinenden Dialog kann nun das bevorzugte Trennzeichen angegeben werden.

Speichern von Excel-Dateien als CSV

Wenn tabellarische Daten im Programm Excel erstellt werden, müssen diese anschließend in korrektes CSV exportiert werden. Standardmäßig verwendet Excel zum Trennen von Werten Kommata. Um dies zu Semikola zu ändern, wie es auch in diesem Dokument empfohlen wird, muss zunächst die Windows Systemsteuerung geöffnet werden. Nach Auswahl des Menüpunktes »Datums-, Uhrzeit- oder Zahlenformat ändern« der Kategorie »Zeit und Region« öffnet sich ein neues Fenster. Unter »Weitere Einstellungen« kann anschließend das »Listentrennzeichen« festgelegt werden. Dieser Vorgang ist in Abbildung 14 dargestellt.

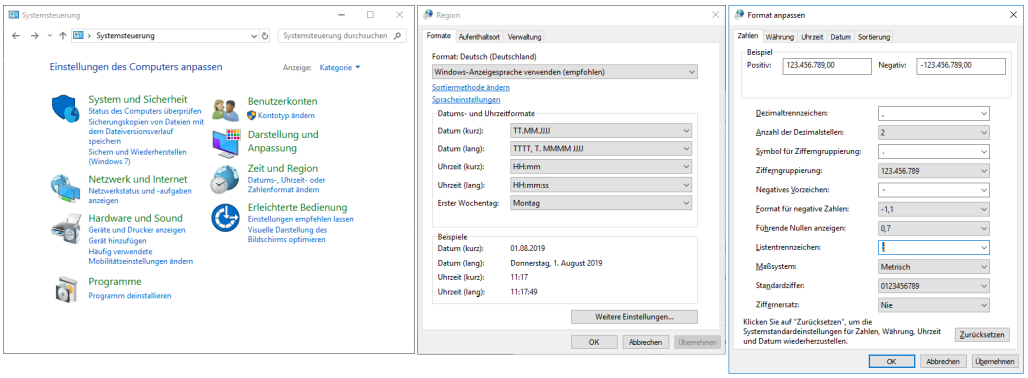


Abbildung 14: Ändern der beim CSV-Export verwendeten Trennzeichen.

Für größtmögliche Kompatibilität ist die Verwendung der UTF-8 Kodierung wichtig. Dies kann im Dialog »Speichern unter« festgelegt werden. Dort befindet sich ein ausklappbares Menü, in dem »Weboptionen« ausgewählt werden muss. Im erscheinenden Fenster ist dann im Reiter »Codierung« der Wert »Unicode (UTF-8)« auszuwählen. Beim Speichern ist weiterhin darauf zu achten, dass als Dateityp »CSV (Trennzeichen-getrennt) (*.csv)« ausgewählt ist. Abbildung 15 veranschaulicht dies.

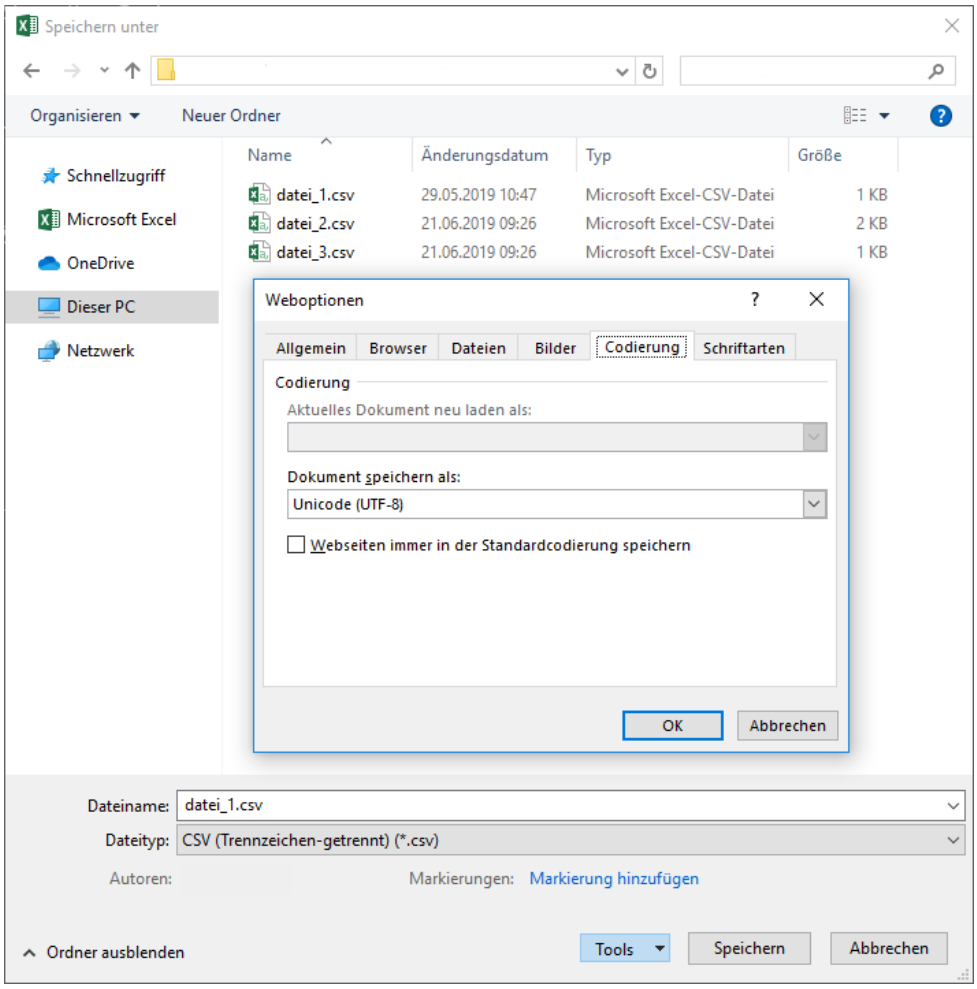


Abbildung 15: Speichern als UTF-8 in Excel.

7.2.4 Nützliche Links

Beschreibung	Link
CSV-Spezifikation	https://tools.ietf.org/html/rfc4180
Sammlung hilfreicher Tools für CSV, bspw. für das Konvertieren in andere Formate oder das Zuschneiden von Zeilen und Spalten	https://onlinecsvtools.com/
Webseite, die CSV-Dateien auf Konsistenz und Konformität prüft	https://csvlint.io/
Anwendung zum Bearbeiten von CSV-Dateien	https://de.libreoffice.org/ https://csved.sjfrancke.nl

7.3 XML-Dateien

XML (Extensible Markup Language) ist ein Format zur hierarchischen Strukturierung von Daten. Das Ergebnis ist eine baumartige Struktur, die sowohl gut maschinell zu verarbeiten als auch von Menschen lesbar ist. XML wurde entwickelt, um plattformübergreifend Daten austauschen zu können, speziell auch über das Internet.[25]

7.3.1 Aufbau und Struktur

Zur Strukturierung stehen zwei Optionen zur Auswahl. Daten können entweder innerhalb von sogenannten **Tags** eingefasst oder mithilfe von **Attributen** definiert werden. Tags kommen paarweise zum Einsatz, das heißt es gibt einen öffnenden und einen schließenden Tag. Bei ersterem ist der Bezeichner in spitzen Klammern eingefasst, bei letzterem kommt zusätzlich hinter die öffnende Klammer ein Slash („/“). Abbildung 16 veranschaulicht dies.

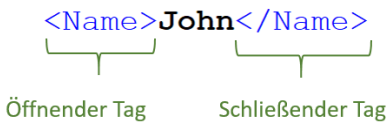


Abbildung 16: Beispiel für einen öffnenden und schließenden Tag in XML.

Eine Einheit aus öffnendem und schließendem Tag wird **Element** genannt. Elemente strukturieren die XML-Datei. Zusatzinformationen zu Elementen werden über Attribute angegeben. Attribute werden in der Form *Bezeichner="Wert"* in den öffnenden Tag geschrieben. Die Reihenfolge ist dabei irrelevant, es können also sowohl Elemente sowie Attribute beliebig sortiert werden. In Abbildung 17 ist ein Beispiel für ein Element mit Attribut abgebildet.

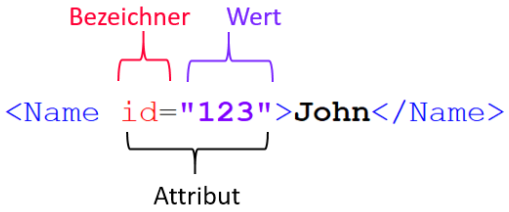


Abbildung 17: Beispiel für die Angabe eines Attributs in XML.

XML-Dateien besitzen genau ein **Wurzelelement** (in Abbildung 18 ist das Wurzelelement `obstsorten`). Ein Element darf nicht mehrere Attribute mit demselben Namen haben. Bei Bezeichnern wird zwischen Groß- und Kleinschreibung unterschieden, Leerzeichen sind nicht erlaubt. Es wird empfohlen, jede Datei mit einer **Deklaration** einzuleiten. Dabei handelt es sich um eine Zeile, die den verwendeten

[25] Weitere Informationen unter https://www.w3schools.com/xml/.

XML-Standard und die Kodierung definiert. Abbildung 18 zeigt beispielhaft den Aufbau einer XML-Datei mit Deklaration in der obersten Zeile.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<obstsorten>
  <frucht id="1">
    <typ>Apfel</typ>
    <herkunft>Deutschland</herkunft>
    <kernobst>true</kernobst>
  </frucht>
  <frucht id="2">
    <typ>Traube</typ>
    <herkunft>Italien</herkunft>
    <kernobst>false</kernobst>
  </frucht>
</obstsorten>
```

Abbildung 18: Beispiel einer XML-Datei mit Deklaration, Tags und Attributen.

Daten können sowohl aus kleinen Informationenteilen bestehen, wie beispielsweise IDs, als auch aus langen Textabschnitten, etwa Beschreibungstexten. Für letzteren Fall besteht die Möglichkeit, Inhalte von der XML-Struktur »auszunehmen«. Texte innerhalb eines so deklarierten Abschnitts werden nicht vom verarbeitenden Programm interpretiert, können also auch selbst XML-Tags enthalten. Dafür muss der betreffende Abschnitt in den Zeichenfolgen *<![CDATA[“ und „]]>* eingefasst werden (siehe Abbildung 19). Auch für vorformatierte Texte ist diese Option geeignet.

