AP-1 DFKI: Begründungen

Georg Rehm, Elena Leitner, Melina Plakidis, Julian Moreno-Schneider

AP-1.1 Datennormalisierung

F1: Warum wurde das graphbasierte Datenmodell ausgewählt?

Auf einer abstrakten Ebene ist zwischen denjenigen Datenmodellen unterscheiden, die in der Graphdatenbank (GDB) zu speichern sind und denjenigen, die zu visualisieren sind. Im optimalen Fall sollte die Visualisierung unmittelbar aus der Datenbank ableitbar sind. Das Ziel des DFKI war es, ein generisches und einheitliches Datenmodell zu entwickeln, das die heterogene Menge von Ausgangsdaten abdeckt und es ermöglicht, die Daten in der GDB zu speichern.

• Die Ausgangsdaten (insbesondere GND) lassen sich als Knoten und Kanten modellieren. Die Gründe dafür sind wie folgt: Einzelne Entitäten werden als Einträge zusammengefasst. Die Entitätstypen sind vorgegeben (Person, Körperschaft, Ort etc.). Verknüpfungen zu anderen Entitäten und ihre Besonderheiten (Art der Verknüpfung, zeitliche Gültigkeit etc.) sind in den Ausgangsdaten als Verweise kodiert.[1] Die Relationen wurden über ihre jeweilige Feldnummer bzw. Paths nach Entitätstypen aufgeteilt. Diese Verknüpfungen werden als Relationen modelliert.

• In einem Workshop im November 2019 wurde das graphbasierte Datenmodell präsentiert und von den Partnern akzeptiert.[2]

• Im Rahmen dieses Workshops wurden im Rahmen einer Abstimmung mit den Partnern relevante Merkmale (Properties in der GDB) ausgewählt.

F2: Warum wurden die Daten nicht vollständig auf Inkonsistenzen geprüft?

Die Ausgangsdaten umfassen in gezippter Form 12,11 GB Material (ohne Angaben zu Volltexten). Es war weder möglich noch zielführend, alle kodierten Informationen automatisch auf Konsistenz zu prüfen. Es wurden die Codes und Relators geprüft, die in spezifischen Feldern verwendet werden.[3] Fehlermeldungen wurden in Logdateien geschrieben.[4]

F3: Warum sind im SoNAR-Datenbestand nicht alle Daten zu finden, die online in der GDB zu finden sind?

Es existieren im Wesentlichen zwei Gründe:

• Wir verwenden bzgl. GND, DNB, ZDB und KPE Datendumps aus dem Jahr 2019. Die Usprungsdaten wurden seit 2019 entsprechend gepflegt und aktualisiert. Da es sich bei SoNAR um eine Machbarkeitsstudie handelt, spielt die Aktualität der verwendeten Datenbasis nur eine untergeordnete Rolle.

• In der GND existieren Entitäten, bei denen mittels Verknüpfung verbundene Entitäten keine eindeutigen Identifikatoren besitzen. Diese können nicht eindeutig automatisch verarbeitet werden, weshalb sie nicht in die SoNAR-Daten integriert werden konnten.

---

[1] s. Ordner *„Dokumentation der Ausgangsdaten“*
[2] s. Dok *„Workshop zur Datenmodellierung“*

[3] s. Ordner *„Dokumentation der Ausgangsdaten“*
[4] https://github.com/sonar-idh/Transformer/blob/main/log/Readme.md

F4: Warum wurden im ETL-Prozess MARC21 und EAD als Inputformate ausgewählt?

Die SBB hat in der ersten Phase des Projekts mit diesen Datenformaten gearbeitet und Daten in diesen Formaten zur Verfügung gestellt. Andere Formate können berücksichtigt werden, wobei insbesondere die implementierte Prozesskette für die Datentransformation entsprechend anzupassen wäre.

F5: Warum wurde GraphML als Outputformat ausgewählt?

GraphML bietet die effizienteste Lösung für den Import und das Management großer Datenmengen mit einer komplexen Struktur auf der hochperformanten Graphdatenbank Neo4j an. Zum einen wäre es mit sehr viel Aufwand verbunden, eine komplexe Graphdatenstruktur mit vielen Merkmalen in einem Format wie z.B. CSV zu repräsentieren – GraphML ist dazu nativ geeignet. Die derzeit einzige Alternative stellt RDF dar, für das jedoch aktuell keine hochperformanten Datenbanken bzw. Triple-Stores zur Verfügung stehen. Ferner kann der Datenimport bei Neo4j mit anderen Formaten um ein Vielfaches länger dauern (z.B. Cypher). Hinsichtlich des Problems der veralteten Identifikatoren (siehe Frage F7) kann eine alternative Lösung erarbeitet werden.

F6: Warum wurde ein Regelset erarbeitet, das implizite Verknüpfungen zwischen Entitäten widerspiegelt? Sind die Ausgangsdaten nicht ausreichend? Sind die Ausgangsdaten falsch modelliert?

Das implementierte Regelset dient als Grundlage für die Analyse der Personennetzwerke, die in den Daten enthalten sind, allerdings oftmals nur implizit, so dass eine Explizierung dieser Relationen erfolgte, um maschinell auf die ausgedrückten Relationen zugreifen zu können. Die Ausgangsdaten ermöglichen es, auch weitere implizite Informationen basierend auf zusätzlich formulierten und implementierten Regeln zu explizieren, um weitere Relationen zu modellieren. Das aktuell implementierte Regelset5 ist als ein erster Test dieses Ansatzes zu verstehen.

F7: Gab es technische Herausforderungen bei der Transformation?

Die Adressierung der beiden folgenden Aspekte hat für Mehraufwände bei der Implementierung der Prozesskette gesorgt:

• Einige XML-Entitäten entsprachen nicht dem GraphML-Format und verursachten Fehlermeldungen beim Import in Neo4j. Dies wurde durch die Implementierung von Normalisierungsfunktionen adressiert.[6]

• In den Daten existierten veraltete oder fehlende Identifikatoren für Ausgangs- oder (meistens) Zielknoten, die bei der Beschreibung der Verknüpfungen verwendet wurden. Die Prüfung aller betroffenen Knoten war sehr zeitaufwendig. Für unseren Datendump wurde eine Liste[7] mit allen nicht validen Identifikatoren erstellt, die im Rahmen der Transformation verwendet wurde. Diese

[5] Dieses Regelset sollte in einem Folgeprojekt überarbeitet und ggf. ergänzt werden, da bislang bei Kalliope viele Genres nicht berücksichtigt wurden. Ferner wäre eine Übersicht über die möglichen Rollenattribute bei Kalliope sehr hilfreich

[6] https://github.com/sonar-idh/Transformer/blob/main/src/MarcTransform.py und https://github.com/sonar-idh/Transformer/blob/main/src/EadTransform.py

[7] https://github.com/sonar-idh/Transformer/tree/main/src

Lösung funktioniert in der Laborpraxis, skaliert aber nicht, d.h. für einen neuen Datendump müsste eine neue derartige Lookup-Tabelle erstellt werden. Es wäre zu eruieren, welche Alternativen existieren. Die Ausgangsdaten könnten z.B. knotenweise mit Cypher-Statements während der Transformation in Neo4j importiert werden.

F8: Wie lauten die Hauptergebnisse der Auswertung der Ausgangsdaten?

Die Auswertung der Ausgangsdaten hat folgende Hauptergebnisse geliefert:

• Informationenkodierung (in MARC21 existieren Präfixe, die die Bedeutung eines Feldes ändern können).

• Welche Informationen werden in welchen Feldern/Paths kodiert?

• Welche Felder/Paths werden hochfrequent verwendet?

• Welche Relationsarten werden hochfrequent verwendet (bzgl. Feldnummer)?

• Welche Relationsarten werden hochfrequent verwendet (bzg. Relators, d.h. interne Codes für Bezeichnung verschiedener Relationsarten)?

• Welche Felder/Paths sind marginal?

• Welche Informationen in Feldern können frei besetzt werden (stichprobenartig)?

• Welche Inkonsistenzen existieren in den Daten?

Diese Auswertung hat geholfen, sowohl die Daten zu modellieren als auch die Transformation der Daten durchzuführen.

AP 1.2 Datenanreicherung

AP 1.2 Abschnitt 1: Ergänzen identifizierter Entitäten

Erstellt: 23.07.2021

Ziel: Fehlende Datenbeschreibungen für eindeutig durch ISNI, VIAF, SNAC, Loc-NACO oder Wikidata identifizierte Entitäten werden im Datenbestand von SoNAR (IDH) maschinell (ex-post) ergänzt.

Beschreibung: Für eine Entität, die in Metadaten durch ISNI, VIAF, SNAC, LoC-NACO oder Wikidata identifiziert wird und für die durch UC2.2 keine GND-Repräsentation ermittelt werden kann, wird der Datensatz aus der referenzierten Normdatei via API übernommen und in den Datenbestand integriert.

Prozess: Um das Anreichern von Informationen von Entitäten zu testen, wurde folgendermaßen vorgegangen:

Personen, welche in den OCR-Zeitungsvolltexten erkannt wurden und durch eine Wikidata ID eindeutig identifiziert werden konnten, allerdings nicht mit der GND ID verlinkt wurden und somit entweder keine entsprechende GND Repräsentation existiert oder denen keine GND Repräsentation eindeutig zuzuordnen ist, wurden mit ausgewählten Wikidata Merkmalen angereichert. Zu diesen ausgewählten Merkmalen gehören das Geburts- und Sterbedatum, der Geburts- und Sterbeort (sowie die entsprechende Wikidata ID dieser Orte) und das Geschlecht der Person. Diese Merkmale wurden gewählt, da sich diese sehr gut automatisch abfragen lassen, einer sehr strikten Form folgen und

häufig[8] auftreten. Beispielsweise trat das Geschlechtsmerkmal in 996/1000 Fällen auf, das Geburtsdatumsmerkmal in 994/1000 Fällen und das Geburtsortsmerkmal in 935/1000 Fällen. Da vermutlich einige noch lebende Personen in der Stichprobe vorhanden waren, kamen die Merkmale für das Sterbedatum (591/1000 Fälle) und den Sterbeort (516/1000 Fälle) deutlich weniger häufig vor, dennoch aber beide in über der Hälfte der Abfragen.

