

Universität Leipzig
Fakultät für Mathematik und Informatik
Wintersemester 2021/22
Bachelorarbeit

Transformation von Daten des Amtlichen Liegenschaftskatasterinfor- mationssystem in semantische Datenformate

Betreuer: Prof. Dr. Hans Gert Gräbe, Dr. Christian Zinke-Wehlmann, Norman Radke

Autor: Robin Seidel

Abgabedatum: .03.2022

Inhaltsverzeichnis

1	Einleitung	1
1.1	Themenbeschreibung/Zusammenfassung	1
1.2	Motivation	1
2	Grundlagen	3
2.1	ALKIS	3
2.2	ALKIS Daten	4
2.3	RDF	7
2.4	RML & Yarrml	8
2.5	SPARQL	8
2.6	Tripelstores	8
2.7	RDF vs XML	8
2.8	Abkürzungen	9
3	State of the Art	10
3.1	Problembeschreibung	10
3.2	Möglichkeiten der Transformation	10
4	Anforderungsanalyse	11
5	Implementierung	11
5.1	genutzte Technologien	16
5.2	Funktionsweise	16
6	Ergebnisse und Evaluation	16
7	Ausblick	16
8	notizen	17
9	Quellenverzeichnis	18

1 Einleitung

1.1 Themenbeschreibung/Zusammenfassung

1.2 Motivation

Mit der Einführung des World Wide Webs im Jahre 1990 durch Tim Burner Lee ist das Internet heute viel mehr als das, was es früher einmal war. Der Mensch nutzt das Internet immer häufiger als Hauptinformations- und Interaktionsmöglichkeit über diverse Unterhaltungskanäle wie Facebook, Youtube oder andere Websites. (QUELLE) Während am Anfang lediglich ein paar Dutzend Webseiten existierten sind es mittlerweile über zwei Milliarden mit 100 Milliarden Webdokumenten, die sich wiederum alle 6 Monate verdoppeln. (vgl. Meinel 2021) Aber es entstehen nicht immer nur mehr Webseiten auch die Anzahl der Internetnutzer hat sich in den letzten Jahren drastisch gesteigert. 1995 hat der Spiegel 250.000 Nutzer in Deutschland geschätzt, während 2018 durch das statistische Bundesamt ein Anstieg auf 90 wurde. (vgl. Köllinger 2003 zitiert nach: Der Tagesspiegel vom 19. August 1995, S. 23) Demnach nutzen mittlerweile 66,5 Millionen Menschen ab 10 Jahren das Internet als Informations- und Interaktionsquelle. (vgl. Statistisches Bundesamt 2018) Bei einer solchen Masse an Informationen und Daten wird es immer schwerer relevante Informationen herauszufiltern, obwohl das Hauptziel des Internets einst nicht die Unterhaltung, sondern die Informationsbeschaffung war. (QUELLE) Deshalb ist es wichtig die vorhandenen Daten mit Hilfe von RDF Graphen in einer geordneten Art und Weise zur Verfügung zu stellen. Das semantische Web war bereits eine Idee des Entwicklers/Gründers Tim Burner Lee. Er hatte die Vision, dass das Netz von Informationen nicht nur Dokumente und Daten miteinander verbindet, sondern das diesen Daten auch eine Bedeutung gegeben wird. Damit ermöglicht man einer Maschine die Anhäufung von Daten automatisiert nutzen zu können.

- beschreibung semantik web und wichtigste Prinzipien - 4 prinzipien: tim berner lee - nutzen von URI's als namen für objekte - zur eindeutigen identifikation für dinge - nutze HTTP URIs - so können Menschen die Objekte nachschauen - nutze anerkannte standarts(RDF, Sparql) - linke URIS weiter Das semantische Web ist eine Art rohe(also unearbeitete) Daten zuübertragen. (Interoperabilität) Um die Informationen die das Web liefert in einen strukturierten Weg wiederzugeben. Mit Hilfe der daraus entstehenden strukturierten Daten kann die Software diese Daten anschließend besser nutzen und verarbeiten. Das semantische Web ist damit sozusagen eine Alternative zu Webseiten die ausschließlich für die menschliche Nutzung geeignet sind. Die Vorteile die eine Semantifizierung der Daten mit sich bringt sind unter anderem das Abfragen durch Sparql Querys, eine Verbesserung der Bedeutung der Daten, das Verbinden von Daten und die Nutzbarkeit für andere Bereiche.