```
<frucht id="1">
  <typ>Apfel</typ>
  <beschreibung>
    <![CDATA[
      Das ist ein Beispiel.
      <ignoriertesElement>Ignorier mich!</ignoriertesElement>
    ]]>
  </beschreibung>
</frucht>
```

Abbildung 19: Verwendung von CDATA.

7.3.2 Empfehlungen

Validierung

Online-Validatoren helfen bei der Überprüfung von XML-Dokumenten auf syntaktische Fehler.

Um darüber hinaus die Einhaltung inhaltlicher Regeln, wie etwa erlaubte Datentypen oder dem Vorhandensein bestimmter Elemente, sicherzustellen, gibt es spezielle Schemata, die den verarbeitenden Programmen eine Validierung der Struktur erlauben. Eine solche Datei besteht selbst aus XML und wird auch XSD (XML Schema Definition) genannt. Hier wird genau spezifiziert, welche Elemente/Attribute erlaubt sind und

welchen Datentyp der Inhalt haben muss. Auch möglich ist die Angabe von Mustern, um schon bei der Validierung die Korrektheit von Datenformaten, wie beispielsweise Postleitzahlen, zu prüfen.

Benennung und Schreibweise von Bezeichnern

Um die Lesbarkeit sowie das Verarbeiten von XML-Dateien zu erleichtern, haben sich einige Konventionen durchgesetzt. Zunächst sollten alle Bezeichner, seien es Tags oder Attribute, aussagekräftige Namen haben und idealerweise nicht doppelt verwendet werden. Bezüglich der Schreibweise der Bezeichner gibt es keine offiziellen Empfehlungen, es kann also beispielsweise **camelCase** oder **PascalCase** verwendet werden. Allerdings sollten dabei nicht mehrere Formen miteinander vermischt werden. Weiterhin sollte auf Sonderzeichen in den Bezeichnern verzichtet werden.

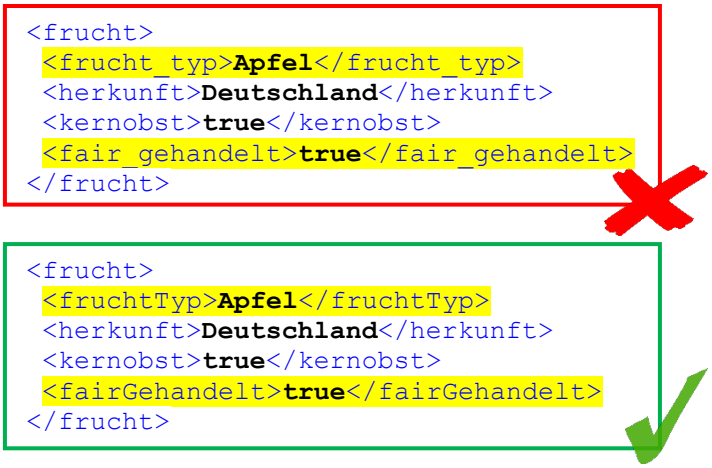
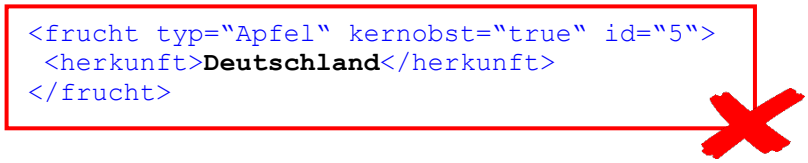


Abbildung 20: Einsatz von camelCase in XML.

Elemente und Attribute

Auch hinsichtlich der Frage, ob Daten in Elementen oder Attributen kodiert werden sollten, gibt es keine verpflichtende Richtlinie. Informationen, die Teil der eigentlichen Daten sind, sollten mit Elementen abgebildet werden. Metadaten hingegen, die zusätzliche Informationen enthalten, sollten stattdessen als Attribute realisiert werden. Beispielsweise ist in Abbildung 21 die »id« ein Teil der Metadaten und somit ein Attribut eines Elementes vom Typ »frucht«.



```
<frucht id="5">
 <typ>Apfel</typ>
 <herkunft>Deutschland</herkunft>
 <kernobst>true</kernobst>
</frucht>
```

Abbildung 21: Beispiel zu Elementen und Attributen in XML.

Null-Werte

Analog zu CSV sollte die Abwesenheit von Werten explizit gekennzeichnet werden. Bei Elementen sollte das selbige dafür leer gelassen und ein spezielles Attribut gesetzt werden.[26] Zur besseren Lesbarkeit können leere Elemente auch direkt geschlossen werden (<*beispiel* />). Andere potenziell zum Null-Element gehörenden Attribute können trotzdem gesetzt sein. Sollten einzelne Attribute keinen Wert haben, können diese ersatzlos weggelassen werden. Abbildung 22 veranschaulicht die besprochenen Konzepte am Beispiel des Elements *Beschreibung*.

```
<obst>
  <frucht id="1" >
    <typ>Apfel</typ>
    <beschreibung sprache="deutsch" >
      Sehr lecker!
    </beschreibung>
  </frucht>
  <frucht id="2" aktualisiert="2019-01-01" >
    <typ>Birne</typ>
  </frucht>
</obst>
```

```
<obst>
  <frucht id="1" >
    <typ>Apfel</typ>
    <beschreibung sprache="deutsch" >
      Sehr lecker!
    </beschreibung>
  </frucht>
  <frucht id="2" aktualisiert="2019-01-01" >
    <typ>Birne</typ>
    <beschreibung xsi:nil="true" sprache="unbekannt" />
  </frucht>
</obst>
```

Abbildung 22: Null-Werte in XML.

[26] W3C (2004): XML Schema Part 1: Structures Second Edition. Zuletzt aufgerufen im August 2019 unter https://www.w3.org/TR/xmlschema-1/#xsi_nil.

Plattformunabhängigkeit

In jedem Fall sollte die Datenstruktur programmunabhängig sein. Das bedeutet, dass keine Elemente oder Attribute existieren, die von den eigentlichen Informationen losgelöste Daten enthalten. Dabei kann es sich etwa um Versionsnummern der verarbeitenden Programme handeln, die für die Weiterverarbeitung keine Rolle spielen.

Versionierung

Da sich Daten und dementsprechend auch deren Darstellung immer wieder ändern können, ist es sinnvoll, an das Wurzelelement einer XML-Datei eine Versionsnummer als Attribut anzuhängen. Auf diese Weise können verarbeitende Programme sofort bestimmen, welche Struktur zu erwarten ist und welche Elemente beziehungsweise Attribute zulässig sind.

Struktur

Sollten sehr große Datenmengen im XML-Format gespeichert werden, empfiehlt es sich, die Daten aufzuteilen und in mehreren Dateien abzulegen. Das verbessert vor allem die Verarbeitungsgeschwindigkeit der eingesetzten Programme. Grundsätzlich sollten bei der Aufteilung der Datenmengen domänenspezifische Standards berücksichtigt werden.

7.3.3 Nützliche Links

Beschreibung	Link
XML-Spezifikation	https://www.w3.org/TR/2006/REC-xml11-20060816/
Sammlung von Tools für XML	https://onlinexmltools.com/
XML-Beautifier	http://xmlbeautifier.com/

7.4 JSON-Dateien

7.4.1 Aufbau und Struktur

JSON (JavaScript Object Notation) ist ein Format zur Beschreibung strukturierter Daten. Im Vergleich zu XML positiv zu bewerten ist die einfachere Schreib- und Lesbarkeit von JSON-Dateien, die unter anderem durch den Verzicht auf die Unterscheidung zwischen Elementen und Attributen entsteht, sowie die geringere Menge an **Verwaltungsdaten**. Damit sind Daten gemeint, die keinen Mehrwert besitzen, aber zum Transport der eigentlichen Daten unerlässlich sind. Als Vergleich könnte ein Paket mit zerbrechlichem Inhalt herangezogen werden, bei dem Verpackung sowie Polstermaterial den Verwaltungsdaten entsprächen. Es existieren zwar bisher nur wenige Validierungsmethoden für JSON; hieran wird jedoch gearbeitet (Stand 2018).[27]

Erlaubte Datentypen sind die Folgenden:

- Null-Wert (Abwesenheit eines Wertes), dargestellt durch das Schlüsselwort *null*
- Wahrheitswerte, entweder *true* oder *false*
- Zeichenfolgen, wobei hier die Maskierung einzelner Zeichen genauso wie bei CSV-Dateien funktioniert
- Zahlen, einfache Folgen der Ziffern 0 bis 9, optional mit Vorzeichen und Dezimalpunkt
- Listen, auch *Arrays* genannt, werden in eckigen Klammern eingefasst, die einzelnen Elemente durch Kommata getrennt. Listen können auch leer sein.

Objekte werden in geschweiften Klammern erfasst und enthalten beliebig viele kommagetrennte Schlüssel-Wert-Paare. Abbildung 23 zeigt beispielhaft eine JSON-Datei mit verschiedenen Datentypen.

```
[{
    "typ": "Apfel",
    "menge": 3,
    "bio": true,
    "sorten": ["Granny Smith", "Elstar"]
  },
  {
    "typ": "Traube",
    "menge": 5,
    "bio": false,
    "sorten": ["Vitis amurensis"]
  }
]
```

Abbildung 23: Beispiel einer JSON-Datei mit verschiedenen Datentypen.

[27] Weitere Informationen unter http://json-schema.org.

7.4.2 Empfehlungen

Kodierung

Bezeichner sowie Zeichenketten müssen immer mit doppelten Anführungszeichen eingefasst sein, wie in den Beispielen angegeben. Außerdem sollten nach Möglichkeit die vorhandenen Datentypen verwendet, also keine Zahlen oder Wahrheitswerte als Zeichenketten gespeichert werden (siehe Abbildung 24).

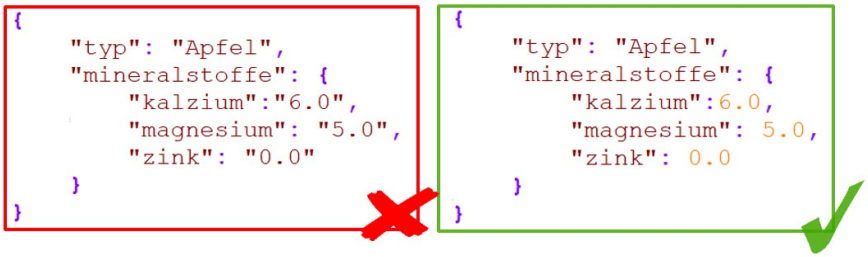


Abbildung 24: Verwendung vorhandener Datentypen.

Weiterhin ist eine Gruppierung von Daten entsprechend deren Zusammengehörigkeit empfehlenswert, da hierdurch die Les- und Verarbeitbarkeit erleichtert wird (siehe Abbildung 25).

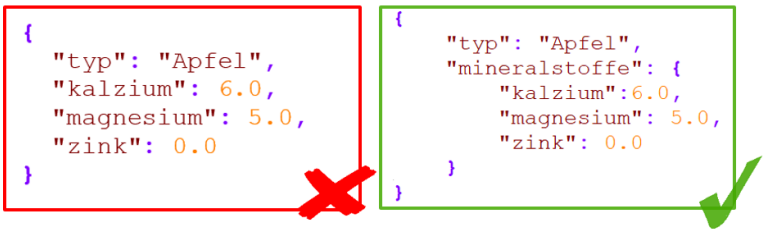


Abbildung 25: Gruppierung von Daten.

Streaming

Für die Übertragung der Daten per **Stream** ist JSON/GeoJSON bedingt geeignet. Da jeder (Geo)JSON-Datensatz einen geschlossenen Block bildet (öffnende und schließende Klammern), können JSON-Daten nicht in kleinere Blöcke zerteilt und einzeln übertragen werden, da bei einer einfachen Zerteilung die kleineren Blöcke nicht alle schließenden Klammern enthalten würden. Mit *ndjson* existiert jedoch eine Streaming-optimierte Variante von JSON.

7.4.3 Nützliche Links

Beschreibung	Link
Einführung und Erklärung zum Aufbau von JSON-Dateien	https://json.org/
Sammlung hilfreicher Tools für JSON-Dateien, bspw. für die Konvertierung in andere Formate	https://onlinejsontools.com/
Streaming-optimierte Variante von JSON	http://ndjson.org/

7.5 Geo-JSON

7.5.1 Aufbau und Struktur

Aufbauend auf dem JSON-Format bietet das GeoJSON-Format die zusätzliche Möglichkeit, geografische Informationen zu beschreiben. Die Beschreibung von geografischen Informationen richtet sich nach der **Simple-Feature-Access**-Spezifikation[28].

Die Geometrien bzw. deren Koordinaten werden dabei durch die Angabe von Längengrad und Breitengrad beschrieben. Die Reihenfolge von Längengrad und Breitengrad ist fest und kann nicht vertauscht werden. Optional kann jeder Koordinate noch ein Höhenwert beigefügt werden. Ebenfalls optional ist die Angabe einer **Bounding Box**, die das Feature umschließt.

Die Angabe der Koordinaten erfolgt im **Referenzsystem WGS-84**[29]. Die Verwendung dieses Referenzsystems ist verbindlich und die Verwendung von alternativen Referenzsystemen demnach nicht erlaubt.

Geometrien, die ursprünglich (abhängig vom Format) in ihrer Definition Kreisbögen und Kurven enthalten, müssen durch eine Abfolge von interpolierten Kanten ersetzt werden, da Bögen und Kurven nicht vom GeoJSON-Format unterstützt werden.

Erlaubte Geometrietypen sind:

- Punkt
- Linie
- Polygon
- MultiPunkt
- MultiLinie
- MultiPolygon

Abbildung 26 zeigt Beispiele für die Darstellung eines Punktes, einer Linie, Polygon und MultiLinie:

[28] Weitere Informationen unter https://www.opengeospatial.org/standards/sfa.
[29] Weitere Informationen unter http://de.dwhwiki.info/konzepte/geo-daten/wgs84.