Der Code, welcher für das Anreichern verwendet wurde, befindet sich auf GitHub[9]. Eine Darstellung des Datenmodells, welches angereicherte Informationen beinhaltet, ist in dem Dokument *„AP1 Begründungen“* (AP-1.2 Abschnitt 3) zu finden. In diesem Datenmodell werden Informationen, welche aus Wikidata stammen, gesondert durch eigenständige Merkmale gekennzeichnet (Merkmalsbezeichnungen bspw. “WdDateOfBirth”, “WdGender”, “WdDateOfDeath”).

AP 1.2 Abschnitt 2: Anreichern von Entitäten

Erstellt: 07.06.2021

Ziel: Beschreibungen über Personen, Familien und Körperschaften sollen angereichert werden.

Beschreibung: Daten über Personen, Familien und Körperschaften sollen ergänzt werden, wenn Werte im SoNAR-Datenbestand fehlen: Geburts- / Sterbedatum, Geburts- / Sterbeort, Berufe, Auszeichnungen und soziale Relationen. Hierzu werden Wikidata und SNAC regelmäßig abgefragt. Personen, Familien und Körperschaften, die mit den sozialen Beziehungen in Wikidata und SNAC identifiziert werden und keine Repräsentation in SoNAR haben, werden ergänzt (s. Ergänzung identifizierter Identitäten).

Abschließende Schlussfolgerung:

Eines der Hauptprobleme bei der semantischen Datenintegration aus heterogenen Datenquellen, in diesem Fall die Anreicherung der GND durch Wikidata, scheint zu sein, dass es vor allem schwer ist, Informationen nicht doppelt hinzuzufügen. Merkmale wie z.B. Preise oder Auszeichnungen sind in unserer Datenbank durch die GND schlecht/ chaotisch dargestellt, was eine Überprüfung nach dem Vorhandensein erschwert. Man könnte dies so lösen, indem man sich nur auf eine ausgewählte Zahl an Merkmalen beschränkt, welche sich gut überprüfen lassen (bspw. Geschlecht, Geburtsdatum, Geburts- und Sterbeort), und die restlichen Merkmale nicht zu übernehmen.

Ein weiterer Weg wäre, die Daten trotzdem alle (gekennzeichet als Wikidata (/SNAC)-Information) hinzuzufügen, auch wenn diese dann doppelt vorhanden sind. Allerdings schätze ich, dass dieser Weg nicht effizient wäre und die Datenbank noch viel chaotischer machen würde, als sie durch die GND ohnehin schon ist.

Zudem besteht beim Anreichern von Relationsmerkmalen das Problem, dass dafür die zugehörige Entität eindeutig identifiziert werden muss, was nicht immer möglich ist, wenn z.B. die Wikidata-Entität nicht mit der GND-Id verlinkt ist. Man könnte diese Entitäten trotzdem hinzufügen, allerdings würde das dazu führen, dass gleiche Entitäten potentiell mehrfach in der Datenbank vorhanden sind. Dies würde allerdings wiederum die Übersichtlichkeit der Datenbank erheblich beeinträchtigen und das Abfragen von Informationen enorm erschweren. Dementsprechend würde es sich empfehlen, beim

[8] Siehe Dok. *„wikidata_per_properties_1000“* für Statistiken, wie häufig welches Merkmal bei einer zufälligen Stichprobe von 1000 Personen auf Wikidata vorkam.
[9] https://github.com/sonar-idh/Transformer/tree/main/enrich

automatisierten Anreichern immer Entitätsmerkmale statt Relationsmerkmale zu wählen wenn möglich.

Spezifische Schlussfolgerung: Organisationen anreichern (Anhang A)

Die Namen der Organisationen in Wikidata (Label) und in der GND scheinen in der kleinen Stichprobe immer übereinzustimmen. Das Problem für das automatisierte Anreichern von Informationen durch Wikidata stellt meines Erachtens vor allem die Struktur der GND dar. In der kleinen Stichprobe von 10 Organisationen sind schon einige Informationen in Feldern, die dafür nicht vorgesehen sind bzw. nicht einheitlich dargestellt. Beispielsweise Erläuterungen: Definition: Sitz: Hamburg und Erläuterungen: Definition: Sitz des NATO-Hauptquartiers ist Brüssel oder Weitere Angaben: African Studies Association (ASA), Univ. of Calif., Los Angeles; founded 1957 . Das macht das Abgleichen, ob Werte schon in der Datenbank sind, so gut wie unmöglich.

Außerdem gibt es einige sehr abweichende Werte, wie z.B. bei den Entstehungsdaten. Aufgrund vorheriger Arbeit vermute ich, dass dies teilweise durch Namensänderungen der jeweiligen Organisationen zu begründen ist, wodurch es hierbei scheinbar wesentlich häufiger zu Unstimmigkeiten kommt als bei dem Entitätstyp *Person.*

Falls das automatisierte Anreichern trotzdem erwünscht ist, sollte zu bedenken sein, dass die Anzahl verlinkter Organisationen nur sehr gering ist (bei einer Stichprobe von 1000 Organisationen unserer Datenbank waren nur rund 12% der Entitäten mit der Wikidata-Id verlinkt). Auch wenn man sich hier nur auf eine bestimmte Auswahl an Merkmalen für das Anreichern begrenzen würde, schätze ich den Aufwand, damit es nicht zu Dopplungen kommt, als relativ hoch ein. Es stellt sich die Frage, ob es sinnvoll ist, für einen Zuwachs an Informationen geringere Übersichtlichkeit und Effizienz der Datenbank in Kauf zu nehmen.

Spezifische Schlussfolgerung: Personen anreichern (Anhang B)

Es ist nicht möglich, den Namen automatisch immer fehlerfrei in das Format der GND *Nachname, Vorname* zu bringen. Bei Wikidata sind nicht immer die Merkmale *given name, family name* angegeben, sodass es nicht ersichtlich ist, welcher der Vor- und der Nachname ist. Selbst wenn die Merkmale angegeben sind, kommt es vor, dass bei mehreren Vornamen nur ein Name genannt ist, oder gar mehr als bei dem Label angegeben wurde. Auch wenn bei den Vornamen alle im Merkmal *given name* genannt wurden, ist nicht immer auch die Reihenfolge der Vornamen angegeben. Bei der Kategorie *Andere Namen* ist es genauso unmöglich, die Namen in dasselbe Format zu bringen, da bei Wikidata die Spalte *Also known as* teilweise komplett chaotisch ist, d.h. Namen in den unterschiedlichsten Formaten beinhaltet.

Schon bei der kleinen Stichprobe traten außerdem Abweichungen der einzelnen Werte, bspw. bei dem Geburtsdatum oder dem Geburtsort auf. Außerdem sind bei Wikidata teilweise mehrere Geburts-/ Sterbedaten angegeben. Hier stellt sich die Frage, ob die Werte entweder immer (gekennzeichnet) übernommen werden sollen, d.h. ob es immer zwei Angaben (von der GND und von Wikidata) zu den Merkmalen, also bspw. zwei Geburtsdaten, in der Datenbank geben sollte, oder ob ein Referenzsystem bevorzugt gewählt wird, wenn es zu Differenzen kommen sollte.

AP 1.2 Abschnitt 3 Erweitertes Datenmodell

Letzte Änderung: 22.09.2021

Dokumentation mit alter Version des Datenmodells: Dok. *„Datenmodell_Volltexte“*

Beschreibung von Clemens auf GitHub, die Dateien stellen immer jeweils eine Zeitungsseite dar: https://github.com/sonar-idh/nerdl

- Beschreibung Datenmodells auf GitHub: https://github.com/sonar-idh/Transformer/blob/main/doc/Datamodel.md
- Erweitertet Code, der für die Integration der Volltextentitäten verwendet wurde: https://github.com/sonar-idh/Transformer/tree/main/enrich

Vorschlag von Clemens zur Integration der Volltexte (Stand Februar 2021): Dok. *„Integration der Volltexte mit NER und NEL in GraphML“*

- Nicht umgesetzt bisher: Relation Extraction nach der „sophisticated“ Variante:

  Eine „sophisticated" Variante wäre hingegen die im TSV enthaltenen Texte mit NER/NEL Markup auch noch mit einer Relation Extraction zu verarbeiten, um so ggf. noch „echte" Kanten aus den Sätzen zu extrahieren wie bspw. „Person A erwähnt Person B". Hierfür haben wir aber an der SBB aktuell keinerlei Verfahren, man könnte ggf. nur mit off-the-shelf Tools wie spacy oder dergl. experimentieren - aber vielleicht habt ihr hier am DFKI bereits geeignete Verfahren um das einmal experimentell zu untersuchen?

Neues Datenmodell, veranlasst durch Felix' Kommentar (nur mit alten Daten von Clemens)

Neue Entitäten

- OCRDocument

Neue Relationen

- DocContainsEnt: OCRDocument → PerName | CorpName | GeoName
- SameAs:

  PerName (Source: GND) ↔ PerName (Source: Wikidata)

  CorpName (Source: GND) ↔ CorpName (Source: Wikidata)

  GeoName (Source: GND) ↔ GeoName (Source: Wikidata)

OCRDocument

- id: “OCR” + *Zdb Id* + *Date of Issue. Beispiel: OCR1161410918821002*
- labels: “OCRDocument”
- IdZDB: Entspricht der Zdb Id. *Beispiel: 11614109*
- Name: Entspricht dem Dateinamen (ohne “.tsv”)
- DateStrictBegin: *Beispiel: 12.01.1975*

- DateApproxBegin: *Beispiel: 1975*
- issue: (0 = morning issue, 1 = evening issue etc., default 0)

- page: page/image number
- article: article id (not used, default 0)
- version: not used, default 0

- url: URL. *Beispiel: https://content.staatsbibliothek-berlin.de/zefys/SNP11614109-18821002-0-1-0-0/full/full/0/default.jpg*

Example Nodes

<node id="OCR1161410918821002" labels=":OCRDocument"><data key="labels">:OCRDocument</data><data key="Name">11614109_1882-10-02_1_52_001</data><data key="IdZDB">11614109</data><data key="DateStrictBegin">02.10.1882</data><data key="DateApproxBegin">1882</data><data key="issue">0</data><data key="page">1</data><data key="article">0</data><data key="version">0</data><data key="url">http://content.staatsbibliothek-berlin.de/zefys/SNP11614109-18821002-0-1-0-0/full/full/0/default.jpg</data></node>

Wiki Entititäten

Alle gefundenen Entitäten werden, wie gewohnt, als Entität PerName | CorpName | GeoName hinzugefügt. Jede der Wiki Entitäten besitzt allerdings das Attribut “Source” mit dem Wert “Wikidata”. Zudem gibt es für jede Information, welche aus Wikidata stammt, ein Extra-Attribut mit dem Präfix “Wd”.