- überleitung auf Alkis Thematik Alkis Daten(siehe Begriffsklärungen) werden zwar in gewissen Standart und Einheitlichkeit erhoben, allerdings ist die Verarbeitung dieser

Daten hauptsächlich für GeoProgramme(näher erklären) vorgesehen. Die Art der Daten ist nicht für die optimale Verarbeitung von ComputerProgrammen und zur Nutzung für den gewöhnlichen Nutzer vorgesehen. Eine Überführung in Rdf Graphen ermöglicht es das Potenzial der Daten freizuschalten und die Nutzung für zahlreiche andere Bereiche zur Verfügung zu stellen. Quellen: (<https://www.spektrum.de/kolumne/semantic-web-wie-das-internet-inhalte-verstehen-koennte/> 1841389) paper über tim terner lees arbeit https://www.researchgate.net/publication/307845029_Tim_Berners-Lee_s_Semantic_Web paper von burnier über das semantic web 10.3.2022 <https://www.w3.org/2000/Talks/0906-xmlweb-tbl/text.htm> statistik bundesamt https://www.destatis.de/DE/Presse/Pressemitteilungen/2018/09/PD18_330_634.html Köllinger, Philipp <https://www.econstor.eu/handle/10419/151237> (Verlag und Jahr noch angeben)

2 Grundlagen

Hier werden die wichtigsten Konzepte erklärt, die in dieser Arbeit Anwendung gefunden haben.

2.1 ALKIS

Das Amtliche Liegenschaftskatasterinformationssystem, kurz ALKIS, ist ein Teil Des AAA-Modells(AFIS-ALKIS-ATKIS Modell). AFIS(Amtliches Festpunktinformationssystem) ist ein Raumbezugssystem, indem Informationen zu Festpunkten modelliert werden. In ATKIS(Amtliche Topographisch-Kartographische Informationssystem) geht es um digitale Karten zu Landschaften und Topografie. (seite 8 geoinfodoc). ALKIS umfasst die für uns relevanten Informationen aus dem Liegenschaftsbuch und Karte sowie Punktnachweisen. Die Verwaltung dieses AAA- Modells erfolgt durch die ADV(Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland). Das Modell ist seit 2015 in allen Bundesländern aktiv. Ziel der Entwicklung eines solchen Systems war es eine einheitliche und fachübergreifende Nutzung von Geodaten zuermöglichen. (seite 2 geoinfodoc). Bei der Erschaffung des Datenmodells für das System wurden Standards und Normen verwendet die vom W3C(World Wide Web Consortium), OGC(open geospatial consortium) und von der ISO(internationalen Organization for Standardization) anerkannt sind. Durch die Einführung von ALKIS wurden alle Daten des Liegenschaftskatasters redundanzfrei zusammengeführt. (quelle s 10) Kataster- und Liegenschaftsinformationen sind Geobasisdaten über sogenannte Liegenschaften, also zum Beispiel Flurstücke oder Gebäude. Diese Geobasisdaten werden durch die Vermessungs- und Katasterverwaltungen der Länder erhoben. Bereits vor der Veröffentlichung geobasierter Daten über ALKIS wurden Daten über das automatische Liegenschaftsbuch (Textform) und die automatische Liegenschaftskarte (geographischer Teil) gesammelt. Der Nachteil liegt darin, dass teilweise technisch veraltete Konzepte, verschiedene Datenformate, getrennte und teilweise redundante Datenhaltung aufeinander treffen und zu dem länderspezifisch implementiert wurden. (Quelle s8 13.3) vorteile von alkis gegenüber vorgängersystem(seite11)

<https://www.adv-online.de/AdV-Produkte/Liegenschaftskataster/ALKIS/broker.jsp?uMen=e5670f15-8e71-3c01-e1f3-351ec0023010>
<https://www.adv-online.de/AdV-Produkte/Liegenschaftskataster/ALKIS/>