```
{
    "type": "Point",
    "coordinates": [0,0]
}

{
    "type": "LineString",
    "coordinates": [[0,0],[10,10]]
}

{
    "type": "Polygon",
    "coordinates": [[[0,0],[10,10],[10,0],[0,0]]]
}


{
    "type": "MultiLineString",
    "coordinates": [[[170.0,45.0],[180.0,45.0]],[[-180.0,45.0],[-170.0,45.0]]]
}
```

Abbildung 26: Beispiele für die Darstellung verschiedener Geometrietypen.

Wird die reine Beschreibung der Geometrie um ergänzende Informationen erweitert, z.B. Namen oder Angaben von IDs, so bilden Geometrie und Sachdaten zusammen jeweils ein »Feature«.

Abbildung 27 zeigt ein Beispiel für ein »Feature«, welches Sachdaten und eine Geometrie enthält:

```
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [10.985365, 47.421066]
  },
  "properties": {
    "name": "Zugspitze",
    "bundesland": "Bayern",

  }
}
```

Abbildung 27: Beispiel für ein Feature mit Sachdaten und Geometrie.

Sollen mehrere Features zusammengefasst werden, wird der Typ *FeatureCollection* verwendet. Wenn keine Attribute benötigt werden und die reine gruppenweise Beschreibung von Geometrien ausreicht, kann der Typ *GeometryCollection* verwendet werden. Darüber hinaus können diese Datentypen wiederum kombiniert und verschachtelt werden.

Um ein Objekt der echten Welt zu beschreiben, reichen häufig einfache Geometrietypen nicht aus. In diesen Fällen greift man auf die Multigeometrien zurück oder, wenn verschiedene Geometrietypen benötigt werden, auf die *GeometryCollection*. So kann beispielsweise eine Stadt als einfacher (Mittel-)Punkt modelliert werden und zusätzlich das Stadtgebiet (im Beispiel die Bounding Box) als

Polygon. Beide Geometrien können dann als *GeometryCollection* innerhalb eines Features zusammengefasst werden.

Abbildung 28 zeigt ein Beispiel für eine *FeatureCollection*: Zwei Feature werden zu einer *FeatureCollection* zusammengefasst:


```
{
    "name": "Die beiden höchsten Berge Deutschlands",
    "type": "FeatureCollection",
    "features": [{
        "type": "Feature",
        "geometry": {
            "type": "Point",
            "coordinates": [10.985365,47.421066]
        },
        "properties": {
            "name": "Zugspitze",
            "bundesland": "Bayern"
        }
    },
    {
        "type": "Feature",
        "geometry": {
            "type": "Point",
            "coordinates": [11.054167,47.395833]
        },
        "properties": {
            "name": "Hochwanner",
            "bundesland": "Bayern"
        }
    }]
}
```


Abbildung 28: Beispiel für eine FeatureCollection.

Abbildung 29 zeigt ein Beispiel für ein Feature mit einer *GeometryCollection*. Die *GeometryCollection* enthält zwei unterschiedliche Geometrietypen:

```
{
  "type": "Feature",
  "geometry": {
    "type": "GeometryCollection",
    "geometries": [{
      "type": "Point",
      "coordinates": [7.6261347, 51.9606649]
    }, {
      "type": "Polygon",
      "coordinates": [[[7.774345,52.060022],[7.473816,52.060022],
        [7.473816,51.840152],[7.774345,51.840152],[7.774345,52.060022]]]
    }]
  },
  "properties": {
    "name": "Münster i. Westf."
  }
}
```

Abbildung 29: Beispiel für ein Feature mit einer GeometryCollection.

7.5.2 Empfehlungen

Koordinatenreihenfolge

Obwohl es für die reine Geometriedarstellung einer Linie ohne Bedeutung ist, in welcher Richtung die Koordinaten angegeben werden, spielt dies bei der Interpretation durch Tools und Werkzeuge eine Rolle. Viele Tools interpretieren die Reihenfolge als »Fließrichtung«, beispielsweise bei der Darstellung von Flüssen.

Die GeoJSON-Spezifikation aus dem Jahr 2016[30] enthält eine Regel für Polygone: Der äußere Umring sollte gegen den Uhrzeigersinn angegeben werden und innere Polygone (Löcher) sollten im Uhrzeigersinn verlaufen. Da es in einer älteren Spezifikation keine Angabe über die Reihenfolge gab, sind viele Tools in der Lage, auch mit Polygonen umzugehen, welche diese Regel nicht beachten. Trotzdem kann es sein, dass einige Werkzeuge Probleme bei der Darstellung haben oder manche Algorithmen (z.B. für die Flächenberechnung) langsamer laufen, wenn diese Regel nicht beachtet wird.

Antimeridian

Beim Umgang mit Koordinaten am sogenannten Antimeridian, also in dem Bereich, in welchem der Wechsel von +180° zu -180° stattfindet, kann es zu Fehlinterpretationen des Geometrieverlaufs kommen. Verläuft eine Linie von 170° nach -170°, kann eine Software nicht entscheiden, ob die Geometrie einmal rund um den Globus verläuft oder über den Antimeridian und dadurch der Vorzeichenwechsel zustande kommt.

[30] Weitere Informationen unter https://tools.ietf.org/html/rfc7946.

Die Empfehlung ist in diesem Fall, eine zusätzliche Koordinate direkt auf dem Antimeridian anzulegen. Dabei wird diese Position doppelt angegeben, einmal mit positivem und einmal mit negativem Vorzeichen, wie in Abbildung 30 dargestellt.

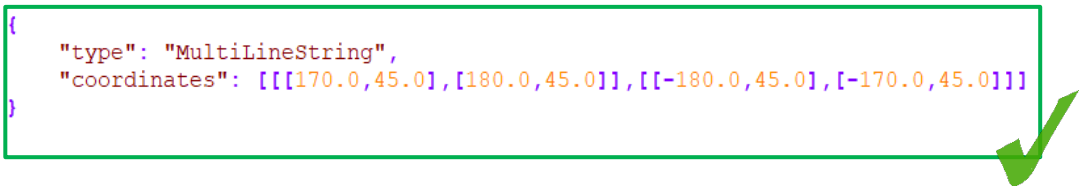


Abbildung 30: Beispiel für die Angabe von Koordinaten am Antimeridian.

Performance

Da die Spezifikation keine Vorgabe macht, wie komplex eine Geometrie sein darf (wie viele Stützpunkte verwendet werden sollen), kann der Textblock einer Geometrie sehr groß werden. Es gibt keine Vorgabe über die Anzahl der Nachkommastellen, also wie genau eine Koordinate angegeben werden soll. Auch hier kann die Verwendung von sehr vielen Nachkommastellen die Größe der Datei massiv beeinflussen und außerdem zu Fehlinterpretationen führen: So beschreibt eine Koordinate mit sechs Dezimalstellen die Lage eines Messpunktes auf ca. 11 cm genau. Die Anzahl der Dezimalstellen sagt jedoch nichts über die Messgenauigkeit aus. So könnte durch Messfehler z.B. eine Koordinate mit sechs Dezimalstellen einen größeren Versatz zur realen Position auf der Erde aufweisen als eine exakt gemessene Koordinate mit fünf Dezimalstellen. Umgekehrt bedeuten weniger Dezimalstellen nicht, dass die Lagegenauigkeit der Daten »schlecht« ist.

Wie viele Nachkommastellen und wie viele Stützpunkte bei einer Geometrie nötig sind, hängt immer vom Verwendungszweck und dem gewünschten Detailgrad ab. Auch die Performance von Darstellung und Berechnungen sollte bedacht und an den Zweck angepasst werden. Grundsätzlich sollten so wenig Nachkommastellen und Stützpunkte wie möglich verwendet werden. Überflüssige Leer-Attribute sind zu vermeiden.

Streaming

Für die Übertragung der Daten per **Stream** ist JSON/GeoJSON bedingt geeignet. Da jeder (Geo)JSON-Datensatz einen geschlossenen Block bildet (öffnende und schließende Klammern), können JSON-Daten nicht in kleinere Blöcke zerteilt und einzeln übertragen werden, da bei einer einfachen Zerteilung die kleineren Blöcke nicht alle schließenden Klammern enthalten würden. Mit *ndjson* exisitiert jedoch eine Streaming-optimierte Variante von JSON.

Topologie

Mit GeoJSON können keine Topologien dargestellt werden. Wenn Topologien dennoch verwendet werden sollen, muss auf das TopoJSON-Format zurückgegriffen werden. TopoJSON ist von der Dateigröße kleiner als GeoJSON; das Format ist jedoch komplexer als GeoJSON.

7.5.3 Nützliche Links

Beschreibung	Link
Online Tool zur Validierung und grafischen Darstellung von GeoJSON-Daten	http://geojson.io
Die wichtigsten Fakten zu GeoJSON mit anschaulichen Beispielen (englisch)	https://macwright.org/2015/03/23/geojson-second-bite.html
Übersicht über Formate, die Koordinaten in lat/lon verwenden	https://macwright.org/lonlat/
GeoJSON-Einführung, mit Beispielen (deutsch)	https://blog.codecentric.de/2018/03/geojson-tutorial/
Übersicht über die verschiedenen Geometrietypen	https://de.wikipedia.org/wiki/GeoJSON
Übersicht über das TopoJSON-Format	https://github.com/topojson/topojson/wiki

7.6 RDF-Dateien

7.6.1 Aufbau und Struktur

RDF (Resource Description Framework) ist am ehesten als Modell bzw. System zur Speicherung von Daten und Metadaten zu verstehen. Der Fokus liegt dabei weniger auf guter Lesbarkeit für Menschen, sondern auf der Verknüpfung zusammenhängender Daten. Diese Verknüpfungen werden **Relationen** oder auch **Tripel** genannt. Sie folgen immer dem Muster »*Subjekt-Prädikat-Objekt*«. Der Aufbau lehnt sich damit an die natürliche Sprache an. Abbildung 31 verdeutlicht diese Struktur anhand eines Beispiels.

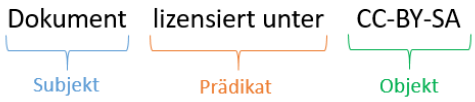


Abbildung 31: Beispiel für ein Tripel.

In dem Beispiel besteht das Prädikat *lizensiert unter* aus zwei Wörtern, was allerdings nur der besseren Lesbarkeit dient und keinen Einfluss auf die Struktur hat.

RDF unterscheidet zwischen **Ressourcen** und **Literalen**. Während es sich bei ersterem um eindeutige, wiederverwendbare Informationen handelt, sind mit letzteren einfache Zeichenketten gemeint, die nicht weiter mit anderen Ressourcen verknüpft werden. Subjekt und Prädikat sind immer Ressourcen, Objekte hingegen können entweder Ressourcen oder Literale sein. Im genannten Beispiel wäre das Objekt eine Ressource, da mehrere Dokumente unter der Lizenz »CC-BY-SA« veröffentlicht sein könnten. Diese Dokumente würden dann ebenfalls mit der einzigartigen Ressource CC-BY-SA verknüpft. Der Titel des Dokumentes hingegen würde als Literal gespeichert, da es sehr unwahrscheinlich ist, dass mehrere Dokumente unter demselben Namen veröffentlicht werden.

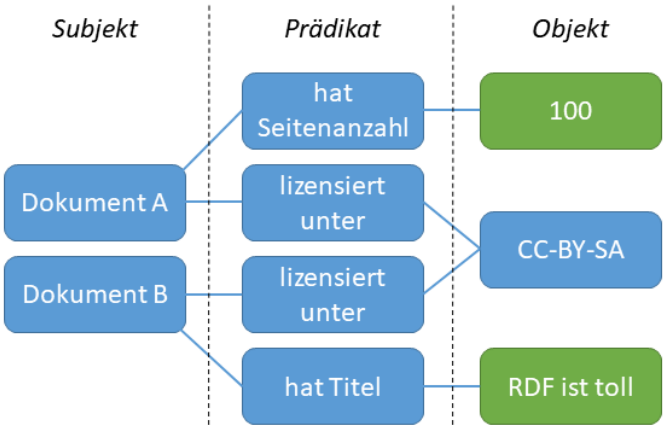


Abbildung 32: Ressourcen (blau) können wiederverwendet werden, Literale (grün) sind einfache Werte

Bildlich betrachtet entsteht durch die Verknüpfungen der Ressourcen ein Graph, in dem Knoten entweder ein Subjekt, Prädikat oder Objekt darstellen. Tripel werden

dementsprechend durch die jeweiligen Verbindungen der Knoten dargestellt (Abbildung 32).

Um Ressourcen wieder auffindbar und eindeutig identifizierbar zu machen, werden diese in Form von **Uniform Resource Identifier** (kurz: URI) kodiert. In dem Beispiel aus Abbildung 31 würde aus dem Subjekt *Dokument* dann beispielsweise *http://nqdm.de/Dokument* werden. Später hinzugefügte Tripel können auch diesen einzigartigen Bezeichner als Subjekt verwenden und dadurch unmissverständlich auf diese Ressource verweisen.

Diese Form der Datenspeicherung ermöglicht auch mächtige Abfragen über die Daten. Um alle Informationen zu der Ressource *Dokument* zu bekommen, könnte eine Abfrage folgendermaßen aussehen: »Gib mir alle Tripel, in denen das Subjekt *http://nqdm.de/dokument* ist«. Die dafür zum Einsatz kommende Abfragesprache nennt sich **SPARQL**. Entsprechende Anfragen werden von einer speziellen Datenbank beantwortet, einem sogenannten **Triple-Store**.

Um zu verhindern, dass alle Datenbereitsteller unterschiedliche URIs für semantisch gleiche Ressourcen verwenden, gibt es sogenannte **Vokabularien**. Diese standardisieren die Repräsentation bestimmter Prädikate und Objekte in RDF. Als Beispiel sei hier Dublin Core[31] genannt, das »Best Practice« für Metadaten fördern soll. Dort ist unter anderem festgelegt, dass das Objekt *Lizenz* mittels des Prädikats *http://purl.org/dc/terms/license* als Ressource vom Typ *http://purl.org/dc/terms/LicenseDocument* definiert werden sollte.

Vokabularien können auch aufeinander aufbauen beziehungsweise einander ergänzen. Um nicht bei jeder Ressource die komplette URI angeben zu müssen, können außerdem am Anfang eines RDF-Dokuments Präfixe bestimmt werden.

Um die entstehenden Graphen anderen verfügbar zu machen, lässt sich der Inhalt des Triple-Stores auch in Textform darstellen. Dafür existieren verschiedene Formate, die entweder auf bereits bestehenden Technologien aufbauen (RDF/XML, JSON-LD) oder aber eine Neuentwicklung darstellen (N3, Turtle). Ein Beispiel in Turtle-Syntax ist in Abbildung 33 zu sehen.

[31] Weitere Informationen unter http://dublincore.org/.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix dc: <http://purl.org/dc/elements/1.1/>.
@prefix dcterms: <http://purl.org/dc/terms/>.
@prefix dctype: <http://purl.org/dc/dcmitype/>.

<http://nqdm.org/johnDoe> a dcterms:Agent;
                          rdf:value "John Doe".

<http://nqdm.org/ccBySa> a dcterms:LicenseDocument .

<http://nqdm.org/document.html> a dctype:Text;
                                dcterms:creator <http://nqdm.org/johnDoe>;
                                dcterms:license <http://nqdm.org/ccBySa>;
                                dcterms:title "Sample Document".
```