PerName , CorpName, GeoName (generische Attribute)

- id: “Wiki_” + Wikidata Id. *Beispiel: Wiki_Q20775499*
- IdWikidata: *Beispiel: Q20775499*
- IdGND: GND Id (falls Verlinkung bei Wikidata vorhanden)
- Name: Name der Entität

- Source: “Wikidata”

PerName: spezifische Merkmale[1]

- WdDateApproxBegin: *Beispiel: 1893*
- WdDateStrictBegin: *Beispiel: 01.01.1893*
- WdDateApproxOriginal: *Beispiel: 1893-1976*

- WdDateStrictOriginal: *Beispiel: 01.01.1893-01.01.1976*
- WdDateApproxEnd: *Beispiel: 1976*
- WdDateStrictEnd: *Beispiel: 01.01.1976*
- WdGender: Wert entweder 0 (weiblich) oder 1 (männlich)
- WdPlaceOfBirth: Geburtsort Name

- WdPlaceOfBirthId: Geburtsort Wikidata Id
- WdPlaceOfDeath: Sterbeort Name
- WdPlaceOfDeathId: Sterbeort Wikidata Id

Example Nodes

<node id="Wiki_Q183149" labels=":GeoName"><data key="labels">:GeoName</data><data key="IdWikidata">Q183149</data><data key="Name">Berliu</data><data key="Source">Wikidata</data></node>

<node id="Wiki_Q166971" labels=":GeoName"><data key="labels">:GeoName</data><data key="IdWikidata">Q166971</data><data key="Name">Heiligen See</data><data key="Source">Wikidata</data></node>

<node id="Wiki_Unknown2" labels=":PerName"><data key="labels">:PerName</data><data key="IdWikidata">Unknown</data><data key="Name">KaiÅ¿er Wilhelms U .</data><data key="Source">Wikidata</data></node>

DocContainsEnt

- id: "FromOCR" + *OCRDocument Id* + "ToWiki" + *Entitätstyp* + "_" + *Wikidata Id* + *Nr. der Kante*. *Beispiel: FromOCR11614109188210 02ToWikiCorp_Q694714_1*
- label: "DocContainsEnt"

- source: Id des OCRDocuments, welches die Entität enthält.
- target: Id der Entität PerName | CorpName | GeoName (Source:Wikidata)
- TypeAddInfo: "directed" (Gerichtete Relation)
- Sent: indicates the sentence position (≥1, 0 marks sentence boundaries)
- Name: Entspricht dem Namen der gefundenen Entität

- Emb: contains the embedded entity label (BIO chunking)
- Left: hold the token OCR coordinates as absolute pixel values
- Top: hold the token OCR coordinates as absolute pixel values
- Width: hold the token OCR coordinates as absolute pixel values
- Height: hold the token OCR coordinates as absolute pixel values

SameAs

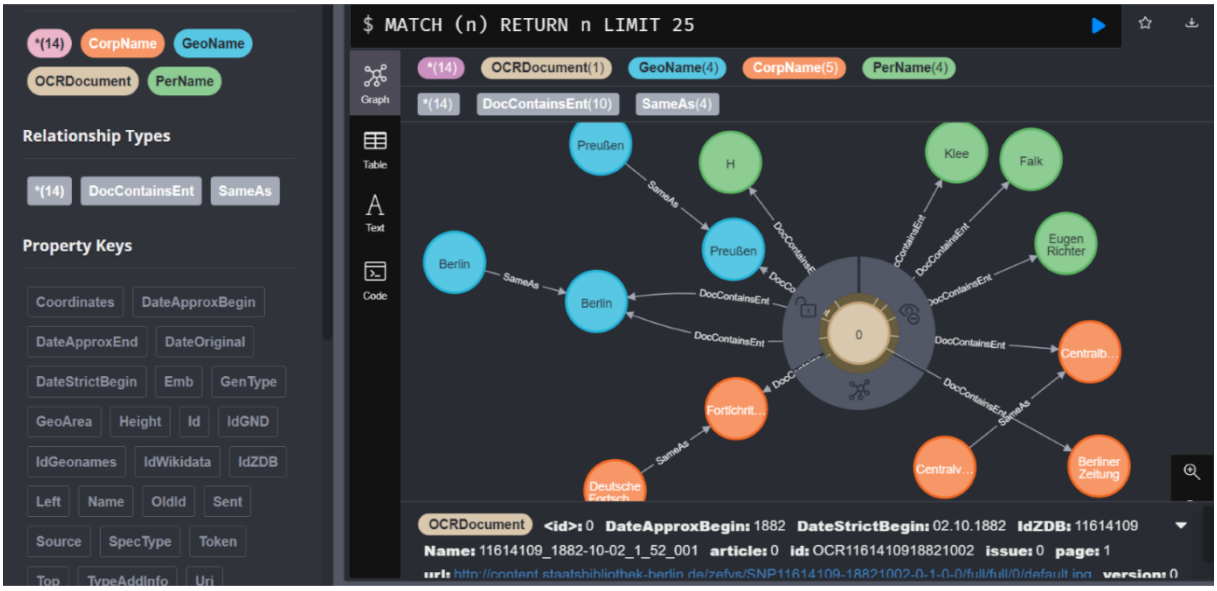
- id: "From" + *GND Id* + "ToWiki" + *Entitätstyp* +"_" + *Wikidata Id*. *Beispiel: From(DE-588)4005728-8ToWikiLOC_Q64*
- label: "SameAs"
- source: Id zum Source Node

- target: Id zum Target Node
- TypeAddInfo: "undirected" (= Die Relation ist ungerichtet)

Example Edges

<edge id="FromAut4047194_9ToWikiLOC_Q38872" source="Aut4047194_9" target="Wiki_Q38872" label="SameAs" directed="false"><data key="label">SameAs</data><data key="TypeAddInfo">undirected</data></edge>

<edge id="FromAut42375_0ToWikiORG_Q1054116" source="Aut42375_0" target="Wiki_Q1054116" label="SameAs" directed="false"><data key="label">SameAs</data><data key="TypeAddInfo">undirected</data></edge>



Stand 27.07.2021 - Call mit Clemens

- Embedded Entities: Embedded Entities besitzen momentan keine Wikidata ID. Dementsprechend sollen die embedded Entities erstmal so in dem Datenmodell bleiben (d.h. als Merkmal “Emb” bei der Relation DocContainsEnt). Wenn die embedded Entities eine Wikidata ID bekommen sollten, dann kann man immer noch darüber nachdenken, sie anders hinzuzufügen
- Bisher standen nur die alten Daten von Clemens zur Verfügung. Die neuen Daten enthalten auch einen Confidence Score. Dieser soll bei der Relation DocContainsEnt als Merkmal integriert werden, sobald die Daten zur Verfügung stehen.
- Die bisher übernommenen URLs sind nicht klickbar. Ein URL Beispiel: http://content.staatsbibliothek-berlin.de/zefys/SNP11614109-18821002-0-1-0-0/left,top,width,height/full/0/default.jpg Um einen validen Link zu erstellen, muss für ein Foto der ganzen Zeitungsseite “full” eingesetzt werden: http://content.staatsbibliothek-berlin.de/zefys/SNP11614109-18821002-0-1-0-0/full/full/0/default.jpg Für ein Foto der exakten Position der Entität müssen die Werte “Left”, “Top”, “Width”, “Height” an den entsprechenden Stellen eingesetzt werden. Hier stellt sich die Frage, ob gewünscht ist, auch ein Foto für die exakte Position der gefundenen Entität mit in das Datenmodell aufzunehmen, oder ob ein Foto der ganzen Zeitungsseite genügt. Falls auch ein Foto der exakten Position gewünscht ist, so kann man dies bei der Entität DocContainsEnt als Merkmal “URL” hinzufügen. Die Merkmale “Left”, “Top”, “Width”, “Height” bleiben vorerst als Platzhalter im Datenmodell.

Frage, welche an die Runde gestellt wurde:

- Ist das Foto der ganzen Zeitungsseite ausreichend, oder ist zusätzlich ein Foto der exakten Position der gefundenen Entität erforderlich? (Siehe oben beschriebenes Problem mit dem URL)

  Antwort (30.08.): Der einzelne Ausschnitt einer Entität ist nicht so nützlich, wenn man den Kontext um die Entität herum nicht lesen kann (abgeschnittene Sätze). Entweder wäre daher ein Ausschnitt der Entität sinnvoll, bei dem man den Kontext gut lesen kann, oder die ganze Zeitungsseite, auf der die gefundenen Entitäten entsprechend farblich markiert werden (vielleicht sogar je nach Entitätstyp mit unterschiedlichen Farben). Um Speicher zu sparen,

wäre vielleicht eine gute Lösung, die ganze Zeitungsseite bei der OCRDocument Entität mit ihren gefundenen, farblich markierten Named Entities darzustellen (und dafür nicht pro gefundene Named Entity ein Bild).

AP 1.3 Datenmanagement

Georg Rehm, Melina Plakidis, Felix Ostrowski (SBB), erstellt: 13.09.2021

Für die Versionierung wird vorgeschlagen, dass Datendumps von externen Datenquellen von SoNAR bereitgestellt und im monatlichen Rhythmus aktualisiert werden. Dies soll das Referenzieren und die Reproduzierbarkeit der Abfrageergebnisse (bspw. das Teilnetzwerk und die Visualisierung) gewährleisten. Dabei werden nicht nur die Datenquellen jeweils mit einer URL versehen, sondern auch die entsprechenden ETL-Konfigurationen, welche für die Datendumps zur Transformation der Daten verwendet wurden. Die Kombination aus der Bereitstellung der verwendeten Datenquellen zu Zeitpunkt *x* sowie der ETL-Konfiguration (z.B. Code auf GitHub) ermöglicht, dass Nutzer*innen die Version der Datenbank jederzeit selbst replizieren können und auch eventuelle Fehler, welche während des ETL-Prozesses unterlaufen sind, transparent einsehen können. Sogenannte „Provenienzdaten“, welche den aktuellen Stand der Datenbank dokumentieren (Informationen zu den Datendumps und der ETL-Prozesse), können beim Download der Daten zur Verfügung gestellt werden. Jede Version von SoNAR könnte eine URL mit diesen Daten enthalten, die wiederum als Referenz für die Abfrageergebnisse angegeben werden könnte. So müssten Nutzer*innen von SoNAR nur eine URL angeben, um die Reproduzierbarkeit ihrer Ergebnisse zu gewährleisten.