11.3.22 ADV Online - ALkis Beschreibung

2.2 ALKIS Daten

Die Bestandsdaten aus ALKIS, können über eine sogenannte Normbasierte Austausch-schnittstelle(NAS) abgerufen werden. Diese wurde speziell für den Austausch von Daten des AAA - Models vom ADV entwickelt. Es werden als Standards unter anderem XML als Datenaustauschformat verwendet und zum Abrufen der Alkis Objekte ein sogenannter WFS(Web Feature Server)oder auch ein WMS(Web Map Server). Diese vom OGC(Open Geospatial Consortium Inc) definierten Geodatendienste sind ein Teil der NAS Spezifikation. Geodatendienste sind standartisierte Services die über eine URL aufgerufen werden und zum Austausch von Daten genutzt werden. (https://geodatenonline.bayern.de/geodatenonline/seiten/wms_webdienste 20.3.2022 Die Hauptaufgabe eines WMS ist die Visualisierung der Geodaten, er ist ein Darstellungsdienst. Grundlage eines WMS sind Vektor und Rasterdaten. Das Ergebnis(Response) einer Anfrage(Request) auf einen WMS ist ein Kartenausschnitt. Der WFS ist ein Download Service. Ihm zugrunde liegen die Rohdaten in Form der vom ADV definierten Objekte. Beim WFS bleibt die volle Struktur der Daten erhalten und die Daten werden in vollem Ausmaß bereit gestellt. Und sind somit optimal zur Weiterverarbeitung geeignet. Um an die Daten zukommen die für das Projekt benötigt werden, wird der WFS genutzt. Diese Schnittstelle besitzt drei verschiedene Abfrageoptionen. Die Basis URL ist für das Beispiel NRW: https://www.wfs.nrw.de/geobasis/wfs_nw_alkis_vereinfacht?. An diese URL können nun verschiedene Parameter angehängt werden um die richtige Antwort(Response) des Servers zu erhalten. Der WFS hat verschiedene Abfrageoptionen. Mit Request="GetCapabilities"werden alle Fähigkeiten und Metadaten des aufgerufenen Dienstes abgerufen. Durch "Request=DescribeFeatureType" wird die Struktur eines FeatureTypes abgefragt. Die wichtigste Abfrageoption ist Request="GetFeature". Eine komplette Anfrage(Request) an den WFS könnte so aussehen: https://www.wfs.nrw.de/geobasis/wfs_nw_alkis_vereinfacht?Service=WFS&REQUEST=GetFeature&VERSION=2.0.0&Typenames=Flurstueck&Countasdfasdfasdfasdf=10. Die Anfrage ist ein HTTP Get Call und beinhaltet neben der Basis URL verschiedene Parameter die erforderlich(Beispiel: Version, Typenames) oder optional(Beispiel: Count, Resulttype) sein können. Da der WFS ein Downloaddienst ist, erhält man als Response ein XML Dokument das die abgefragten ALKIS Objekte enthält. Eine Übersicht über alle möglichen Objekte findet man im Objektartenkatalog Seite 22. Das für uns wichtigste Objekt ist das AX_Flurstueck. Laut Definition(Objektartenkatalog seite 33) ist ein Flurstück ein Teil der Erdoberfläche, das mit einer Grenzlinie umschlossen ist. Das Objekt AX_Flurstueck besitzt die Attribute zustaendigeStelle, gemarkung, flurstuecksnummer, flurstueckskennzeichen, amtlicheFlaeche, flurnummer, und die Relationen istGebucht, zeigtAuf, weistAuf, gehoertAnteiligZu, beziehtSichAufFlurstueck etc..

Die Veröffentlichung der ALKIS Daten erfolgt über Schnittstelle,n die von den je-

weiligen Bundesländern angeboten werden. Einige Bundesländer haben jedoch gewisse Zugriffsbeschränkungen. 7 Bundesländer haben zumindestens einige Geodatenätze frei zugänglich gemacht. Sie unterliegen der Open Data Lizenz: "Datenlizenz Deutschland - Namensnennung- Version2.0"<https://www.govdata.de/dl-de/by-2-0> Andere WFS Schnittstellen sind nur durch Anmeldung und oder mit Kosten verbunden. (siehe Tabelle)(<https://learn.opengeoedu.de/opendata/vorlesung/offene-geodaten-amtliche-geodaten>

<https://www.adv-online.de/icc/extdeu/binarywriterservlet?imgUId=8f830072-8de8-9221-d5ad-8f138a438ad1&uBasVariant=11111111-11111111-11111111-11111111>
geoInfoDoc

https://www.lgln.niedersachsen.de/download/126790/Basiswissen_ALKIS_ETRS89_Schulungsmaterial_Stand_12.04.2010.pdf bAsiswissen
alkis 10.3

https://www.bezreg-koeln.nrw.de/brk_internet/geobasis/webdienste/anleitung_wfs.pdf