Abbildung 33: RDF, dargestellt in Turtle-Syntax.

7.6.2 Empfehlungen

Vokabularien

Wo immer möglich sollten bereits existierende Vokabularien zur Speicherung von Ressourcen verwendet werden. Speziell für Metadaten aus dem öffentlichen Sektor hat sich DCAT-AP[32] als europäischer Standard durchgesetzt. Im deutschsprachigen Raum kommt häufig die Abwandlung **DCAT-AP.de**[33] zum Einsatz.

Kennung von Ressourcen

Zur Identifikation der Ressourcen sollten HTTP-URIs verwendet werden, da diese idealerweise einen direkten Aufruf der Ressourcen ermöglichen. Außerdem kann dann von diversen Techniken profitiert werden, die in erster Linie für HTTP entwickelt wurden, wie etwa die Indexierung durch Suchmaschinen, was wiederum für eine bessere Auffindbarkeit sorgen kann. Weiterhin sollten sich die URIs nicht ändern und keine Elemente enthalten, die sich auf absehbare Zeit ändern könnten, wie beispielsweise Statusinformationen oder Login-Informationen[34].

Validierung

Wie auch schon bei XML-Dateien besteht mit SHACL[35] ein mächtiges Werkzeug, um RDF-Dateien nach verschiedenen Kriterien zu validieren. Von dieser Möglichkeit sollte unbedingt Gebrauch gemacht werden, um hochwertige und somit leichter weiterzuverarbeitende Daten zu garantieren.

[32] Weitere Informationen unter https://joinup.ec.europa.eu/solution/dcat-application-profile-data-portals-europe.
[33] Weitere Informationen unter https://www.dcat-ap.de/.
[34] W3C (o.D.): Cool URIs don't change. Zuletzt aufgerufen im August 2019 unter https://www.w3.org/Provider/Style/URI.
[35] Weitere Informationen unter https://www.w3.org/TR/shacl/.

Namespaces und Bezeichner

Auch wenn RDF nicht primär für gute Menschenlesbarkeit entworfen wurde, sollten trotzdem die folgenden Konventionen eingehalten werden. Die Verwendung von Namespaces in Kombination mit Präfixen sorgt für übersichtlichere Dateien und hält deren Größe so klein wie möglich. Außerdem sollten Klassen in **PascalCase** und Eigenschaften in **camelCase** geschrieben werden. Wenn immer möglich, sollten existierende Vokabularien verwendet werden, die bereits standardisiert sind.

7.6.3 Nützliche Links

Beschreibung	Link
RDF-Spezifikation	https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/
SPARQL-Spezifikation	https://www.w3.org/TR/sparql11-overview/
Liste von RDF-Vokabularien	https://lov.okfn.org/dataset/lov/
Visualisierung von RDF	http://en.lodlive.it/

7.7 REST-Schnittstellen

Außer bei Datenformaten sind einheitliche Standards und Konventionen auch mit Bezug auf den eigentlichen Austausch der erstellten Ressourcen von Bedeutung. Wenn Dateien im Internet abrufbar sind, wird in diesem Kontext oft von sogenannten Schnittstellen, auch **API** genannt, gesprochen. Damit kann der Zugang durch Nutzerinnen und Nutzer gemeint sein, aber auch Möglichkeiten, diese Dateien automatisiert mithilfe von Programmen abrufbar zu machen. Um Datenverarbeitern den Zugang zu bereitgestellten Daten so leicht wie möglich zu machen, haben sich auch in diesem Bereich Standards und Konventionen etabliert. Ein weit verbreitetes Paradigma wird **Representational State Transfer** genannt, kurz REST[36].

7.7.1 Eigenschaften

Ein grundliegendes Prinzip von REST ist es, dass alle Nachrichten, die zwischen **Server** (Anbieter) und **Client** (Verarbeiter) ausgetauscht werden, stets sämtliche Informationen enthalten, um diese Nachricht interpretieren und verarbeiten zu können. Damit ist gemeint, dass weder der Server noch der Client die Zustandsinformationen speichern muss. Jede Nachricht ist dadurch in sich schlüssig und verlangt nicht das Vorhandensein weiterer Nachrichten. Eine Nachricht könnte etwa sein: »Gebe mir, was du anzubieten hast«, adressiert an einen Server im Internet. Als Adresse kann jede beliebige Internetadresse, auch **Uniform Resource Locator** (URL) genannt, dienen. Ein Beispiel ist in Abbildung 34 zu sehen.

```
{
  "success": true,
  "hits": 50,
  "page": 3,
  "totalPages": 25,
  "result": [{
    "type": "apple",
    "colour": "red",
    "origin": "Germany"
    "hasSeeds": true
  },
  {
    "type": "grape",
    "colour": "green",
    "origin": "Italy"
    "hasSeeds": false
  }]
}
```

Abbildung 34: Beispiel einer REST-Antwort im JSON-Format, die Informationen zu GOVDATA enthält.

[36] Wikipedia (05. August 2019): Representational State Transfer. Zuletzt aufgerufen im August 2019 unter https://de.wikipedia.org/wiki/Representational_State_Transfer.

Methoden

Die bereits erwähnten Adressen können auf unterschiedliche Art und Weise aufgerufen werden. Hierfür stehen sogenannte **Methoden** zur Verfügung, die im Protokoll HTTP spezifiziert sind. Der Anbieter muss zusätzlich zum eigentlichen Inhalt eine Antwort zurücksenden. Diese Antwortmöglichkeiten sind ebenfalls durch HTTP standardisiert und anhand von dreistelligen Zahlen codiert, die **Statuscode** genannt werden. So bedeutet etwa der Code *200*, dass alles in Ordnung ist, während *404* für »Not Found«, also »Nicht verfügbar« steht. Eine vollständige Liste der möglichen Codes ist in Abschnitt 7.7.3 verlinkt. Eine Auswahl der wichtigsten Methoden sowie der üblichen Antwortcodes ist in Tabelle 5 dargestellt.

Tabelle 5: Die wichtigsten Methoden für REST-Schnittstellen im Überblick.

NAME	BESCHREIBUNG	STATUSCODE	
		Fehlerfrei	Fehlerfall
GET	Ruft eine Ressource ab, ohne sie zu verändern.	200	404
POST	Lädt eine neue Ressource zum Server hoch.	201	400, 401, 403
PUT	Ersetzt eine vorhandene Ressource auf dem Server mit einer neuen, vollständigen Ressource.	200, 204	400, 401, 403
PATCH	Ersetzt einzelne (Meta-)Daten einer auf dem Server vorhandenen Ressource, ohne sie komplett zu ersetzen.	200, 204	400, 401, 403
DELETE	Löscht eine vorhandene Ressource.	200, 204	400, 401, 403

URL-Struktur

Am ehesten vergleichbar ist das REST-Prinzip mit Dateien auf einer Festplatte. Analog zu lokalen Dateien stellen Ressourcen eine Informationseinheit dar, die in unterschiedlichen Ordnern abgelegt sein können. Informationen zu einer bestimmten Zutat einer bestimmten Sorte Kuchen wären dann beispielsweise über die folgende URL abrufbar:

http://example.com/recipies/cake/ingredients/apples

Die grün hinterlegten Teile der URL können dabei als *Sammlung* bezeichnet und vom Anbieter selbst definiert werden. Im Gegensatz dazu fungieren die mit der Farbe Lila hinterlegten Pfade als Parameter, die vom Client selbst bestimmt werden können. Typischerweise handelt es sich dabei um die Kennungen (*ID*) einer Ressource einer Sammlung. Insgesamt ergibt sich daraus eine URL, die genau eine Ressource beschreibt. Diese kann anschließend mithilfe der bereits beschriebenen Methoden abgefragt oder manipuliert werden.

7.7.2 Empfehlungen

URL

Die vom Server angebotenen Adressen zum Auffinden von Ressourcen sollten aussagekräftig sein. Weiterhin sollten die Sammlungen mittels Substantive im Plural benannt werden, wie in Tabelle 6 dargestellt. Auch ist darauf zu achten, dass die Kennungen von Ressourcen einzigartig sind, sich also nicht mehrere Ressourcen eine **ID** teilen.