Falls Neo4j als Graphdatenbank weiterhin für SoNAR verwendet wird, könnte man ergänzend über die Implementation des Neo4j Plugins „Neo4j-Versioner-Core“ [1]nachdenken. Das Plugin ermöglicht die komplette Versionierung des Graphen mithilfe eines Entity-State Modells.

AP 1.4 Datenspeicherung

- Anleitung zur Neo4j Installation im Anhang (C)
- Cypher Guidelines im Anhang (D)

F1: Warum wurde Neo4j als Graphdatenbank ausgewählt?

Neo4j besitzt folgende Vorteile:

• Umfassende Möglichkeiten bzgl. Modellierbarkeit des Datenmodells. Neo4j ermöglicht, Entitäten und Relationen sowie ihre Merkmale direkt als Knoten und Kanten darzustellen. Bei Linked-Data-Ansätzen (getestet wurde das Produkt GraphDB) lassen sich keine Merkmale zu Kanten speichern. In LD existieren zwar theoretische Arbeiten, die dieses Problem auf unterschiedliche Weisen lösen, allerdings werden diese noch nicht von Datenbanken umgesetzt.

• Flexibles Datenmodell, das jederzeit geändert werden kann (dies ist bei allen GDB der Fall).

• Native Möglichkeiten der Datenvisualisierung, was während des Projekts eine immense Hilfe bei der Arbeit mit den Daten war.

• Performanz hinsichtlich komplexer Abfragen ist gut. Diese hängt von der Datenmenge ab – einige Abfragen, insb. z.B. Sortierung, feinkörnige Filterung o.Ä., haben eine sehr lange

Laufzeit. LD bietet keine nachweislich bessere Performanz. Einige Abfragen können Sekunden dauern, komplexere Abfragen hingegen Minuten oder Stunden.

• Die Performanz beim Import ist sehr gut, wobei GraphML die schnellsten Ergebnisse liefert. Der Import benötigt ca. drei Stunden für 40-60 GB Daten. Der Import via Cypher dauert mehrere Tage.

• Die Performanz beim Export ist sehr gut.

• Nachnutzbarkeit. Grundsätzlich existiert die Möglichkeit, die entwickelte Technologie in Folgeprojekten nachzunutzen und weiterzuentwickeln.

• Die Evaluation der beiden Abfragesprachen hat ergeben, dass Cypher praktikabler ist. Abfragen sind weniger komplex, die Syntax ist verständlicher.

Ferner sind mit Neo4j auch verschiedene Nachteile verbunden:

• Verknüpfung mit weiteren externen Daten ist nur bedingt möglich. Linked Data (LD) bietet in dieser Hinsicht weitaus bessere Möglichkeiten. Erschwerend kommt hinzu, dass zahlreiche bibliothekarische Projekte bereits jetzt Linked Data-Schnittstellen zur Verfügung stellen, die prinzipiell in Frage kämen, die SoNAR-Datenbasis zu ergänzen.

• Die Kosten der Neo4j Enterprise Version sind signifikant.

F2: Welche Alternativen existieren?

Es existiert eine Vielzahl von GDB bzw. Triple-Stores (für LD). Getestet wurde GraphDB. Neo4j bietet diverse technische Vorteile (interne Graphalgorithmen, Schnittstellen für zahlreiche Programmiersprachen, einfachere Abfragesprache, diverse Import-/Exportformate), siehe Frage F1.

F3: Welche Probleme existieren?

• Die meisten Probleme bereitete der Import in das GraphML-Format. Jeder Knoten soll eine eindeutige ID haben, aber unsere Ausgangsdaten enthalten veraltete IDs (bzgl. GND), nicht-individualisierte Personen oder Tn-Datensätze (GND) sowie ungültige IDs bei Ressourcen (DNB, ZDB, SBB). Es musste somit jede ID geprüft werden. Mit jedem neuen Datendump muss eine erneute Prüfung erfolgen. Eine Lösung wäre es, Knoten und Kanten dynamisch mittels entsprechender Cypher-Abfragen hinzuzufügen.

• In GraphML nicht erlaubte Zeichen haben während des Imports Fehlermeldungen verursacht, siehe z.B. die Normalisierungsfunktion für str.[10]

AP 1.5 Anwendungsschnittstellen (API)

F1: Warum wurden APIs zur Datenabfrage zur Verfügung gestellt?

[10] https://github.com/sonar-idh/Transformer/blob/main/src/MarcTransform.py

Wir ermöglichen Abfragen zur Filterung und Suche nach Entitäten und Relationen basierend auf einem Identifikator, Label oder beiden Angaben.[11] Für Visualisierungen wird für eine Entität ein Personennetzwerk abgefragt (realisiert in Abstimmung mit der FH Potsdam). Der Export der Daten ist ebenfalls möglich. Der primäre Zweck des APIs ist es, ein einfaches Werkzeug für das Querying zur Verfügung zu stellen. Erweiterungen können vorgenommen werden. Neben den eigentlichen Abfragen kann auch eine Statistik zum aktuellen Datenbestand erzeugt werden, die Angaben zu Entitäten und Relationen enthält.

F2: Welche Alternativen zu formulierten Querys existieren?

Derartige API-basierte Abfragen können beliebig komplex formuliert werden. Es ist eine technische Aufgabe, diese Abfragen zu implementieren. Wenn die Menge der präformulierten Query-Templates (ggf. informiert durch NutzerInnen oder durch Evaluationsergebnisse) erweitert werden soll, kann dies in der API umgesetzt werden.

[11] https://github.com/sonar-idh/api/blob/main/SoNAR%20API%20Demo%20Swagger%20UI.pdf

Anhang

Anhang (A) - Wikidata & GND Vergleich der Werte (Organisationen)

Auswahl: 10 Organisationen

Rot markiert: Problematisch aufgrund von Differenzen

Beobachtung und Fazit:

Name der Organisation in Wikidata (Label) und in der GND scheinen in der kleinen Stichprobe immer übereinzustimmen. Das Problem für das automatisierte Anreichern von Informationen durch Wikidata stellt meines Erachtens vor allem die Struktur der GND dar. In der kleinen Stichprobe von 10 Organisationen sind schon einige Informationen in Feldern, die dafür nicht vorgesehen sind bzw. nicht einheitlich dargestellt. Beispielsweise Erläuterungen: Definition: Sitz: Hamburg und Erläuterungen: Definition: Sitz des NATO-Hauptquartiers ist Brüssel oder Weitere Angaben: African Studies Association (ASA), Univ. of Calif., Los Angeles; founded 1957 . Das macht das Abgleichen, ob Werte schon in der Datenbank sind, so gut wie unmöglich.

Außerdem gibt es einige sehr abweichende Werte, wie z.B. bei den Entstehungsdaten. Aufgrund vorheriger Arbeit vermute ich, dass dies teilweise durch Namensänderungen der jeweiligen Organisationen zu begründen ist, wodurch es hierbei scheinbar wesentlich häufiger zu Unstimmigkeiten kommt als bei dem Entitätstyp *Person.*

Falls das automatisierte Anreichern trotzdem erwünscht ist, sollte zu bedenken sein, dass die Anzahl verlinkter Organisationen nur sehr gering ist (bei einer Stichprobe von 1000 Organisation unserer Datenbank waren nur rund 12% der Entitäten mit der Wikidata-Id verlinkt). Auch wenn man sich hier nur auf eine bestimmte Auswahl an Merkmalen für das Anreichern begrenzen würde, schätze ich den Aufwand, damit es nicht zu Dopplungen kommt, als relativ hoch ein. Es stellt sich die Frage, ob es sinnvoll ist, für einen Zuwachs an Informationen geringere Übersichtlichkeit und Effizienz der Datenbank in Kauf zu nehmen.

GND-ID: 275-6	***Wikidata-ID: Q49115***
Organisation: Cornell University	Label: Cornell University Official name: Cornell University (English) Native label: Cornell University (English)
Andere Namen: Kornel'skii Universitet Cornell Univ. University (Ithaca, NY)	Label in other languages + Also known as
Zeit: 1865-	Inception: 1865
Land: USA (XD-US)	Country: United States of America

Geografischer Bezug: Ort: Ithaca, NY Wirkungsraum: USA	Located in the administrative territorial entity: Ithaca Headquarters location: Ithaca
Oberbegriffe: Beispiel für: Universität	Instance of: private university, land-grant university, sun grant institution, research university, private not-for-profit educational institution
Typ: Organisation (kiz)	Instance of: organisation not in instance of

277-X	***Q464880***
Organisation: American Heart Association	Label: American Heart Association
Andere Namen: Heart Association (USA) Amerikanische Herzgesellschaft AHA (Abkürzung) A.H.A. (American Heart Association) AHA (American Heart Association)	Label in other languages + Also known as
Zeit: 1922-	Inception: 26 February 1924 (Gregorian)
Land: USA (XD-US)	Country: United States of America
Geografischer Bezug: Wirkungsraum: USA	Country: United States of America
Typ: Organisation (kiz)	Instance of: nonprofit organization

298-7	***Q1202821***
Organisation: Deutsche Gesellschaft für Biophysik	Label: Deutsche Gesellschaft für Biophysik

Andere Namen: Gesellschaft für Biophysik (Deutschland, Bundesrepublik) German Biophysical Society Biophysical Society (Deutschland, Bundesrepublik) Biophysical Society (Deutschland) Gesellschaft für Biophysik (Deutschland)	Label in other languages + Also known as
Zeit: 1961-	-
Land: Deutschland (XA-DE)	Country: Germany
Geografischer Bezug: Wirkungsraum: Deutschland (Bundesrepublik) Wirkungsraum: Deutschland	Country: Germany
Typ: Organisation (kiz)	Instance of: pressure group