Bundesland	Zugriffsart	Zugriffsbeschränkung	Endpoint
NRW	WFS	keine	siehe Quellenverzeichnis
Berlin	WFS	keine	siehe Quellenverzeichnis
Thüringen	WFS	auth notwendig	siehe Quellenverzeichnis
Brandenburg	WFS	keine	siehe Quellenverzeichnis
Mecklenburg-Vorpommern	WFS	keine	siehe Quellenverzeichnis
Hamburg	WFS	keine	siehe Quellenverzeichnis
Sachsen	Download	keine	siehe Quellenverzeichnis
Sachsen-Anhalt	WFS	kostenpflichtig	siehe Quellenverzeichnis
Bayern	WFS	auth notwendig	siehe Quellenverzeichnis
Niedersachsen	WFS	kostenpflichtig	siehe Quellenverzeichnis
Hessen	WFS	keine	siehe Quellenverzeichnis
Rheinland-Pfalz	WFS	kostenpflichtig	siehe Quellenverzeichnis
Schleswig-Holstein	WFS	auth notwendig	siehe Quellenverzeichnis
Baden-Württemberg	WFS	auth notwendig	siehe Quellenverzeichnis
Bremen	nicht auffindbar		siehe Quellenverzeichnis
Saarland	WFS	auth	siehe Quellenverzeichnis

2.3 RDF

Das Resource Description Framework kurz RDF, ist ein vom W3C konzipiertes Datenformat zum Beschreiben von Objekten im Internet. <https://www.w3.org/TR/rdf-concepts>

RDF ist ein universelles, maschinenlesbares Format, das für den Austausch und der Darstellung von Daten gedacht ist. Original wurde es genutzt um Metadaten zu beschreiben (Beleg) und später wurde es in vielen verschiedenen Gebieten und generellen Anwendungen benutzt. Heute gilt es als grundlegender Baustein des Semantischen Webs. Die Daten, die im RDF Format vorliegen, sind als Graph angeordnet. Ein Graph besitzt Kanten und Knoten. Ein RDF Graph ist eine Ansammlung von Triples, Tripel sind Aussagen über Dinge. Ein Tripel besteht aus einem Subjekt, was konkret eine URI oder eine blank node sein kann, einem Prädikat was in dem Tripel die Eigenschaft widerspiegelt. Weiterhin besteht das Tripel aus einem Objekt. Ein Beispiel wäre:

```
Author hatName Robin
      wohntIn Haus
            type    Altbau
            in      Leipzig
```

In diesem Beispiel ist `ex:RobinSeidel` das Subjekt, `a` das Prädikat und `foaf:Person` das Objekt.

RDF ist ein Datenmodell was viele verschiedene Notationen/Serialisierungen besitzt. Es gibt also verschiedene Schreibweisen. Die ursprüngliche Notation ist RDF/XML, welche sich durch seine Bekanntheit und Austauschbarkeit auszeichnet.

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF xmlns:gndo="http://d-nb.info/standards/elementset/gnd#">
  <rdf:Description rdf:about="http://d-nb.info/gnd/118540238">
    <gndo:placeOfBirth rdf:resource="http://d-nb.info/gnd/4018118-2"/>
  </rdf:Description>
</rdf:RDF>
```

Sie ist jedoch im Vergleich zu Turtle eher schlecht durch Menschen lesbar. Turtle zeichnet sich dadurch aus, dass es wenig Speicherplatz benötigt und für den Menschen intuitiv lesbar ist.

```
@prefix gndo: <http://d-nb.info/standards/elementset/gnd#> .
```

```
<http://d-nb.info/gnd/118540238>
  gndo:placeOfBirth <http://d-nb.info/gnd/4018118-2> .
```

2.4 RML & Yarrml

Die RDF Mapping Language(RML)<https://rml.io/specs/rml/> ist eine Sprache, mit der man Regeln aufstellt um Datenstrukturen die in verschiedenen Formaten vorliegen in RDF umzuwandeln. RML beruht und erweitert die vom W3C definierte(?) Sprache R2RML<https://www.w3.org/TR/r2rml/> die dafür genutzt wird um ein spezielles Mapping anzufertigen um Daten aus einer relationalen Datenbank in ein RDF Datenset umzuwandeln. Sowohl R2RML Mappings und RML Mappings sind RDF Graphen und werden in Turtle geschrieben. Mit RML ist es möglich Daten aus heterogenen Datenquellen auf das RDF Format zu mappen. RML ist keine offiziell bestätigter Standart, es erweitert nur R2RML, zur Anwendung auf weitere Daten Strukturen und nicht nur relationale Datenbanken. Da in dieser Arbeit Daten im XML Format umgewandelt werden, ist die Verwendung von RML Regeln zu bevorzugen. Yarrml <https://rml.io/yarrml/spec/> wiederum ist eine Sprache, die auf Yaml <https://yaml.org/spec/1.2.2/> beruht. Sie ist dafür gemacht sehr gut menschen lesbar zu sein und wird dafür genutzt um auf einfache Art und Weise deklarative Regeln für das Mapping von verschiedenen Datenquellen auf RDF zu ermöglichen. Yarrml hat also den gleichen Zweck wie RML, nur das es für den Menschen einfacher ist die Mapping Regeln zu erstellen. Somit wird die Effektivität RDF Graphen zu erstellen erhöht.

2.5 SPARQL

Erklärung von SPARQL eventuell noch hier

2.6 Tripelstores

Erklärung von Triplestores eventuell noch hier

2.7 RDF vs XML

Nun könnte man sich die Frage warum sollte man RDF nutzen, wenn dies ohnehin schon auf XML beruht und die ALKIS Daten schon in diesem universellem Format vorliegen. Was ist der Vorteil von Rdf gegenüber anderen Datenaustauschformaten? Eine Aussage, kann in XML durch verschiedene Arten ausgedrückt werden. Für den Menschen der Sie liest bedeuten Sie das Gleiche. Für eine Maschine/Computer sind es verschiedene XML Bäume. In RDF wird die Aussage: der Autor der Bachelorarbeit ist Robin Seidel als Triple dargestellt. `triple(Autor, Bachelorarbeit, RobinSeidel)` in XML kann diese Aussage auf verschiedene Art und Weise dargestellt werden