Die nachfolgende Tabelle zeigt exemplarisch Aufrufe sowie die empfohlene URL-Struktur. Dort wird veranschaulicht, dass die Unterscheidung, mit welcher Absicht eine Adresse aufgerufen wird, ausschließlich über die Methode geschehen sollte.

Tabelle 6: Beispielszenarien einer REST-Schnittstelle.

ABSICHT	METHODE	URL
Alle Ressourcen abfragen	GET	http://example.com/recipies
Eine Ressource hinzufügen	POST	http://example.com/recipies
Eine einzelne Ressource abfragen	GET	http://example.com/recipies/cake
Eine Ressource ersetzen	PUT	http://example.com/recipies/cake
Teile einer Ressource ersetzen	PATCH	http://example.com/recipies/cake
Eine Ressource löschen	DELETE	http://example.com/recipies/cake
Eine Unterressource abfragen	GET	http://example.com/recipies/cake/ingredients/apples

Mit der Weiterentwicklung einer Schnittstelle und der Erweiterung des Kataloges der angebotenen Daten kann es notwendig sein, die Schnittstelle anzupassen. Um weiterhin Abwärtskompatibilität zu gewährleisten, sollte die API (am besten von

Anfang an) versioniert sein. Dies kann auf verschiedene Art und Weise geschehen. Eine Möglichkeit ist das Nutzen eines weiteren Teils der URL:

*http://example.com/api/**v1**/recipies/cake/ingredients/apples*

*http://example.com/api/**v2**/recipies/desserts/cake/ingredients/apples*

Um Datenverarbeitern mitzuteilen, wie genau die eigene Schnittstelle gestaltet ist, also welche Methoden auf welchen Adressen welche Statuscodes zurückgeben können, empfiehlt es sich, die API präzise zu spezifizieren. Die *OpenAPI Initiative*[37] hat einen Standard entwickelt, der eben diese Spezifizierung erlaubt.

Große Datenmengen

Da potentiell sehr viele Daten über eine URL abrufbar sind, die eventuell nicht vollständig benötigt werden, oder die Auslieferung dieser Daten auf dem Server des Anbieters sehr viel Last erzeugen würde, können Daten mithilfe virtueller »Seiten« ausgeliefert werden. Dabei wird dem Client vorerst nur eine begrenzte Menge an Ressourcen gesendet, beispielsweise 100. Der Client kann anschließend festlegen, welche Seite er gerne ausgeliefert haben möchte und wie viele Ressourcen auf dieser Seite vorhanden sein sollen. Es gibt mehrere Methoden, wie dies geschehen kann. Empfohlen wird, die Wahl der ausgelieferten Seiten anhand der in Tabelle 7 genannten URL Parameter zu ermöglichen. Bei Verwendung der vorgeschlagenen Einstellungen würden damit die ersten 20 Ressourcen ausgeliefert.

Tabelle 7: Empfohlene Parameter, um Ressourcen seitenweise anzubieten.

PARAMETER	FUNKTION	VOREINGESTELLT
offset	Beschreibt, bei welcher Ressource angefangen werden soll (Versatz).	0
limit	Beschreibt, wie viele Ressourcen insgesamt ausgeliefert werden sollen.	20

Um die Rezepte 30 bis 40 abzurufen, würde in unserem Beispiel die URL also wie folgt aussehen:

http://example.com/api/v1/recipies?offset=30&limit=10

[37] Weitere Informationen unter: https://www.openapis.org/.

7.7.3 Nützliche Links

Beschreibung	Link
HTTP-Statuscode	https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html
OpenAPI-Editor	https://swagger.io/tools/swagger-editor/

7.8 WFS-Dienste

7.8.1 Aufbau und Struktur

Die Web Feature Service (WFS)-Spezifikation wurde durch das Open Geospatial Consortium (OGC) erstellt und definiert eine standardisierte Schnittstelle, welche einen internet-gestützten Zugriff auf Geodaten (-Objekte/Features) ermöglicht.

Die WFS-Spezifikation umfasst ausschließlich **Vektordaten**, welche **GML**-codiert übermittelt werden. Für Rasterdaten gibt es die Web Coverage Service (WCS-) Spezifikation.

Die WFS-Spezifikation liegt in drei Versionen vor (Version 1.0.0, 1.1.0 und 2.0). Alle Versionen sind gültig, d.h. die höheren Versionen ersetzen nicht die niedrigeren Versionen. Die drei Versionen bestehen parallel und werden produktiv eingesetzt. Auf welche Version bei der Implementierung eines WFS zurückgegriffen werden sollte, hängt vom Einsatzszenario ab. Die Versionen bieten einen unterschiedlichen Funktionsumfang und sind mit verschiedenen GML-Versionen und OGC-Standard Filter Encoding Versionen verbunden (FE). Die FE-Version legt z.B. fest, welche Möglichkeiten der räumlichen Filterung zur Verfügung stehen.

Die folgende Tabelle zeigt eine Übersicht, welche FE- und GML-Version in den unterschiedlichen WFS-Versionen unterstützt werden müssen:

Tabelle 8: WFS-Versionen und ihre Unterstützung von FE- und GML-Versionen.

WFS	FE	GML
1.0	1.0	2.1.1
1.1	1.1	3.1
2.0	2.0	3.2

Diese Kombinationen sind die Standardversionen und stellen die Mindestanforderung dar, welche bei der Implementierung umgesetzt werden muss. Es ist durchaus erlaubt, dass ein WFS 1.1 als Rückgabe GML 3.2 oder GML 2.1 (auf Anfrage des Clients) liefert, solange mindestens GML 3.1 implementiert ist. Der Client muss das gewünschte Rückgabeformat im Request als MIME-Type angeben. Es sind weitere, zusätzliche Rückgabeformate erlaubt. Zum Beispiel darf der WFS die Geometrien im KML-Format zurückgeben, solange er auch GML unterstützt.

Operationen

Die fünf Operationen, welche in der WFS-Version 1.0 spezifiziert sind, können als Basisoperationen gesehen werden, da sie in allen Versionen enthalten sind:

Tabelle 9: WFS-Versionen und ihre Operationen.

WFS 1.0	WFS 1.1.0	WFS 2.0
GetCapabilities	GetCapabilities	GetCapabilities
DescribeFeatureType	DescribeFeatureType	DescribeFeatureType
GetFeature /	GetFeature /	GetPropertyValue
GetFeatureWithLock	GetFeatureWithLock	GetFeature /
LockFeature	GetGmlObject	GetFeatureWithLock
Transaction	LockFeature	LockFeature
	Transaction	Transaction
		CreateStoredQuery
		DropStoredQuery
		ListSoredQueries
		DescribeStoredQueries

Mit einem *GetCapabilities*-Request können die Fähigkeiten und *FeaturesTypes* eines WFS abgefragt werden. Mit dem *DescribeFeatureType* kann die Struktur der angebotenen *FeatureTypes* abgefragt werden und mit *GetFeature* dann konkrete Instanzen der *FeatureTypes*.

Ein *LockFeature*-Request sperrt ein Feature für den Zugriff durch andere Nutzerinnen und Nutzer. Möchte eine Nutzerin oder ein Nutzer ein Feature aktualisieren, muss er zuerst ein *GetFeature*-Request durchführen, um das Feature mit seiner aktuellen Ausprägung abzurufen. Dann kann er es auf seinem Client verändern und abschließend mit einem *Transaction*-Request aktualisieren. Um zu verhindern, dass das Feature in der Zwischenzeit (in der das Feature auf dem Client bearbeitet wird) von einer dritten Person verändert wird, kann man es vorher mit dem *LockFeature*-Request sperren. Wird ein erfolgreicher *Transaction*-Request auf einem Feature ausgeführt, wird das Feature automatisch wieder entsperrt, andernfalls wird das Feature nach einer definierten Zeitspanne automatisch wieder freigegeben. *Transaction*-Requests steuern das Einfügen, Ändern und Löschen von Objekten.

Konformität

Da große Teile der Spezifikationen optionale Bestandteile eines WFS beschreiben, enthält jede Spezifikation ein Kapitel zur Konformität. Die Konformität beschreibt, welche Teile der Spezifikation umgesetzt sind. Die kleinste Stufe enthält dabei alle Pflichtbestandteile. Auf diesen Pflichtbestandteilen aufbauend folgen die optionalen Bestandteile.

In der Version 1.1.0 werden zusätzlich drei WFS-Typen unterschieden: Werden die ersten drei Operationen unterstützt, so bietet der WFS nur lesenden Zugriff und man spricht von einem »Basic WFS«. Wird zusätzlich die Operation *GetGmlObject* unterstützt, spricht man von einem »XLink WFS«. Kommt die Operation *Transaction* zu den ersten drei dazu, liegt »Transaction WFS« oder »WFS-T« vor. Beim WFS-T sind die Operationen *LockFeature* und *GetGmlObject* optional. Basic-WFS ist also die Grundlage

für XLink WFS und WFS-T, der Typ XLink WFS ist aber nicht zwingend in einem WFS-T enthalten.

7.8.2 Empfehlungen

WFS-Version und Konformität

Bei der Bereitstellung eines WFS-Dienstes steht zu Beginn ein Verwendungszweck bzw. eine Anforderung. Um unnötigen Aufwand bei der Entwicklung des WFS-Dienstes zu vermeiden, sollte anhand des Verwendungszwecks ermittelt werden, welche WFS-Spezifikation verwendet wird und welcher Grad an Konformität umgesetzt werden soll. Soll einer möglichst breiten Menge an Clients der Konsum eines WFS ermöglicht werden, so macht es Sinn, parallele Dienste in mehreren Versionen bereitzustellen.

Wichtig ist daher die Abwägung zwischen Aufwand für die Bereitstellung und der späteren Nutzbarkeit für die Anwenderinnen und Anwender bzw. der Software-Clients.

Zusätzliche Bereitstellung als GeoJSON

Die Daten sollten zusätzlich auch im GeoJSON-Format bereitgestellt werden, da dieses Format verbreiteter und verständlicher ist und somit eine größere Zielgruppe anspricht.

Koordinatenreihenfolge

Obwohl es für die reine Geometriedarstellung einer Linie ohne Bedeutung ist, in welcher Richtung die Koordinaten angegeben werden, spielt dies bei der Interpretation durch Tools und Werkzeuge eine Rolle. Viele Tools interpretieren die Reihenfolge als »Fließrichtung«, beispielsweise bei der Darstellung von Flüssen.

Antimeridian

Beim Umgang mit Koordinaten am sogenannten Antimeridian, also im Bereich, in dem der Wechsel von +180° zu -180° stattfindet, kann es zu Fehlinterpretationen des Geometrieverlaufs kommen. Verläuft eine Linie von 170° nach -170°, kann eine Software nicht entscheiden, ob die Geometrie einmal rund um den Globus verläuft oder über den Antimeridian und dadurch der Vorzeichenwechsel zustande kommt.

Die Empfehlung ist in diesem Fall, eine zusätzliche Koordinate direkt auf dem Antimeridian anzulegen. Dabei wird diese Position doppelt angegeben, einmal mit positivem und einmal mit negativem Vorzeichen, wie in Abbildung 35 dargestellt.

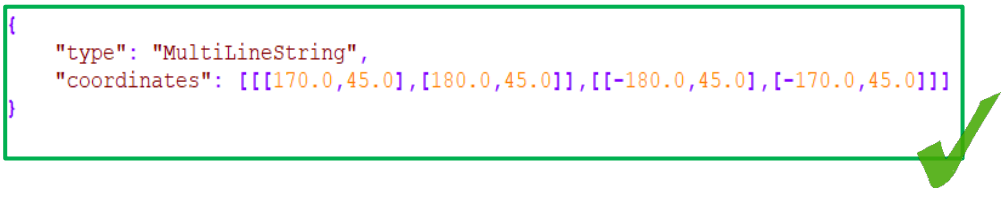


Abbildung 35: Beispiel für die Angabe von Koordinaten am Antimeridian.