299-9	***Q1202828***
Organisation: Deutsche Gesellschaft für Chirurgie	Label: Deutsche Gesellschaft für Chirurgie Official name: Deutsche Gesellschaft für Chirurgie (German) Short name: DGCh (German)
Andere Namen: German Society of Surgery DGCH (Abkürzung) German Society for Surgery	Label in other languages + Also known as
Zeit: 1872-	Inception: 1872
Land: Deutsches Reich (XA-DXDE); Deutschland (XA-DE); Berlin (XA-DE-BE)	Country: Germany
Geografischer Bezug: Ort: Berlin Wirkungsraum: Deutschland	located in the administrative territorial entity: Berlin Country: Germany
Oberbegriffe: Beispiel für: Medizinische Gesellschaft	
Thematischer Bezug: Chirurgie	
Typ: Organisation (kiz)	Instance of: association

305-0	***Q17353492***
Organisation: Deutsche Gesellschaft für Pathologie	Label: Deutsche Gesellschaft für Pathologie

Andere Namen: Gesellschaft für Pathologie (Deutschland, Bundesrepublik) German Society of Pathology Society of Pathology (Deutschland, Bundesrepublik) Gesellschaft für Pathologie (Deutschland) Society of Pathology (Deutschland)	Label in other languages + Also known as
Erläuterungen: Definition: Sitz: Hamburg	Headquarters location: Berlin
Zeit: 1947-	Inception: 1897
Land: Deutschland (XA-DE)	Country: Germany
Vorgänger: Deutsche Pathologische Gesellschaft	-
Geografischer Bezug: Ort: Berlin Wirkungsraum: Deutschland	located in the administrative territorial entity: Berlin Country: Germany
Typ: Organisation (kiz)	Instance of: association, learned society

309-8	***Q819187***
Organisation: Deutsche Physikalische Gesellschaft	Label: Deutsche Physikalische Gesellschaft Official name: Physikalische Gesellschaft zu Berlin (German) start time 14 January 1845 end time 31 December 1898 Deutsche Physikalische Gesellschaft (German) start time 1 January 1899
Andere Namen: German Physical Society DPG (Abkürzung)	Label in other languages + Also known as

Deutsche Physikalische Gesellschaft e.V.	
Zeit: 1899-1945 1963-	Inception: 14 January 1845Gregorian
Land: Deutsches Reich (XA-DXDE); Nordrhein-Westfalen (XA-DE-NW)	Country: German Reich; Germany
Vorgänger: Physikalische Gesellschaft (Berlin)	-
Zeitweiser Name: Verband Deutscher Physikalischer Gesellschaften	-
Geografischer Bezug: Ort: Bad Honnef Wirkungsraum: Deutschland	Headquarters location: Bad Honnef Country: German Reich; Germany
Typ: Organisation (kiz)	Instance of: association, professional society

332-3	***Q1282089***
Organisation: Econometric Society	Label: Econometric Society
Andere Namen: Sociedad Econométrica ES (Abkürzung) Colorado Springs, The Econometric Society Société d'économétrie Société internationale d’économétrie International Society for the Advancement of Economic Theory and Its Relation to Statistics and Mathematics	Label in other languages + Also known as
Zeit: 1930-	Inception: 29 December 1930
Land: Internationale Staatengemeinschaften, internationale Organisationen, internationale Körperschaften (XP)	Country: United States of America
Typ: Organisation (kiz)	Instance of: scientific society, learned society, open-access publisher

333-5	***Q1065***
Organisation: Vereinte Nationen	Label: Vereinte Nationen Official name: official names in multiple languages Native label: Native label in multiple languages

	Short name: short name in multiple languages
Andere Namen: Nations Unies (Französisch, Code: fre) […….] United Nations Organization UNO (Abkürzung) United Nations. Headquarters. Hauptabteilung Abrüstung (Spitzenorgan)	Label in other languages + Also known as
Zeit: 26.06.1945-	Inception: 1945
Land: Internationale Staatengemeinschaften, internationale Organisationen, internationale Körperschaften (XP); USA (XD-US)	Legal form: intergovernmental organization
Geografischer Bezug: Ort: New York, NY	Headquarters location: New York City
Oberbegriffe: Beispiel für: Internationale staatliche Organisation	Instance of: intergovernmental organization, international organization
Typ: Organisation (kiz)	Instance of: intergovernmental organization, international organization

377-3	***Q7184***
Organisation: NATO	Label: NATO Official name: official names in multiple languages Native label: North Atlantic Treaty Organization (English) Short name: short name in multiple languages
Andere Namen: OTAN (Abkürzung) (Französisch, Code: fre) [……..] OTAN (Französisch, Code: fre)	Label in other languages + Also known as
Erläuterungen: Definition: Sitz des NATO-Hauptquartiers ist Brüssel. Verwendungshinweis: In der Sacherschließung auch benutzt für das Gebiet der Körperschaft sowie für die Mitgliedsstaaten als Staatengruppe, soweit im Dokument als geographische Angabe behandelt. Das SW Mitgliedsstaaten tritt nur	-

hinzu, wenn im Dokument über den geographischen Bezug hinaus thematisiert, insbesondere wenn das Verhältnis Körperschaft / Mitgliedsstaaten thematisiert ist.	
Zeit: 1949-	Inception: 4 April 1949
Land: Internationale Staatengemeinschaften, internationale Organisationen, internationale Körperschaften (XP)	-
Oberbegriffe: Beispiel für: Internationale Organisation Beispiel für: Bündnis	Instance of: military alliance, intergovernmental organization, international organization, multinational military coalition
Typ: Organisation (kiz)	Instance of: military alliance, intergovernmental organization, international organization, multinational military coalition

490-X	***Q4689923***
Organisation: African Studies Association	Label: African Studies Association Official name: official names in multiple languages Native label: North Atlantic Treaty Organization (English) Short name: short name in multiple languages
Andere Namen: ASA (African Studies Association) (Abkürzung) A.S.A. (African Studies Association) (Abkürzung) African Studies Association of America	Label in other languages + Also known as
Land: USA (XD-US)	-
Weitere Angaben: African Studies Association (ASA), Univ. of Calif., Los Angeles; founded 1957	Inception: 1957
Typ: Organisation (kiz)	Instance of: organization, learned society

Anhang (B) - Wikidata & GND Vergleich der Werte (Personen)

Auswahl: 20 Personen

Rot markiert: Problematisch aufgrund von Differenzen

Beobachtung und Fazit: Es ist nicht möglich, den Namen automatisch immer fehlerfrei in das Format der GND *Nachname, Vorname* zu bringen. Bei Wikidata sind nicht immer die Merkmale *given name, family name* angegeben, sodass es nicht ersichtlich ist, welcher der Vor- und der Nachname ist. Selbst wenn die Merkmale angegeben sind, kommt es vor, dass bei mehreren Vornamen nur ein Name genannt ist, oder gar mehr als bei dem Label angegeben wurde. Auch wenn bei den Vornamen alle im Merkmal *given name* genannt wurden, ist nicht immer auch die Reihenfolge der Vornamen angegeben. Bei der Kategorie *Andere Namen* ist es genauso unmöglich, die Namen in dasselbe Format zu bringen, da bei Wikidata die Spalte *Also known as* teilweise komplett chaotisch ist, d.h. Namen in den unterschiedlichsten Formaten beinhaltet.

Schon bei der kleinen Stichprobe traten außerdem Abweichungen der einzelnen Werte, bspw. bei dem Geburtsdatum oder dem Geburtsort auf. Außerdem sind bei Wikidata teilweise mehrere Geburts-/ Sterbedaten angegeben. Hier stellt sich die Frage, ob die Werte entweder immer (gekennzeichnet) übernommen werden sollen, d.h. ob es immer zwei Angaben (von der GND und von Wikidata) zu den Merkmalen, also bspw. zwei Geburtsdaten, in der Datenbank geben sollte, oder ob ein Referenzsystem bevorzugt gewählt wird, wenn es zu Differenzen kommen sollte.

GND-ID: 100000193	***Wikidata-ID: Q55861550***
Person: Bauer, Johann Gottfried	Label: Johann Gottfried Bauer Given name: Johann (1); Gottfried (2) Family name: Bauer
Akademischer Grad: Prof.	-
Geschlecht: männlich	Sex or gender: male
Zeit: Lebensdaten: 1695-1763	Date of birth: 20 February 1695 Date of death: 2 March 1763 1763
Land: Deutschland (XA-DE)	-
Geografischer Bezug: Geburtsort: Leipzig Sterbeort: Leipzig Wirkungsort: Leipzig Wirkungsort: Merseburg	Place of birth: Leipzig Place of death: Leipzig Employer: Leipzig University
Beruf(e): Jurist, Hochschullehrer	Occupation: jurist Employer: Leipzig University
Weitere Angaben: Ordinarius an der jurist. Fakultät der Univ. Leipzig	-
Beziehungen zu Personen: Bauer, Gottfried (Vater)	Child: Heinrich Gottfried Bauer

Bauer, Heinrich Gottfried (Sohn) Bauer, Friedrich Wilhelm (Sohn)	
Typ: Person (piz)	Instance of: human

100000231	***Q324487***
Person: Baur, Johann Wilhelm	Label: Johann Wilhelm Baur Given name: Johann
Geschlecht: männlich	Sex or gender: male
Zeit: Lebensdaten: 1607-1642 (nach aktuellem Forschungsstand; anderslt. Todesjahr: 1640)	Date of birth: 31 May 1607, 13 May 1607 Date of death: January 1640, 1642, 1 January 1642
Land: Frankreich (XA-FR); Deutschland (XA-DE); Österreich (XA-AT)	Country of citizenship: Germany, Austria
Geografischer Bezug: Geburtsort: Straßburg Sterbeort: Wien	Place of birth: Strasbourg Place of death: Vienna
Beruf(e): Maler, Radierer	Occupation: painter, illustrator, engraver, drawer
Weitere Angaben: Dt. Radierer und Miniaturmaler Maler	-
Typ: Person (piz)	Instance of: human

100000541	***Q592278***
Person: Mosca, Giuseppe	Label: Giuseppe Mosca Given name: Giuseppe Family name: Mosca
Geschlecht: männlich	Sex or gender: male
Zeit: Lebensdaten: 1772-1839	Date of birth: 1772 Date of death: 14 September 1839
Land: Italien (XA-IT)	-

Beruf(e): Komponist, Musiker	Occupation: composer, impresario
Beziehungen zu Personen: Mosca, Luigi (Bruder)	-
Typ: Person (piz)	Instance of: human