```
<autor>
<von>bachelorarbeit</von>
```

```
<name>Robin Seidel </name>
</autor>
oder auch:
<bachelorarbeit>
<autor>Robin Seidel </autor>
</bachelorarbeit>
```

Für einen Menschen der diese XML Dokumente liest, haben diese die gleiche Bedeutung. Jedoch für eine Maschine sind es zwei verschiedene XML Bäume. Dies zeigt das die Nutzung von Technologien des Semantischen Webs, sehr wertvoll sein können. <https://www.w3.org/DesignIssues/RDF-XML.html> <https://dbs.uni-leipzig.de/html/seminararbeiten/semSS99/arbeit5/Rdf.html>

2.8 Abkürzungen

- ALKIS: Amtliches Liegenschaftskatasterinformationssystem
- RML: RDF Mapping Language
- RDF: Resource Description Framework
- XML: eXtensible markup language
- ADV: Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland
- URL: Uniform Resource Locator
- URI: Uniform Resource Identifier
- NAS: Normbasierte Austauschchnittstelle

3 State of the Art

3.1 Problembeschreibung

(Allgemeine Beschreibung des Problems der Bachelorarbeit) ohne konkrete Implementierung der Lösung zu beschreiben. Der ETL-Prozess (Extract, Transform, Load) ist ein typisches Vorgehen in der Welt von Big-Data. Ziel dieses Prozesses ist es, die Daten aus verschiedenen Datenquellen für eine weitere Verarbeitung vorzubereiten und wieder bereitzustellen. Auch bei diesem Problem (der Transformation von ALKIS Daten in semantische Formate) findet er Anwendung. Das vorliegende Problem ist folgendermaßen aufgebaut. Der Extract (auf deutsch extrahieren) Teil des Prozesses ist es, die notwendigen Alkis Daten zusammenzufassen. Dies gelingt durch HTTP-Requests an die Schnittstellen, die von den Behörden der Bundesländer bereitgestellt werden. (siehe Tabelle 2.2) Durch gezielte Angabe der richtigen Parameter erhält man so die notwendigen Alkis Rohdaten. Der Teilprozess der Transformation enthält wiederum mehrere Einzelschritte. Ziel der Transformation ist es, die Daten in ein semantisches Format so umzuwandeln, dass sie verknüpft sind und durch eine SPARQL Schnittstelle abgerufen werden können. Die ALKIS Daten müssen nun mithilfe von Mapping Regeln so konstruiert werden, dass sie am Ende ein möglichst nützliches und strukturiertes Graphen ergeben. Im Load Teil des Prozesses werden die Daten dann in einen Triple Store geladen. Dieser dient als Datenbank für Graphen. Die transformierten RDF Daten können nun über SPARQL Abfragen abgerufen werden.

<https://www.bigdata-insider.de/was-ist-etl-extract-transform-load-21.3>

3.2 Möglichkeiten der Transformation

Zunächst war es nötig, die möglichen Herangehensweisen an das Projekt/Problem zu identifizieren. ALKIS Daten zum einen, gibt es als vielen verschiedenen Datensätzen. Es gibt Datensätze über zum Beispiel Gebäude, Bodenschätzung und über Flurstücke. Es gibt Datensätze ohne Zugriffsbeschränkung und welche, auf die nur durch Bezahlung zugegriffen werden kann. Die Art und Weise, wie die Daten aufgebaut sind und abgerufen werden, ist zwar durch den ADV geregelt worden, jedoch kommt es in den verschiedenen Bundesländern noch zu großen Unterschieden. (näheres siehe 2.2 ALKIS DATEN) Da es Ziel dieser Arbeit ist, einen Wissensgraphen über ALKIS Daten zu erstellen, wurde sich hier auf den Datensatz "Flurstuecke" konzentriert. Dieser enthält alle wichtigen Daten über die Liegenschaften und Flurstücke der jeweiligen Bundesländer. Zur Transformation von XML Dokumenten in RDF gibt es verschiedene Herangehensweisen.

- mapper speziell für Alkis Daten selber zu programmieren
- vorgefertigtes Tool nutzen

- welche Vorteile und nachteile hätte dies?

Bei der Abspeicherung der fertig umgewandelten RDF Graphen gibt es auch mehrere Möglichkeiten. Zur Speicherung von Graphen werden keine herkömmlichen relationalen Datenbanken genutzt. (Warum nicht erklären) Sondern speziell dafür konzipierte Graphen Datenbanken. Dadurch dass Beziehungen/Relationen der Graphen in so einer Datenbank abgespeichert sind. Müssen diese bei einer Abfrage nicht erst berechnet werden sondern stehen bereits zur Verfügung. Dadurch ergibt sich eine bessere Performance gegenüber relationalen Datenbanken. - neo4j - graphDB - fuseki Welche ist am besten geeignet für Anwendungsfall ALKISTransform <https://www.ionos.de/digitalguide/hosting/hosting-technik/graphdatenbank/#:~:text=Eine%20der%20St%C3%A4rken%20der%20Graph,Geschwindigkeit%20selbst%20bei%20komplizierten%20Abfragen.>

4 Anforderungsanalyse

5 Implementierung

- python
- konsolenanwendung
- prototyp

- wie wurde ETL Prozess umgesetzt

Download raw alkis Data von gewünschten Bundesland und Gemeindenummer

•

•

•