7.8.3 Nützliche Links

Beschreibung	Link
WFS-Spezifikation (korrigierte Fassung)	http://docs.opengeospatial.org/is/04-094r1/04-094r1.html
Kurze Einführung in WFS 2.0	https://www.weichand.de/2011/11/30/grundlagen-web-feature-service-wfs-2-0/
Grundsätzlicher Aufbau von HTTP Requests an einen WFS	https://enterprise.arcgis.com/de/server/latest/publish-services/linux/communicating-with-a-wfs-service-in-a-web-browser.htm

8 Metadaten

Metadaten sind Daten, die Informationen über den Inhalt eines Datensatzes liefern. Damit sind Metadaten streng genommen wichtiger, als die eigentliche Ressource, denn sie entscheiden, ob eine Ressource aufgefunden wird. Außerdem nutzen Datenanwenderinnen und -anwender Metadaten, um abzuschätzen, ob eine gefundene Ressource dem entspricht, was er oder sie gesucht haben. Die Wichtigkeit von Metadaten kann an dem Beispiel eines Buches verdeutlicht werden. Metadaten sind hier beispielsweise Autor, Titel, Verlag, Erscheinungsjahr, ISBN etc. Ohne diese Angaben könnte das Buch in einer Bibliothek kaum aufgefunden werden. Analog zu diesem Beispiel bedürfen auch Daten in einem Datenkatalog entsprechende Angaben, um von Nutzerinnen und Nutzern gefunden werden zu können.

Liegen Metadaten in unzureichender Qualität vor, kann dies zum einen das Auffinden der Ressource erschweren oder gar verhindern. Zum anderen kann es dazu führen, dass Datenanwenderinnen und -anwender die gefundene Ressource nicht einordnen können oder sie ihnen unverständlich bleibt. Daher wird in diesem Kapitel die Qualität von Metadaten behandelt. Abschnitt 8.2 listet Hinweise für den Umgang mit Metadaten auf. In Abschnitt 8.4 wird der deutsche Metadatenstandard für offene Verwaltungsdaten behandelt. Abschließend wird in Abschnitt 8.5 eine Bewertungsmatrix zur Erreichung einer hohen Metadatenqualität vorgestellt.

8.1 Metriken zur Messung der Metadatenqualität

Es existieren mehrere verschiedene Verfahren zur Beurteilung der Daten- und Metadatenqualität, die unterschiedliche Kriterien wie Datenformate und Lizenzen berücksichtigen. Die Daten-Metrik 5 Star Data (siehe Abschnitt 4.1) unterscheidet zwischen fünf Stufen, wobei jede Stufe genau eine zusätzliche Bedingung an die Daten stellt. Die Open Definition[38] konzentriert sich vor allem auf die genaue Spezifizierung der als »offen« geltenden Lizenzen und führt auch keine Abstufungen durch – entweder wird der Metrik entsprochen oder nicht. Deutlich ausführlicher ist das Open Data Barometer[39], das präzise Anforderungen formuliert und die Datenqualität auf eine Skala zwischen 0 und 100 abbildet.

Eine Gemeinsamkeit finden die meisten Verfahren bei den für die Messung verwendeten Metriken. The Continuum of Metadata Quality[40] bezeichnet folgende sechs Metriken als bekannteste Merkmale für die Messung von Metadatenqualität:

Vollständigkeit

Metadaten sollten das zu beschreibende Objekt so ausführlich und vollständig wie möglich beschreiben. Es ist außerdem zu beachten, dass die Beschreibungen auch auf die Objekte angewendet werden können. Es nützt wenig, bestimmte

[38] Weitere Informationen unter https://opendefinition.org/od/2.0/de/.
[39] Weitere Informationen unter https://docs.google.com/document/d/1_6FR6nZ6Uj4xr2tMsnhGBjocbxOd4MRcBsXe2OMh1CE/edit#heading=h.6w58sp75k6zv.
[40] Bruce, T. R.; Hillmann, D. I. (2004): The Continuum of Metadata Quality: Defining, Expressing, Expointing. In: Metadata in Practrice, D. Hillmann & E Westbrooks, eds.

Informationen vorzuschreiben, wenn ein Großteil davon nicht angewendet werden kann oder für die gesamte Objektmenge nicht verlässlich ist.

Genauigkeit

Die Beschreibung der Metadaten sollte genau sein. Es empfiehlt sich, auf ausschweifende Formulierungen zu verzichten und stattdessen korrekte und sachliche Beschreibungen zu formulieren. Wiederkehrende Begriffe sollten abgestimmt sein und bestenfalls einem Vokabular folgen.

Herkunft

Hinsichtlich einer Beurteilung auf die Verlässlichkeit der Daten sollte ihre Herkunft durch die Metadaten immer nachvollziehbar sein.

Erwartungen

Die Festlegung eines Standards zur Beschreibung der Metadaten weckt auch Erwartungen der Bearbeiter oder einer Community, wie und in welchem Umfang die Daten beschrieben werden. Sie sollten keine falschen Versprechungen enthalten, d.h. Elemente, die wahrscheinlich nicht verwendet werden, weil sie überflüssig, irrelevant oder nicht umsetzbar sind.

Konsistenz und Kohärenz

Metadaten sollten einem allgemeinen Schema folgen und nicht als proprietäre Beschreibungen in einem geschlossenen Silo verwendet werden. Stattdessen sollten sie kompatibel zu öffentlichem System sein und idealerweise Standards folgen. Einer dieser Standards (DCAT-AP.de[41]) wird in Kapitel 8.4 vorgestellt.

Aktualität

Metadaten sollten aktuell gehalten werden. Änderungen an den Daten sollten sich zeitnah auch in den Metadaten wiederspiegeln, beispielsweise durch ein Datum, das beschreibt, wann zuletzt Änderungen an den Daten vorgenommen wurden. Ebenso sollten Metadaten bereits mit der Veröffentlichung von Daten zur Verfügung stehen.

8.2 Allgemeine Hinweise zum Umgang mit Metadaten

Grundsätzlich gilt, dass Metadaten stets aktuell, inhaltlich richtig und zutreffend sein sollten. Daher empfiehlt es sich, Metadaten in regelmäßigen Abständen zu überprüfen und im Falle von Änderungen zu aktualisieren.

Folgende Hinweise sollten des Weiteren beachtet werden:

- Der Titel des Datensatzes sollte prägnant gewählt werden, damit die Daten sowie ergänzende Beschreibungen leicht identifiziert und eingeordnet werden können. Beziehen sich die Daten auf einen bestimmten Geltungsbereich (geografisch oder zeitlich), sollte dieser in den Titel aufgenommen werden.
- Bei der Eingabe von Schlagwörtern sollten, wenn möglich, bestehende Vokabularien verwendet werden.
- Neben Fachvokabular sollten auch leicht verständliche Synonyme für die Beschreibung der Daten verwendet werden, sodass auch Laien die Daten verstehen können.
- Auf Abkürzungen sollte verzichtet werden, um sicherzustellen, dass Nutzerinnen und Nutzer unabhängig von ihrem fachlichen Hintergrund die Beschreibung verstehen und die Ressource aufgefunden wird.

[41] Weitere Informationen unter https://www.dcat-ap.de/.

- Freitextfelder sollten genutzt werden, um die Daten und ihre Beschaffenheit zu beschreiben. Hier können auch Erläuterungen zur Erhebungsmethode, zusätzliche Informationen zur Spalteninhalten o.ä. hinterlegt werden.
- Umfassen die Daten einen bestimmten Zeitraum oder beziehen sie sich auf einen konkreten geografischen Raum, so sollten diese Angaben in den Metadaten stets mitgeliefert werden (möglichst im Titel), damit Datenanwenderinnen und -anwender die Daten schnell in einen Kontext einordnen können.
  - Beispiele für Titel mit genauer temporaler oder räumlicher Abgrenzung:
    - »Tourismuszahlen in München (Januar 2018)«
    - »Stadt Wesel: Anmeldungen zur weiterführenden Schule in Wesel 2016«

8.3 Verwendung von kontrollierten Vokabularen

Wie bereits in Kapitel 5 beschrieben, sollten wenn immer möglich kontrollierte Vokabulare zur Beschreibung von definierten Informationen, wie z.B. Ortschaften, Lizenzen und Einrichtungen, verwendet werden. Die Europäische Kommission hat mit den EU Vocabularies[42] dazu ein umfassendes Angebot veröffentlicht. Vokabulare helfen dabei, Heterogenität in den Metadaten über die Organisationsgrenzen hinaus zu reduzieren und schaffen dadurch Klarheit und fördern zudem die Maschineninterpretierbarkeit. Jedes Vokabular wird üblicherweise in Form einer RDF-Datei zum Herunterladen angeboten, die jedoch je nach Domäne unterschiedlich groß ausfallen können.

8.4 DCAT-AP.de

Der deutsche Metadatenstandard für offene Verwaltungsdaten ist DCAT-AP.de. Daneben existieren weitere Standards, wie beispielsweise INSPIRE und GeoDCAT-AP für Geodaten, DCAT-AP.de ist jedoch führend. Als gemeinsames deutsches Metadatenmodell zum Austausch von offenen Verwaltungsdaten wird DCAT-AP.de von **GovData** spezifiziert. DCAT-AP.de ist seit Ende 2017 bei GovData als Metadatenmodell im Einsatz. DCAT-AP.de ist als eine Ableitung vollständig kompatibel zum europäischen Standard DCAT-AP.

Das »AP« in DCAT-AP.de steht für Application Profile. Dies stellt eine Spezifikation[43] dar, »die Begrifflichkeiten bzw. Konzepte eines oder mehrerer grundlegender Standards weiterverwendet. Eine größere Bestimmtheit wird erreicht, indem für eine bestimmte Anwendung Klassen und Klassenattribute (Eigenschaften) als *obligatorisch*, *empfohlen* oder *optional* eingeordnet werden. Zusätzlich werden Empfehlungen für die Verwendung von kontrollierten Vokabularen gegeben.«[44]

[42] Weitere Informationen unter https://publications.europa.eu/en/web/eu-vocabularies/home.
[43] Die DCAT-AP.de- pezifikation kann unter folgendem Link eingesehen werden: https://www.dcat-ap.de/def/dcatde/1.0.1/spec/specification.pdf.
[44] DCAT-AP.de-Spezifikation, S. 9.

Die gesamte DCAT-AP.de-Spezifikation, Vokabulare, das Konventionenhandbuch sowie weitere Dokumente und Artefakte können unter folgendem Link eingesehen werden: https://www.dcat-ap.de/def/.

8.5 Orientierungshilfe zur Erreichung einer hohen Metadatenqualität

Tabelle 10 zeigt eine Orientierungshilfe für Metadaten auf Basis von DCAT-AP.de. Die Tabelle basiert auf den Ergebnissen zum »Automated Quality Assessment of Metadata across Open Data Portals« [45] der Wirtschaftsuniversität Wien.

Die Tabelle soll maßgeblich dabei helfen zu verstehen, welche Metainformation hilfreich sind, um die Verbreitung und Wiederverwendung von Metadaten und Daten zu fördern. Aus diesem Grund muss die nachfolgende Empfehlung nicht zu 100% mit der Einteilung der Pflichtfelder, empfohlenen Felder und optionalen Felder in DCAT-AP.de übereinstimmen. Es wird hingegen empfohlen, Informationen auch über die zu verpflichtenden Felder hinaus anzugeben. Denn diese können entscheidend an der Auffindbarkeit und Wiederverwendbarkeit mitwirken. Die nachfolgende Tabelle gliedert sich in folgende drei Bereiche:

Vorhandensein wichtiger Informationen
: Dieser Abschnitt nennt Felder, die unabhängig von den Pflichtangaben ausgefüllt werden sollten, da sie die Auffindbarkeit stark beeinflussen.

Konformität zu DCAT-AP.de, Beispiele
: Beim Angeben von Information definiert DCAT-AP.de, in welcher Form die Informationen angegeben werden soll. Teilweise wird auch die Verwendung von Vokabularien verlangt. Dieser Abschnitt zeigt an einigen Beispielen, auf welche Spezifika geachtet werden sollte.