100000789	***Q94903116***
Person: Vacchiery, Karl Albrecht von	Label: Karl Albrecht Edler von Vacchiery Given Name: Karl
Geschlecht: männlich	Sex or gender: male
Andere Namen: Vacchiery, Carl Albrecht von Vacciery, Carl Albrecht von Vacchieri, Karl Albrecht von Vachiery, Carl Albrecht von Vachiery, Karl Albrecht von Vacchiery, Karl Albrecht Vacchieri, Carl Albrecht von Vacchieri, Carl von Vacchieri, Karl von Vachiery, Karl Albert von Vacciery, Carl von Vacchiery, Karl A. von	Only label in other languages
Zeit: Lebensdaten: 1746-1807	Date of birth: 1746 Date of death: 1807
Land: Deutschland (XA-DE)	-
Beruf(e): Historiker	Occupation: historian
Weitere Angaben: Geheimrat, Schulmann, Historiker	-
Typ: Person (piz)	Instance of: human

100000983	***Q55193058***

Person: Chlingensperg, Christoph von	Label: Christoph Chlingensperg auf Berg Given Name: Christoph
Akademischer Grad: Prof.	-
Geschlecht: männlich	Sex or gender: male
Andere Namen: Chlingensberg, Christoph von Chlingensberg, Christophorus de Chlingensberger, Christoph von Chlingensperger, Christoph von Chlingensperger, Christophorus de Chlingenspergerus, Christophorus de Clingensperger, Christoph von Klingensberger, Christoph Klingensperg, Christoph von Klingensperger, Christoph von Chlingensperg, Christophorus de Chlingensperger, Christophorus	Only label in other languages
Zeit: Lebensdaten: 1651-1720	Date of birth: 1651 Date of death: 1720
Land: Deutschland (XA-DE)	-
Beruf(e): Jurist	-
Weitere Angaben: 1693 in den Reichsadelsstand erhoben; Prof. der Rechte in Ingolstadt	-
Beziehungen zu Personen: Chlingensperg auf Berg, Hermann Anton Maria von (Sohn) Chlingensperg, Martin Gottlieb von (Sohn)	-
Typ: Person (piz)	Instance of: human

100001009	***Q1640529***
Person: Baïf, Lazare de	Label: Lazare de Baïf Given Name: Lazare; Lázaro

	Name in native language: Lazare de Baïf (French) Family name: de Baïf
Geschlecht: männlich	Sex or gender: male
Andere Namen: Baïf, Lazare Baïf, Lasare Baïf, Lasare de De Baïf, Lazare Baif, Lazare de Baifius, Lazarus Bayfius, Lazarus Balf, Lazare	Label in other languages + Also known as
Zeit: Lebensdaten: 1496-1547 (Geburtsjahr ca.)	Date of birth: 1496 Date of death: 1547
Land: Frankreich (XA-FR)	Country of citizenship: France
Geografischer Bezug: Geburtsort: La Flèche Sterbeort: Paris	Place of Birth: La Flèche Place of Death: Paris
Beruf(e): Humanist, Diplomat	Occupation: Linguist, diplomat, poet, translator, philosopher, classical scholar
Weitere Angaben: Franz. Diplomat und Humanist Botschafter, Humanist, Italien, Frankreich	-
Beziehungen zu Personen: Baïf, Jean Antoine de (Sohn)	Child: Jean-Antoine de Baïf
Autor von: LAZARI|| BAYFII VIRI DOCTISSI||MI COMMENTARIVS DE VE||stium generibus & uocabulis, in L.||uestis, ff. de auro & argento,|| seu re uestiaria Baïf, Lazare de. - [S.l.] : [s.n.], 1530	-
Typ: Person (piz)	Instance of: human

100001092	***Q76834***
Person: Ancher, Peder Kofod	Label: Peder Kofod Ancher Given Name: Peder

	Family name: Ancher
Geschlecht: männlich	Sex or gender: male
Andere Namen: Ancher, Petrus Kofod Ancher, Peter Kofod Ancher, Kofod Ancher, Peder K. Ancher, P. Kofod Kofod-Ancher, Peder Anker, Peter Kofod Kofod Ancher, P. Ancher, K. Ancker, Petrus Kofod	Label in other languages + Also known as
Zeit: Lebensdaten: 1710-1788	Date of birth: 1710 Date of death: 1788
Land: Dänemark (XA-DK)	Country of citizenship: Denmark
Geografischer Bezug: Geburtsort: Österlarsker Sterbeort: Kopenhagen Wirkungsort: Kopenhagen	Place of Death: Copenhagen Employer: University of Copenhagen
Beruf(e): Jurist Hochschullehrer	Occupation: judge, university teacher
Weitere Angaben: Konferenzrat	-
Typ: Person (piz)	Instance of: human

100001343	***Q1352041***
Person: Plotho, Erich Christoph von	Label: Erich Christoph von Plotho Given Name: Erich; Christoph Family name: Plotho
Adelstitel: Freiherr Edler	Noble title: Baron
Geschlecht: männlich	Sex or gender: male
Andere Namen: Plotho, Ehrich Christoph von Plotho, E. C. von Plotho, Erich Christoph Edler Herr von Plotho, Erich Christoph Freiherr von Erich Christoph Edler Herr und Freiherr von Plotho	Label in other languages

Zeit: Lebensdaten: 1707-1788	Date of birth: 23 September 1707 Date of death: 27 January 1788
Land: Deutschland (XA-DE)	Country of citizenship: Germany
Geografischer Bezug: Geburtsort: Elbe-Parey-Parey Sterbeort: Ansbach-Bayreuth Wirkungsort: Frankfurt (Oder) (Studienort) Wirkungsort: Magdeburg Wirkungsort: Berlin	Place of Birth: Parey Place of Death: Zedtwitz
Beruf(e): Politiker Jurist Diplomat	Occupation: diplomat
Weitere Angaben: 1739 Oberappellationsgerichtsrat in Berlin; 1742-1748 Regierungspräsident in Magdeburg	-
Oberbegriffe: Beispiel für: Adel	-
Beziehungen zu Personen: Plotho, Ludwig Otto von (Vater) Friedrich II., Preußen, König	-
Typ: Person (piz)	Instance of: human

100001424	***Q2172331***
Person: André, Rudolf	Label: Rudolf André Given Name: Rudolf
Andere Namen: André, Rudolph	Label in other languages + Also known as
Zeit: Lebensdaten: 1792-1825	Date of birth: 16 January 1792; 9 July 1793 Date of death: 12 January 1825
Land: Deutschland (XA-DE)	Country of citizenship: Germany
Geografischer Bezug: Geburtsort: Gotha Sterbeort: Tišnov	Place of Birth: Gotha Place of Death: Tišnov
Beruf(e): Landwirt Schriftsteller	Occupation: farmer
Weitere Angaben: Dt. Landwirt u. Schriftsteller	-

Beziehungen zu Personen: André, Christian Carl (Vater) André, Emil (Bruder)	Father: Christian Carl Andre Sibling: Emil André
Typ: Person (piz)	Instance of: human

100001467	***Q3620412***
Person: Antonius, Andreas	Label: Antonius Andreas Given Name: Antonius; Antonio Family name: Andreas Pseudonym: Antonius Andreae, Doctor Dulcifluus, Doctor Dulcissimus et Fundatissimus
Andere Namen: Antonius, Andreae (Wikipedia) Andreas, Antonius Andree, Antonius Antoine, André Antonius, Andreas de Aragonia Pseudo-Antonius, Andreas Andrea, Antoine Andreae, Antonius Antoni, Andreu Antonius, de Aragonia Andreae, Antonio	Label in other languages + Also known as
Zeit: Lebensdaten: ca. 1280 - ca. 1320	Date of birth: 1280 Date of death: 1320
Land: Spanien (XA-ES)	Country of citizenship: Spain
Geografischer Bezug: Geburtsort: Tauste Wirkungsort: Lérida	Place of Birth: Tauste Employer: University of Lleida
Beruf(e): Theologe	Occupation: philosopher, theologian
Weitere Angaben: Schüler von Johannes Duns Scotus. "Ihm wird der Verdienst zugeschrieben, eine klare und deutliche Erklärung der Lehren des Scotus gegeben zu haben." OFM; Scriptum in artem veterem; Quaestiones super XII libros Metaphysicae; In novam logicam; etc.	Student of: Duns Scotus
Beziehungen zu Organisationen: Franziskaner	Religious order: Franciscans
Typ: Person (piz)	Instance of: human

Autor von: Quaestiones super duodecim libros Metaphysicae Aristotelis Antonius, Andreas. - Vicenza : Nicolaus PetriVicenza (: Hermann Liechtenstein), 12.05.1477 Handschrift (Thüringer Universitäts- und Landesbibliothek Jena), Ms. G. B. q. 79 Schriftdenkmal (wis)	-

100002188	***Q3263374***
Person: Beausobre, Louis de	Label: Louis de Beausobre Given Name: Louis Family name: Beausobre
Geschlecht: männlich	Sex or gender: male
Andere Namen: Ugtvogt (Pseudonym) Ugtvogt, ..., le Docteur (Pseudonym) (4ELRAKm) Ugtvogt, ... (Pseudonym) D..., le Chevalier (Pseudonym) Beaussobre, Ludwig von (Vorlage) Beausobre, Ludovicus de D. Beausobre, Ludwig B., M. de De Beausobre, Louis Beausobre, M. de Beausobre, L. von Beausobre, Ludwig von	Label in other languages + Also known as
Zeit: Lebensdaten: 1730-1783	Date of birth: 19 August 1730 Date of death: 3 December 1783
Land: Deutschland (XA-DE)	Country of citizenship: Germany
Geografischer Bezug: Geburtsort: Berlin Wirkungsort: Berlin Wirkungsort: Frankfurt (Oder)	Place of Birth: Berlin Work location: Berlin; Frankfurt (Oder)
Beruf(e): Philosoph Schriftsteller Volkswirt	Occupation: philosopher, writer

Weitere Angaben: Philosoph und Nationalökonom, stammte aus alter franz. protestant. Familie, geb. in Berlin; Berlin, Frankfurt/Oder (Wirkungsorte)	-
Typ: Person (piz)	Instance of: human

10000220X	***Q98834306***
Person: Giraud, Pierre François Félix Joseph	Label: Pierre-François-Félix-Joseph Giraud
Zeit: Lebensdaten: 1764-1821	Date of birth: 1764; 20 September 1764 Date of death: 1821; 26 February 1821
Typ: Person (piz)	Instance of: human