```
<AX_Flurstueck gml:id="DEHHALKAn00005Kq">
  <gml:identifier codeSpace="http://www.adv-online.de
/">urn:adv:oid:DEHHALKAn00005Kq</gml:identifier>
  <lebenszeitintervall>
    <AA_Lebenszeitintervall>
      <beginnt>2018-01-06T01:59:38Z</beginnt>
    </AA_Lebenszeitintervall>
  </lebenszeitintervall>
  <modellart>
    <AA_Modellart>
      <advStandardModell>DLKM</advStandardModell>
    </AA_Modellart>
  </modellart>
</AX_Flurstueck>
```

```

</modellart>
<anlass>300000</anlass>
<zeigtAufExternes>
  <AA_Fachdatenverbindung>
    <art>urn:x-gv-hamburg-de:AAA:FDV:autark2web</art>
    <fachdatenobjekt>
      <AA_Fachdatenobjekt>
        <name>6071992000029-VT_____ - _____ -UA_____
        </name>
      </AA_Fachdatenobjekt>
    </fachdatenobjekt>
  </AA_Fachdatenverbindung>
</zeigtAufExternes>
<zeigtAufExternes>
  <AA_Fachdatenverbindung>
    <art>urn:x-gv-hamburg-de:AAA:FDV:autark2web</art>
    <fachdatenobjekt>
      <AA_Fachdatenobjekt>
        <name>6072015282942-VK</name>
      </AA_Fachdatenobjekt>
    </fachdatenobjekt>
  </AA_Fachdatenverbindung>
</zeigtAufExternes>
<zeigtAufExternes>
  <AA_Fachdatenverbindung>
    <art>urn:x-gv-hamburg-de:AAA:FDV:autark2web</art>
    <fachdatenobjekt>
      <AA_Fachdatenobjekt>
        <name>6071999002894-VA_____ - _____ - _____
        </name>
      </AA_Fachdatenobjekt>
    </fachdatenobjekt>
  </AA_Fachdatenverbindung>
</zeigtAufExternes>
<position>
  <gml:Surface gml:id="DEHHALKAn00005Kq_G0"
  srsName="urn:adv:crs:ETRS89_UTM32">
    <gml:patches>
      <gml:PolygonPatch interpolation="planar">

```