Maschinenlesbarkeit
: Maschinenlesbarkeit ist ein wichtiger Aspekt bei der Verwendung der Daten und sollte auch in den Metadaten entsprechend gekennzeichnet sein.

Ferner ist die Tabelle in vier Spalten unterteilt:

Indikator
: Eine Klassifizierung von anzugebenden Metadateninformationen.

Erläuterung
: Eine Beschreibung des Indikators.

dcat:Dataset
: Zeigt, welche Felder in der Klasse dcat:Dataset für den Indikator zutreffend sind.

dcat:Distribution
: Zeigt, welche Felder in der Klasse dcat:Distribution für den Indikator zutreffend sind.

Priorität
: Einstufung der Priorität mit den Stufen hoch, mittel und niedrig hinsichtlich der Relevanz des Bereichs für die Wiederverwendbarkeit und Verbreitung der Daten.

[45] Neumaier, S.; Umbrich, J.; Polleres, A. (2016): Automated Quality Assessment of Metadata across Open Data Portals. In: *Journal of Data and Information Quality* 8 (1), S. 1-29. DOI: 10.1145/2964909.

Die Orientierungshilfe soll dafür sensibilisieren, welche Information der Metadaten qualitätssteigernd sind. Dies steht nicht in Zusammenhang mit den Verbindlichkeitsstufen von DCAT-AP.de. Die Verbindlichkeitsstufen können der aktuellen Version der DCAT-AP.de-Spezifikation[46] entnommen werden.

[46] Weitere Informationen unter https://www.dcat-ap.de/def/.

Tabelle 10: Orientierungshilfe für Metadaten auf Basis von DCAT-AP.de.

Indikator	Erläuterung	dcat:Dataset	dcat:Distribution	Priorität
	Vorhandensein wichtiger Informationen			
Zugang	Sind Informationen für die Distribution vorhanden?		dcat:accessURL dcat:downloadURL	
Auffindbarkeit	Sind Informationen verfügbar, die die. Auffindbarkeit erleichtern?	dct:title dct:description dcat:keyword dcat:theme		
Kontakt	Sind Kontaktinformationen verfügbar	dcat:contactPoint dct:publisher		Hoch
Lizenz	Lizenzinformationen sind vorhanden		dct:license	
Erhaltung	Informationen die die Aktualität, das Format und die Größe der Distribution beschreiben		dct:format dcat:mediaType dcat:byteSize	
Daten	Datumsinformationen der Metadaten und Distributionen sind vorhanden	dct:issued dct:modified	dct:issued dct:modified	
	Konformität zu DCAT-AP.de, Beispiele			
URLs	Werden korrekte URLs verwendet		dcat:accessURL dcat:downloadURL	
Kontaktemail	Wird eine korrekte E-Mail angegeben	dcat:contactPoint dct:publisher		
KontaktURL	Ist eine korrekte KontaktURL angegeben worden	dcat:contactPoint dct:publisher		Mittel
Datumsformat	Wird bei Datumseintragungen das korrekte Format verwendet	dct:issued dct:modified	dct:issued dct:modified	
Lizenz	Wird eine Lizenz verwendet, die allgemein akzeptiert ist		dct:license	
Dateiformat	Wird ein bei IANA registriertes Dateiformat oder Media Type verwendet		dct:format dcat:mediaType	
	Maschinenlesbarkeit			
Offenes Format	Werden Dateiformate verwendet, die einem offenen Standard zugeordnet sind		dct:format dcat:mediaType	Hoch
Maschinenlesbar	Können die verwendeten Formate als maschinenlesbar eingestuft werden		dct:format dcat:mediaType	

A Anhang

A.1 Glossar

API
Eine API (englisch für Application Programming Interface) ist eine Programmierschnittstelle. Sie wird von einem Softwaresystem bereitgestellt und ermöglicht es anderen Programmen, mit diesem Softwaresystem zu kommunizieren. APIs werden oftmals von Datenherausgebern bereitgestellt und ermöglichen es Programmen oder Apps, die Daten direkt über das Web zu lesen. Dazu sendet die App eine Abfrage an die API nach den erforderlichen Daten. Vorteilhaft an der Bereitstellung der Daten per API ist, dass nicht der gesamte Datensatz heruntergeladen werden muss, da die API lediglich die benötigten Daten bereitstellt. Außerdem wird somit sichergestellt, dass die Daten auf dem aktuellen Stand sind.

Attribut
In der XML-Beschreibungssprache stellt ein Attribut ein Namen-Werte-Paar dar, das Teil eines Tags ist. Ein Attribut darf nur einmal pro Tag vorkommen und kann nur einzelne Werte enthalten.

Bounding Box
Eine Bounding Box ist ein einfacher geometrischer Körper, wie beispielsweise ein Quader oder Rechteck, der u.a. in der Geoinformatik eingesetzt wird, um komplexe Gebiete zu umschreiben. Der blaue Rahmen in der Abbildung rechts zeigt die Bounding Box für das ausgewählte Gebiet.


Bounding Box

Bulk-Download
Von einem Bulk-Download spricht man, wenn ein Datensatz in seiner Gesamtheit einfach, effizient und auf einmal heruntergeladen werden kann.

camelCase
Leer- und Sonderzeichen in Bezeichnern können die Datenverarbeitung erschweren. Wenn Bezeichner aus mehreren Wörtern bestehen, wird daher empfohlen, diese Wörter zu einem zusammenzufassen. In der camelCase-Schreibweise werden dabei mit Ausnahme des ersten Wortes die Anfangsbuchstaben großgeschrieben, um die menschliche Lesbarkeit zu erleichtern. Dies ist unabhängig von der Wortart, d.h. auch Verben und Adjektive beginnen je nach Stellung im Wort mit einem Großbuchstaben.

Captcha
Captcha steht für »Completely Automated Public Turing test to Tell Computers and Humans Apart« und beschreibt ein vollautomatisiertes Testverfahren zur Unterscheidung von Menschen und Computern. Captchas werden beispielsweise eingesetzt, um automatisierte Spam-Einträge in Blogs zu verhindern.

Client
Ein Nutzer beziehungsweise Empfänger von Daten. Damit kann ein Computer oder ein Mensch gemeint sein.

Coordinated Universal Time
Die Coordinated Universal Time (UTC) oder auch koordinierte Weltzeit ist ein Zeitstandard, der für die Angabe von Ortszeiten genutzt wird. Dazu werden ausgehend vom Nullmeridian die Zeitzonen und ihre Verschiebung zur UTC angegeben.

Creative Commons Lizenzen
Creative Commons sind vorgefertigte Lizenzverträge, mit denen Urheber auf einfache Weise der Öffentlichkeit mitteilen können, was diese mit den entsprechenden Inhalten tun dürfen und was nicht. Die CC-Lizenzen sind international anerkannt. Es gibt insgesamt sechs verschiedene CC-Lizenzen:

LIZENZ	ERKLÄRUNG
CC BY	Der Name des Urhebers muss genannt werden.
CC BY SA	Der Name des Urhebers muss genannt werden und das Werk muss nach der Veränderung unter der gleichen Lizenz weitergegeben werden.
CC BY ND	Der Name des Urhebers muss genannt werden und das Werk darf nicht verändert werden.
CC BY NC	Der Name des Urhebers muss genannt werden und das Werk darf nicht für kommerzielle Zwecke verwendet werden.
CC BY NC SA	Der Name des Urhebers muss genannt werden, das Werk darf nicht für kommerzielle Zwecke verwendet werden und das Werk muss nach der Veränderung unter der gleichen Lizenz weitergegeben werden.
CC BY NC ND	Der Name des Urhebers muss genannt werden, das Werk darf nicht für kommerzielle Zwecke verwendet werden und das Werk darf nicht verändert werden.

Quelle: Creative Commons: https://de.creativecommons.org/index.php/was-ist-cc/ (Stand August 2019)

CSV-Format
CSV (Comma-separated values) ist ein Standardformat für strukturierte Daten. Aufgrund der Einfachheit, Offenheit und Maschinenlesbarkeit wird CSV häufig für die Veröffentlichung von offenen Daten verwendet.

Datenbereitsteller
Als Datenbereitsteller wird diejenige Stelle bezeichnet, welche Inhalte über eine Plattform für Nutzerinnen und Nutzer zugänglich macht. Die Entscheidung über die Veröffentlichung, Nutzungsbestimmungen und Formaten obliegt dem Datenbereitsteller.

Datennutzer
Als Datennutzerin oder Datennutzer werden diejenigen natürlichen oder juristischen Personen bezeichnet, welche die von Datenbereitsteller zur Verfügung gestellten Daten gemäß den vorgesehenen Nutzungsbestimmungen für ihre eigenen Zwecke verwenden.

Datenlizenz Deutschland 2.0

Die Datenlizenz Deutschland ist eine standardisierte Nutzungsbestimmung für Verwaltungsdaten in Deutschland. Sie liegt derzeit in der Version 2.0 vor und umfasst zwei Varianten:

VARIANTE	BESCHREIBUNG
Datenlizenz Deutschland - Namensnennung - Version 2.0	Der Name der Quelle muss genannt werden.
Datenlizenz Deutschland - Zero - Version 2.0	Uneingeschränkte Weiterverwendung erlaubt.

Quelle: GovData, https://www.govdata.de/web/guest/lizenzen (Stand August 2019).

Datensatz

Ein Datensatz ist eine Menge von Daten, die inhaltlich zusammenhängen. Ein Datensatz enthält in der Regel eine oder mehrere Ressourcen, die beispielsweise unterschiedliche Formate abdecken, sowie Metadaten, die den Inhalt der Ressourcen beschreiben.

DCAT-AP.de

DCAT-AP.de ist der deutsche Metadatenstandard für offene Verwaltungsdaten. Als gemeinsames deutsches Metadatenmodell zum Austausch von offenen Verwaltungsdaten wird DCAT-AP.de von GovData spezifiziert und ist seit Ende 2017 bei GovData im Einsatz. DCAT-AP.de ist als eine Ableitung vollständig kompatibel zum europäischen Standard DCAT-AP.

Deklaration

Im XML-Kontext wird unter Deklaration die erste Zeile eines Dokumentes verstanden. Dort sind Metadaten, wie die verwendete Version, sowie Zeichenkodierung hinterlegt.

Distribution

Eine Distribution stellt eine Datei eines Datensatzes dar.

E-Government-Gesetz

Das Gesetz zur Förderung der elektronischen Verwaltung (kurz E-Government-Gesetz) des Bundes zielt darauf ab, die elektronische Kommunikation mit der Verwaltung zu erleichtern und elektronische Verwaltungsdienste effizienter und nutzerfreundlicher zu gestalten.

Element

Als Element wird bei XML ein Feld mit Daten bezeichnet. Ein Element wird mithilfe von Tags definiert und kann zusätzlich Attribute enthalten.

GML

GML steht für Geography Markup Language. Es handelt sich dabei um eine XML-basierte Auszeichnungssprache, die für den Austausch und die Darstellung von raumbezogenen Objekten genutzt wird.

GovData

GovData ist das nationale Datenportal für Deutschland und bietet eine zentrale Anlaufstelle für Daten aus allen Verwaltungsebenen. Das Portal kann unter folgender URL aufgerufen werden: https://www.govdata.de/.

ID
Als ID wird eine eindeutige Kennung für eine zusammengehörende Menge an Daten bezeichnet. Häufig wird dazu eine fortlaufende Nummerierung verwendet. Auch eine URI ist eine Art ID.

JSON
JSON ist ein leistungsfähiges Format, welches gut geeignet ist für den Datenaustausch zwischen verschiedenen Anwendungen. Es kann komplexe Datenstrukturen beschreiben, ist sowohl für Menschen als auch Maschinen gut lesbar und unabhängig von Plattform und Programmiersprache.