100002226	***Q3083416***
Person: Villemain d'Abancourt, François-Jean	Label: François-Jean Willemain d'Abancourt Name in native language: François-Jean Willemain d'Abancourt (French) Given name: François-Jean Pseudonym: Léonard Gobemouche
Andere Namen: Abancourt, François-Jean Villemain d' Abancourt, François Willemain d'Abancourt, François Jean Abancourt, François Jean Willemain d' Willemain d'Abancourt, François-Jean Villemain-D'Abancourt, François-Jean D'Abancourt, François-Jean Villemain Gobemouche, Léonard (Pseudonym) V. d'A. Villemain D'Abancourt, François Jean Willemain D'Abancourt, François Jean Abancourt, Franọis Jean Villemain d'	Label in other languages + Also known as

Zeit: Lebensdaten: 1745-1803	Date of birth: 22 July 1745 Date of death: 16 June 1803; 10 June 1803
Land: Frankreich (XA-FR)	Country of citizenship: France
Geografischer Bezug: Geburtsort: Paris Sterbeort: Paris	Place of Birth: Paris Place of Death: Paris
Beruf(e): Schriftsteller Dramatiker	Occupation: poet, writer, playwright, translator, fabulist, bibliophile
Typ: Person (piz)	Instance of: human

100002307	***Q518333***
Person: Abbadie, Jacques	Label: Jacques Abbadie Name in native language: Jacques Abbadie (French) Given Name: Jacques Family Name: Abbadie Pseudonym: ***** docteur en théologie; J.A****
Geschlecht: männlich	Sex or gender: male
Andere Namen: Boher, Pierre Abbadie, ... Abbadie, Jacobi Abbadie, Jaques J. A. A., J. Abbadie, Jacob	Label in other languages + Also known as
Zeit: Lebensdaten: 1654-1727	Date of birth: 1654 Date of death: 25 September 1727 (Julian); 15 September 1727 (Gregorian)
Land: Frankreich (XA-FR)	Country of citizenship: France
Beruf(e): Theologe	Occupation: theologian; writer; pastor
Weitere Angaben: Franz. ref. Theologe, bedeutender Redner, Prediger der franz.	-

Gemeinde in Berlin u.a., apologet. Schriftsteller Theologe	
Typ: Person (piz)	Instance of: human
Autor von: [Panégyrique de Marie Stuart ...<dt.>] Lob-Rede/ Der Unvergleichlichen Maria Stuart/ gewesenen Königin in England/ Schottland/ Franckreich/ und Irrland Abbadie, Jacques. - Halle, Saale : Universitäts- und Landesbibliothek Sachsen-Anhalt, 1695	-

100002773	***Q2263863***
Person: Accoltus, Benedictus	Label: Benedetto Accolti the Elder Given Name: Benedetto
Andere Namen: Accolti, Benedetto ((VD-16)) Acolti, Benedictus de Accoltis, Benedictus de Accolti, Benoît Accoltus Aretinus, Benedictus Accolti Aretino, Benedetto Benedetto, Accolti Benoît, Accolti Benoît, Accolti d'Arezzo Benoît, d'Arezzo Benedictus, Accolti Benedictus, Accolti Aretinus Benedictus, de Acoltis Accolti, Benedetto, il Vecchio	Label in other languages + Also known as

Acoltis, Benedictus de Aretinus, Benedictus Arretinus, Benedictus Benedictus, Aretinus	
Zeit: Lebensdaten: 1415-1464	Date of birth: 1415 Date of death: 26 September 1464
Land: Italien (XA-IT)	-
Weitere Angaben: Dialogus de praestantia virorum sui aevi; De bello a christianis contra barbaros gesto; Ital. Humanist	-
Beziehungen zu Personen: Eppendorff, Heinrich von (VD-16 Mitverf.) Florido, Francesco (VD-16 Mitverf.) Gast, Johannes (VD-16 Mitverf.) Gaurico, Luca (VD-16 Mitverf.)	-
Typ: Person (piz)	Instance of: human

100002811	***Q3388217***
Person: Accolti, Pietro	Label: Pietro di Fabrizio Accolti Given Name: Pietro
Zeit: Lebensdaten: 1578-1642	Date of birth: 1579 Date of death: 1642
Land: Italien (XA-IT)	Place of Birth: Pisa
Beruf(e): Architekt Maler Mathematiker	Occupation: politician; scientist; painter; architect
Weitere Angaben: Italien. Maler, Architekt und Mathematiker	-
Typ: Person (piz)	Instance of: human

100002994	***Q2190405***
Person: Abela, Giovanni Francesco	Label: Giovanni Francesco Abela Given Name: Giovanni

Andere Namen: Abela, Giovanfrancesco (LCAuth) Abela, Ioannes Franciscus Abela, Johannes Franciscus Abela, Joannes Franciscus Abela, Ġan Franġisk	Label in other languages + Also known as
Zeit: Lebensdaten: 1582-1655	Date of birth: 1582 (Gregorian); 1582 (Gregorian) Date of death: 4 May 1655
Land: Italien (XA-IT); Malta (XA-MT)	Country of citizenship: Malta
Geografischer Bezug: Geburtsort: Valletta Sterbeort: Valletta Wirkungsraum: Italien Wirkungsraum: Malta	Place of Birth: Valletta Place of Death: Valletta
Beruf(e): Adel Ritter Mönch Jurist Historiker Diplomat	Occupation: historian; archaelogist
Weitere Angaben: 1625 Vicekanzler des Malteserordens	-
Typ: Person (piz)	Instance of: human

100003184	***Q100387362***
Person: Achatius, von Brandenburg	Label: Achatius von Brandenburg Given Name: Rudolf
Andere Namen: Brandenburg, Achatius von Brandenburg, Achaz von	Label in other languages
Zeit: Lebensdaten: 1516-1578	Date of birth: 1516
Land: Deutschland (XA-DE)	-
Geografischer Bezug: Geburtsort: Berlin	-
Beruf(e): Adel	-

Theologe Konsistorialrat Autor	
Weitere Angaben: Natürlicher Sohn von Kurfürst Joachim II. von Brandenburg; Scholastikus an St. Viktor in Mainz; trat um 1550 zur evangel. Religion über; kurbrandenburgischer Konsistorilarat u. mitverordneter geistlicher Visitator	-
Typ: Person (piz)	Instance of: human

100003532	***Q384460***
Person: Acosta, Christóval	Label: Cristóvão da Costa Given name: Cristóbal Family name: Costa
Andere Namen: Acosta, Christobal Costa, Christophorus à Costa, Christóval à Acosta, Cristóbal de Costa, Christophoro a Costa, Christoforo a La Coste, Christophle de (4ELRAKm) Acosta, Cristóval Costa, Christovam da Acosta, Christoforo Acosta, Christoual LaCoste, Christophle de (RAK alt) Costa, Christophorus a Costa, Cristóvão da Acosta, Christóbal	Label in other languages + Also known as

Acosta Buenaventura, Cristóbal Buenaventura, Cristóbal Acosta	
Zeit: Lebensdaten: 1515-1580 (Geburts- und Todesjahr ca.)	Date of birth: 1515 Date of death: 1594
Geografischer Bezug: Wirkungsort: Burgos	Work location: Burgos
Beruf(e): Arzt Botaniker Naturwissenschaftler	Occupation: botanist; physician
Weitere Angaben: Geb. in Afrika, Vater war Portugiese Arzt, Botaniker	Place of birth: Tangier
Typ: Person (piz)	Instance of: human

100010652	***Q3713732***
Person: Altomare, Donato A. d'	Label: Donato Antonio Altomare Given Name: Donato
Andere Namen: Altomari, Donatus A. ab Altomari, Donatus Antonius Altomari, Antoine-Donat Altomare, Antonio Donato Altomare, Doinato Antoine d' Altomare, Donato Antonio d' Altomari, Donatus Antonius ab Altomarus, Donatus Antonius Altomarus, Donatus Antonius ab Altomare, Donato Antonio Altomari, Donatius Antonius Altimarus, Donatius Antonius	Label in other languages

Zeit: Lebensdaten: 1506-1562 (Geburtsjahr nach versch. Quellen evtl. auch 1502 oder 1520) Lebensdaten: 1502-1562 (ca.1502- ca.1562)	Date of birth: 1520 Date of death: 1566
Land: Italien (XA-IT)	-
Geografischer Bezug: Geburtsort: Neapel	Place of Birth: Naples
Beruf(e): Arzt Pharmakologe Philosoph	Occupation: physician
Weitere Angaben: Italien. Arzt, Pharmakologe und Philosoph Arzt, Philosoph	-
Typ: Person (piz)	Instance of: human