```
<gml:exterior>
  <gml:Ring>
    <gml:curveMember>
      <gml:Curve gml:id="DEHHALKAn00005Kq_G1">
        <gml:segments>
          <gml:LineStringSegment>
            <gml:posList>577193.020 5917919.522
              577203.678 5917937.753</gml:posList>
          </gml:LineStringSegment>
          .
          .
          .
        </gml:segments>
      </gml:Curve>
    </gml:curveMember>
  </gml:Ring>
</gml:exterior>
```

Auszug aus alkis_vereinfacht.xml (nrw)

Regeln zur Transformation in .yarrml Format

prefixes:

alkis: <https://w3id.org/alkis/>

gml: <http://www.opengis.net/gml/3.2/>

mappings:

alkis_quelle:

sources:

- access: D://repos/Uni/Ba/ALKIS-transform/TestData/wfs_capabilities
- referenceFormulation: xpath
- iterator: WFS_Capabilities/ServiceIdentification

subjects: [http://www.example.com/quelle/\\${Title}](http://www.example.com/quelle/${Title})

predicateobjects:

- [a, alkis:Quelle]

alkis_flurstueck:

sources:

- access: D://repos/Uni/Ba/ALKIS-transform/TestData/dataFromWFS
- referenceFormulation: xpath
- iterator: FeatureCollection/member/Flurstueck

subjects: [http://www.example.com/\\${idflurst}](http://www.example.com/${idflurst})

predicateobjects:

- [a, alkis:Flurstueck]
- [alkis:flurstueckskennzeichen, \${flstkennz}]
- [alkis:gemeindezugehoerigkeitLand, \${land}]
- [alkis:landschluesel, \${landschl}]

- [alkis:gemarkung, \$(gemarkung)]
- [alkis:gemarkungschluessel, \$(gemaschl)]
- .
- .
- .

Enthält Regeln zur Umwandlung von .xml Dokumenten die das Schema alkis vereinfacht abbilden

-Transformationsregeln in .ttl Format (erstellt mithilfe yarrml-parser)

```
@prefix rr: <http://www.w3.org/ns/r2rml#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix fnml: <http://semweb.mmlab.be/ns/fnml#>.
@prefix fno: <https://w3id.org/function/ontology#>.
@prefix d2rq: <http://www.wiwiss.fu-berlin.de/suhl/bizer/D2RQ/0.1#>.
@prefix void: <http://rdfs.org/ns/void#>.
@prefix dc: <http://purl.org/dc/terms/>.
@prefix foaf: <http://xmlns.com/foaf/0.1/>.
@prefix rml: <http://semweb.mmlab.be/ns/rml#>.
@prefix ql: <http://semweb.mmlab.be/ns/ql#>.
@prefix : <http://mapping.example.com/>.
@prefix alkis: <https://w3id.org/alkis/>.
@prefix gml: <http://www.opengis.net/gml/3.2/>.

:rules_000 a void:Dataset;
    void:exampleResource :map_alkis_quelle_000.
:map_alkis_quelle_000 rml:logicalSource :source_000.
:source_000 a rml:LogicalSource;
    rml:source "D://repos/Uni/Ba/ALKIS-transform/TestData/wfs_capabil
    rml:iterator "WFS_Capabilities/ServiceIdentification";
    rml:referenceFormulation ql:XPath.
:map_alkis_quelle_000 a rr:TriplesMap;
    rdfs:label "alkis_quelle".
:s_000 a rr:SubjectMap.
:map_alkis_quelle_000 rr:subjectMap :s_000.
:s_000 rr:template "http://www.example.com/quelle/{Title}".
:pom_000 a rr:PredicateObjectMap.
:map_alkis_quelle_000 rr:predicateObjectMap :pom_000.
:pm_000 a rr:PredicateMap.
```