Linked Open Government Data
Linked Open Government Data vernetzt durch RDF-Konzepte beschriebene Datenmengen des öffentlichen Sektors. Die Vernetzung der Daten ermöglicht die Entstehung neuer Informationszusammenhänge und die Nutzung der Daten über ihren ursprünglichen Kontext hinweg. LOD ist in hohem Maße selbsterklärend und automatisiert interpretierbar.

Literal
Im Kontext von RDF bezeichnet ein Literal einen einfachen Datenwert. Nur RDF-Objekte dürfen Literale sein. Diese sind, im Gegensatz zu RDF-Ressourcen, nicht mit einer URI codiert und somit auch nicht von außerhalb des »eigenen« Tripels referenzierbar. Literale werden häufig für Daten verwendet, die außerhalb des eigenen Tripels ihre Bedeutung verlieren, beispielsweise der Name von Personen.

Lizenz
Eine Lizenz beschreibt Nutzungsbestimmungen für die Verwendung eines Werkes. Über eine Lizenz räumt der Urheber des Werkes dem Lizenznehmer gewissen Nutzungsrechte zu bestimmten Nutzungsbedingungen ein. Standard-Lizenzverträge, wie z.B. Creative Commons oder Datenlizenz Deutschland 2.0, erleichtern dem Urheber die Festlegung der Nutzungsbestimmungen.

Maschinenlesbarkeit
»Grundsätzlich sind alle von Software interpretierbaren Daten maschinenlesbar. Im Zusammenhang mit Open Data werden darunter vor allem solche Datenformate verstanden, die eine Weiterverarbeitung ermöglichen. Die zu Grunde liegende Datenstruktur und entsprechende Standards müssen öffentlich zugänglich sein und sollten vollständig publiziert und kostenfrei erhältlich sein.«[47]

Metadaten
Metadaten werden für die Erfassung und Beschreibung eines Datensatzes in strukturierter Form verwendet. Sie enthalten bspw. Informationen über den Inhalt, den Titel oder das Format eines Datensatzes. Kurz gesagt sind Metadaten Daten über Daten bzw. Verweise auf die eigentlichen Daten. Dabei folgen Metadaten meist einem

[47] Offene Bund-Länder-Arbeitsgruppe des IT-Planungsrats (Hrsg.) (2011): Offenes Regierungs- und Verwaltungshandeln (Open Government). Eckpunkte zur Förderung von Transparenz, Teilhabe und Zusammenarbeit - Entwurf (Stand: 6.9.2011 Version nach Abstimmung mit offener Bund-Länder-AG). Zuletzt abgerufen im August 2019 unter https://www.b-b-e.de/fileadmin/inhalte/aktuelles/2012/06/nl11_eckpunkte.pdf.

bestimmten Schema, welches obligatorische und optionale Informationen über den Datensatz vorgibt.

Metadatenqualität
Qualitativ hochwertige Metadaten erleichtern das Auffinden und die Nutzung von Daten. Da Metadaten ebenfalls Daten (über Daten) darstellen, können ähnliche Merkmale zur Bewertung ihrer Qualität herangezogen werden wie für Daten.

Methode, HTTP
Die HTTP-Methode zeigt die Absicht der aufrufenden Partei an einen HTTP-Server an. Ein Aufruf mit der Methode *GET* bedeutet, dass Ressourcen abgerufen werden sollen. Ein Aufruf mit der Methode *POST* bedeutet hingegen, dass Ressourcen auf dem Server angelegt werden sollen.

Null-Wert
Ein Null-Wert zeigt die komplette Abwesenheit von Daten an. Dies ist nicht mit einer leeren Zeichenkette oder dem numerischen Wert 0 zu verwechseln, da diese einen tatsächlichen Informationsgehalt besitzen. Ein Null-Wert ist somit eher als »unbekannter Wert« zu verstehen.

Offene Daten
Offene Daten sind Datenbestände, die ohne Einschränkung zur freien Nutzung, Weiterverwendung oder Weiterverarbeitung der Allgemeinheit öffentlich bereitgestellt werden.

Offene Verwaltungsdaten
Offene Verwaltungsdaten sind Datenbestände des öffentlichen Sektors, die von Staat und Verwaltung ohne Einschränkung zur freien Nutzung, Weiterverwendung oder Weiterverarbeitung der Allgemeinheit öffentlich bereitgestellt werden.

Open Data
Siehe »Offene Verwaltungsdaten«.

PascalCase
Leer- und Sonderzeichen in Bezeichnern können die Datenverarbeitung erschweren. Wenn Bezeichner aus mehreren Wörtern bestehen, wird daher empfohlen, diese Wörter zu einem zusammenzufassen. In der PascalCase-Schreibweise werden dabei die Anfangsbuchstaben eines jeden Wortes großgeschrieben, um die menschliche Lesbarkeit zu erleichtern. Dies ist unabhängig von der Wortart, d.h. auch Verben und Adjektive beginnen mit einem Großbuchstaben.

Primärdaten / Sekundärdaten
Primärdaten sind Rohdaten oder Basisdaten, die nach ihrer Erhebung nicht akkumuliert oder bewertet wurden. Teilweise ist es jedoch notwendig, die Rohdaten vor der Veröffentlichung zu bearbeiten, beispielsweise um rechtliche Vorgaben (Datenschutz) zu erfüllen. Werden Primärdaten weiterverarbeitet, bspw. durch Aggregation, Generalisierung, Interpretation oder Klassifizierung, spricht man von Sekundärdaten.

RDF
Das *Resource Description Framework* ist ein Modell zur Speicherung von Daten und Metadaten. Darin werden miteinander verknüpfte Daten in Form von Tripeln gespeichert.

Referenzsystem WGS-84
Das Referenzsystem WGS-84 ist die geodätische Grundlage zur Vermessung der Erde. Es dient zu Bestimmung von Positionsangaben und ist ein Standard für Satellitennavigation einschließlich GPS.

Relation
Mithilfe von RDF verknüpfte Dateneinheiten werden Tripel genannt. Eine Relation beschreibt den Zusammenhang (die Beziehung) zwischen Subjekt und Objekt eines Tripels.

Ressource
Als Ressource wird im Kontext von RDF eine Dateneinheit bezeichnet, die mit anderen Ressourcen in Beziehung stehen kann. Sie ist dafür üblicherweise eindeutig referenzierbar. Subjekt und Prädikat sind Ressourcen, das Objekt kann entweder Ressource oder Literal sein.

REST
Representational State Transfer ist ein Paradigma zum Entwurf von Schnittstellen. Dabei wird jede Ressource über eine eindeutige, hierarchisch aufgebaute URL zugänglich gemacht. Was mit der entsprechenden Ressource getan werden soll, wird mittels der verwendeten HTTP-Methode bestimmt.

Rohdaten
Siehe »Primärdaten«.

Server
Ein Server, z.D. Anbieter, bezeichnet üblicherweise einen Dienst im Internet, der Ressourcen bereitstellt. Diese Ressourcen können üblicherweise über eine API abgefragt werden.

Simple Feature Access
Simple Feature Access ist eine Spezifikation des Open Geospatial Consortium, welche die Speicherung und den Zugriff auf geografische Daten und Geometrien beschreibt.

SPARQL
Die *SPARQL Protocol And Query Language* ist eine graphenbasierte Abfragesprache für RDF. Mithilfe von SPARQL können Ressourcen, die in RDF als Tripel gespeichert sind, abgefragt werden.

Statuscode, HTTP
Ein HTTP-Statuscode ist ein standardisierter, numerischer Wert, der Auskunft über den Erfolg einer HTTP-Anfrage gibt. Alle Werte innerhalb bestimmter Zahlenbereiche haben dabei eine ähnliche Bedeutung, während die konkreten Zahlen eine genauere Differenzierung erlauben. So stehen alle Codes im Bereich von 400 bis 500 für den Misserfolg aufgrund von Fehlern auf der Clientseite. Der Code 403 zeigt beispielsweise an, dass die Anfrage nicht autorisiert war, während 404 anzeigt, dass eine Ressource nicht verfügbar ist.

Streaming
Streaming beschreibt einen Übertragungskanal von Daten. Dabei werden Daten in einzelnen Datenblöcken oder Bytes über das Internet übertragen. Im Gegensatz zur herkömmlichen Datenübertragung müssen die Daten nicht lokal heruntergeladen und gespeichert werden.

Sunlight Foundation
Die Sunlight Foundation ist eine gemeinnützige US-amerikanische Organisation, die sich für die Offenheit, Transparenz und Verantwortlichkeit des Regierungs- und Verwaltungshandelns unter Einsatz moderner Informations- und Kommunikationstechnologien einsetzt.

Tag
Als *Tag* wird in XML die Auszeichnung einer Dateneinheit bezeichnet. Ein in spitzen Klammern eingefasstes Schlüsselwort markiert dabei den öffnenden Tag (<beispiel>). Dasselbe Schlüsselwort, mit einer vorangehenden spitzen Klammer und einem Slash sowie einer schließenden spitzen Klammer eingefasst, markiert den schließenden Tag (</beispiel>).

Tripel
Als Tripel wird in RDF die Kombination aus Subjekt, Prädikat und Objekt bezeichnet. Diese Kombination stellt eine Sinneinheit dar. Daten werden in RDF stets in Form von Tripeln gespeichert. Die entsprechende Datenbank wird Tripel-Store genannt.

URI
Ein *Uniform Resource Identifier* ist eine möglichst eindeutige Referenz auf eine Ressource. Sie kann aus Buchstaben und/oder Zahlen bestehen, Leerzeichen sind nicht erlaubt. Eine URI kann direkt auf den Ort der Ressource verweisen, beispielsweise bei Verwendung einer Netzwerkadresse (URL).

URL
Ein *Uniform Resource Locator* ist eine Unterart der URI. Im Gegensatz dazu verweist eine URL immer auf eine auffindbare Ressource, ist also Identifikator und Adresse zugleich. Internetadressen oder Email-Adressen sind beispielsweise URLs.

UTF-8 Zeichenkodierung
UTF-8 ist eine weit verbreitete Art der Computer-internen Darstellung von Schriftzeichen. Besonders im Zusammenhang mit Sonderzeichen und speziellen Schriftzeichen, etwa Umlauten, wird durch diese Art der Speicherung eine größtmögliche Kompatibilität zu anderen Programmen gewährleistet.

Vektordaten
Vektordaten sind strukturierte Geodaten und werden zur Beschreibung raumbezogener Objekte verwendet.

Versionierung
Versionierung wird zur Unterscheidung verschiedener Bearbeitungsstände verwendet. So können beispielsweise Elemente wie Daten, Dokumente, Programme etc., die sich häufig ändern, mit Versionsnummern versehen werden. Dadurch werden Änderungen dokumentiert, die neueste Version kann einfach identifiziert und auf ältere Zustände zurückgegriffen werden.

Vokabular
Bezogen auf RDF definiert ein Vokabular, welche Relationen abgebildet werden können. Das Vokabular bestimmt somit den Geltungsbereich der zu speichernden Daten.

Wurzelelement (XML)
XML-Dokumente weisen eine Baumstruktur auf. Das Wurzelelement folgt in einem XML-Dokument direkt nach der XML-Deklaration mit einem öffnenden Tag. Am Ende des Dokuments folgt der schließende Tag des Wurzelelements. Das Wurzelelement umschließt alle Elemente und enthält somit die eigentlichen Daten des Dokuments. Jedes XML-Dokument besitzt genau ein Wurzelelement.

XML
XML (Extensible Markup Language) ist ein Format zur hierarchischen Strukturierung von Daten. XML wurde entwickelt, um plattformübergreifend Daten austauschen zu können.

Gefördert durch:

Bundesministerium
für Wirtschaft
und Energie

aufgrund eines Beschlusses
des Deutschen Bundestages

KONTAKT

Lina Bruns
Digital Public Services
Tel. +49 30 3463-7283
Fax +49 30 3463-99 7283
lina.bruns@fokus.fraunhofer.de

Fraunhofer FOKUS
Kaiserin-Augusta-Allee 31
10589 Berlin

www.fokus.fraunhofer.de

Wir
vernetzen
alles

sge | 1910 (Foto: pixabay)