```
:pom_000 rr:predicateMap :pm_000.
:pm_000 rr:constant rdf:type.
:pom_000 rr:objectMap :om_000.
:om_000 a rr:ObjectMap;
    rr:constant "https://w3id.org/alkis/Quelle";
    rr:termType rr:IRI.
:rules_000 void:exampleResource :map_alkis_flurstueck_000.
```

Triples die vom rmlMapper ausgegeben werden im .ttl Format

[illegible]

→ Daten sammeln

- HTTP Get an WFS

- Umsetzung mit HTTP Library Requests"geschrieben in python

ODER für ALKIS Sachsen mit URLLIB, wird genutzt um beliebige URL aufzurufen und

deren response zu downloaden

→ Transformation

- Auswahl welche XML Datei umgewandelt werden soll
 - ausgewählte Datei wird in yarrmlRules Datei eingetragen
 - es erfolgt die Erstellung von RML Regeln mithilfe yarrml Parser und selbstgeschriebener yarrml Regeln
 - Aufruf RDFMapper der mit RML Regeln, die Objekte in der ausgewählten xml datei in RDF im .ttl Format umwandeln
- Daten Speichern
- die produzierten rdf tripel werden in fuseki graph db gespeichert

5.1 genutzte Technologien

rmlmapper: <https://github.com/RMLio/rmlmapper-java> Yarrml Parser: <https://github.com/RMLio/yarrml-parser> RDFLib: <https://github.com/RDFLib/rdfliib>

Welchen store sollte ich nutzen?

<https://www.ontotext.com/knowledgehub/fundamentals/what-is-rdf-triplestore/>
<https://www.youtube.com/watch?v=eDRfKAKZMQE>
<https://db-engines.com/en/system/GraphDB>

5.2 Funktionsweise

- github mit howto zur selbstaussführung
- beschreibung der Transformation mit codeausschnitten

6 Ergebnisse und Evaluation

- ergebnis mit beispielhaften SPARQL abfragen
- da alkis data auffindbarkeit und nutzbarkeit mangelhaft
- arbeit hat sich zum ziel gemacht dies zuverbessern

7 Ausblick

- weitere Alkis Quellen
- benutzermanagment

- anreicherung der rdf daten

8 notizen

- gml:id ist dokumentenweit eindeutig(durch entstehungsdatum-zeit)(quelle geoinfodoc s92)

- 4 prinzipien: tim berner lee - nutzen von URI's als namen für objekte - zur eindeutigen identifikation für dinge - nutze HTTP URIs - so können Menschen die Objekte nachschauen - nutze anerkannte standarts(RDF, Sparql) - linke URIS weiter

9 Quellenverzeichnis

Quellen für Alkis Endpunkte der Bundesländer:

NRW: https://www.wfs.nrw.de/geobasis/wfs_nw_alkis_vereinfacht?Service=WFS&Request=GetCapabilities

Berlin: https://fbinter.stadt-berlin.de/fb/wfs/data/senstadt/s_wfs_alkis?Service=WFS&Request=GetCapabilities

Thüringen: https://fbinter.stadt-berlin.de/fb/wfs/data/senstadt/s_wfs_alkis?Service=WFS&Request=GetCapabilities

Brandenburg: https://isk.geobasis-bb.de/ows/alkis_vereinf_wfs?Service=WFS&Request=GetCapabilities

Mecklenburg-Vorpommern: https://www.geodaten-mv.de/dienste/alkis_wfs_sf

Hamburg: https://geodienste.hamburg.de/WFS_HH_ALKIS_vereinfacht?REQUEST=GetCapabilities&SERVICE=WFS

Sachsen:

Sachsen-Anhalt:

Bayern: https://geoservices.bayern.de/wfs/v1/ogc_alkis_ave.cgi?

Niedersachsen: <https://numis.niedersachsen.de/www.lgln.niedersachsen.de?SERVICE=WFS&REQUEST=GetCapabilities>

Hessen: <https://www.gds.hessen.de/wfs2/aaa-suite/cgi-bin/alkis/vereinf/wfs?Service=WFS&Request=GetCapabilities>

Rheinland-Pfalz: <https://www.geoportal.rlp.de/spatial-objects/353>

Schleswig-Holstein: https://www.schleswig-holstein.de/DE/Landesregierung/LVERMGEOSH/lvermgeosh_node.html?REQUEST=GetCapabilities&SERVICE=WFS

Baden-Württemberg:

Bremen:

Saarland: <https://geoportal.saarland.de/registry/wfs/325?REQUEST=GetCapabilities&VERSION=1.1.0&SERVICE=WFS>

- 4.2.22

zu einleitungserklärungen

- neo4j vs graph db:

Objektenartenübersicht ALKIS: <https://www.adv-online.de/icc/extdeu/nav/a63/binarywriterservlet?imgUid=b001016e-7efa-8461-e336-b6951fa2e0uBasVariant=11111111-1111-1111-1111-111111111111> Objektartenkata-

log Seite: 22

GeoinfoDoc: <https://www.adv-online.de/GeoInfoDok/binarywriterservlet?imgUid=e9e304b7-96d3-de71-d261-dde30c21d06f&uBasVariant=11111111-1111isDownload=true>

WFS Spezifikation: file:///C:/Users/robin/Downloads/05-008c1_OGC_Web_Services_Common_Specification_Corrigendum.pdf