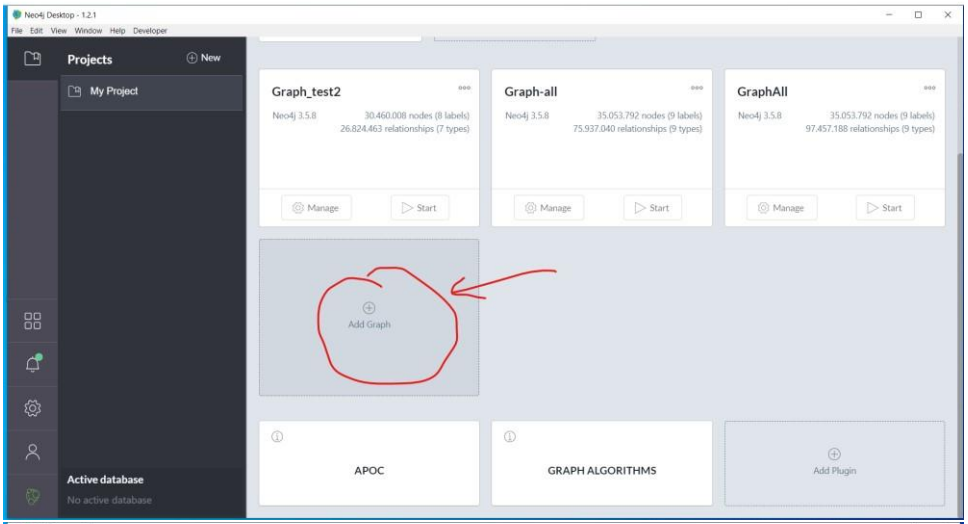
Anhang (C) – Anleitung Neo4j

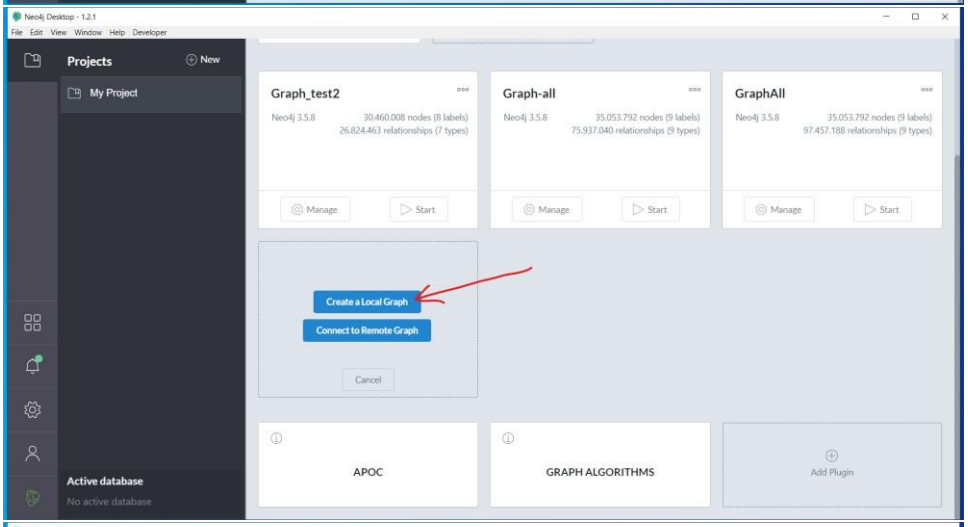
Installation neo4j Desktop

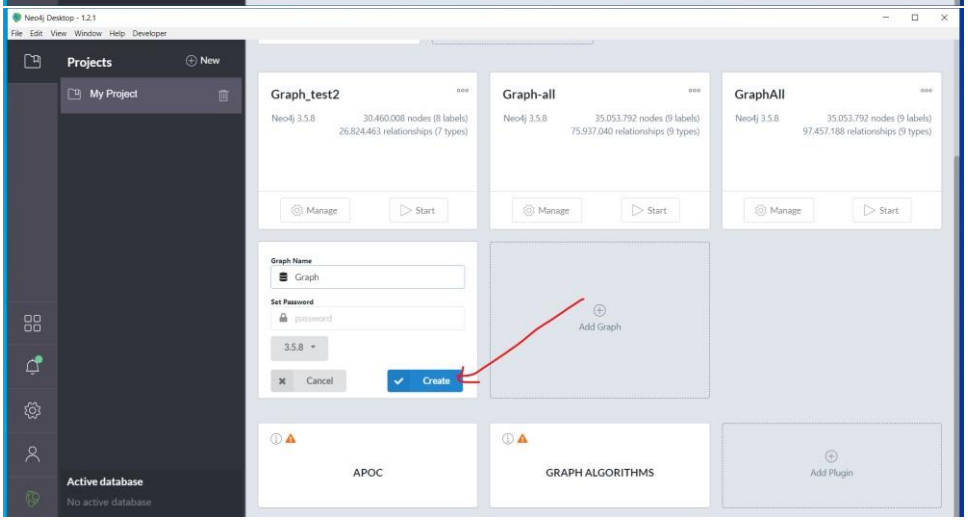
- neo4j runterladen: https://neo4j.com/download-center/#desktop
- Anleitung zur Installation folgen

Erzeugung und Konfiguration eines Graphen

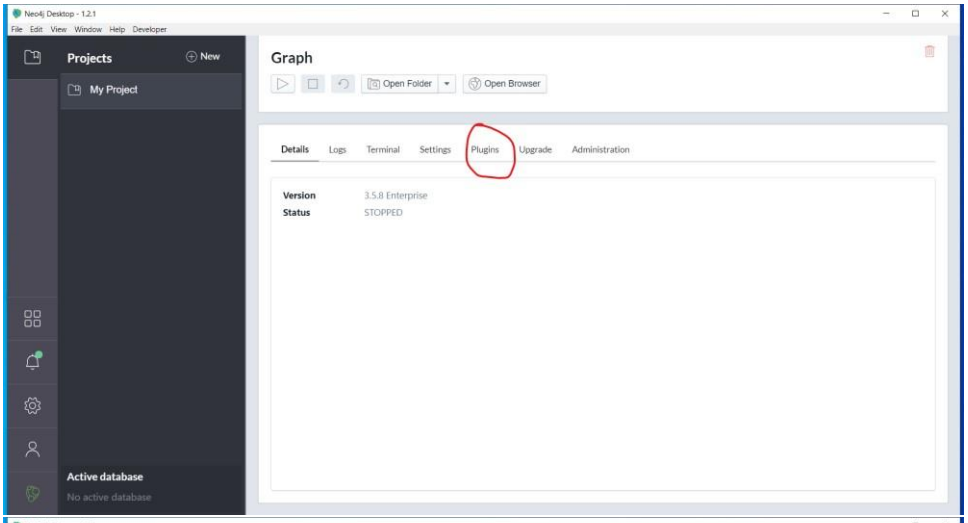
- Neo4j Desktop öffnen
  Add Graph, Create a Local Graph auswählen, Namen und Password hinzufügen

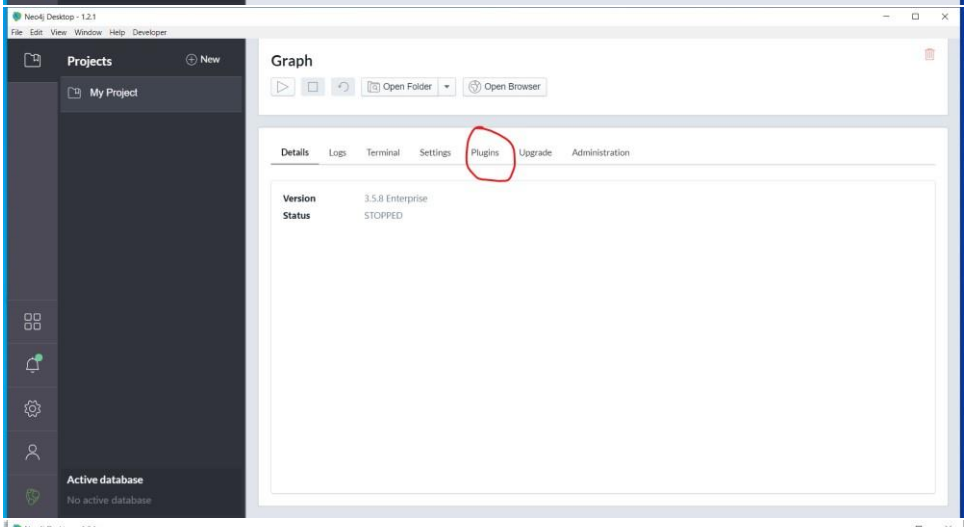


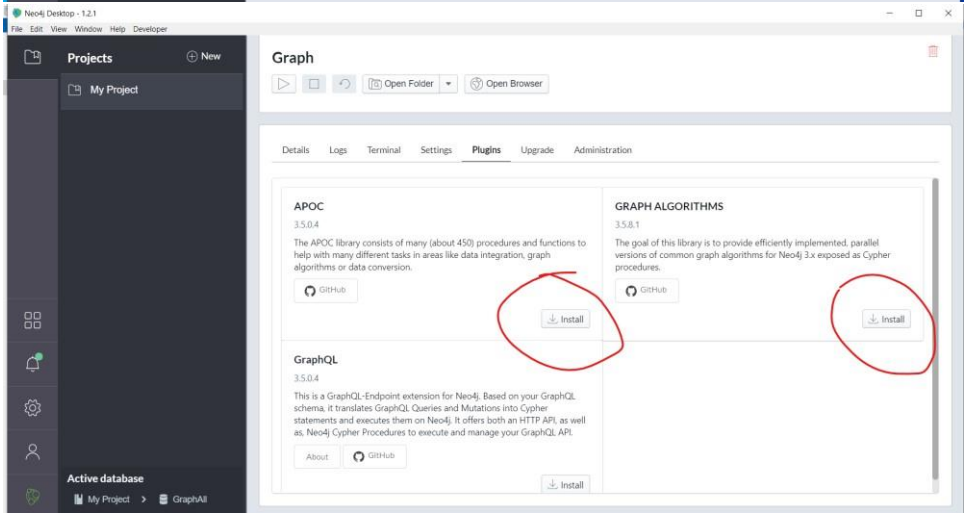


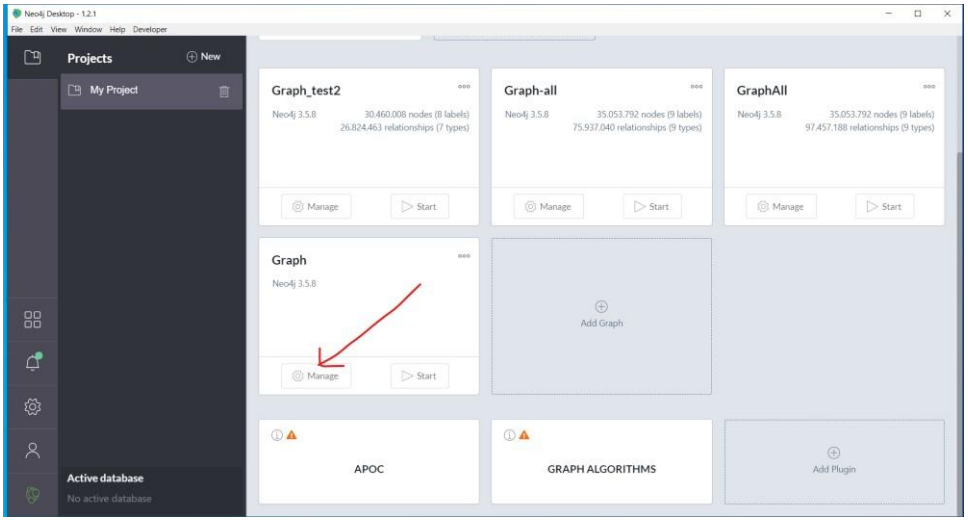


- Zu *Manage*/*Plugings* gehen, **APOC** und **GRAPH ALGORITHMS** installieren

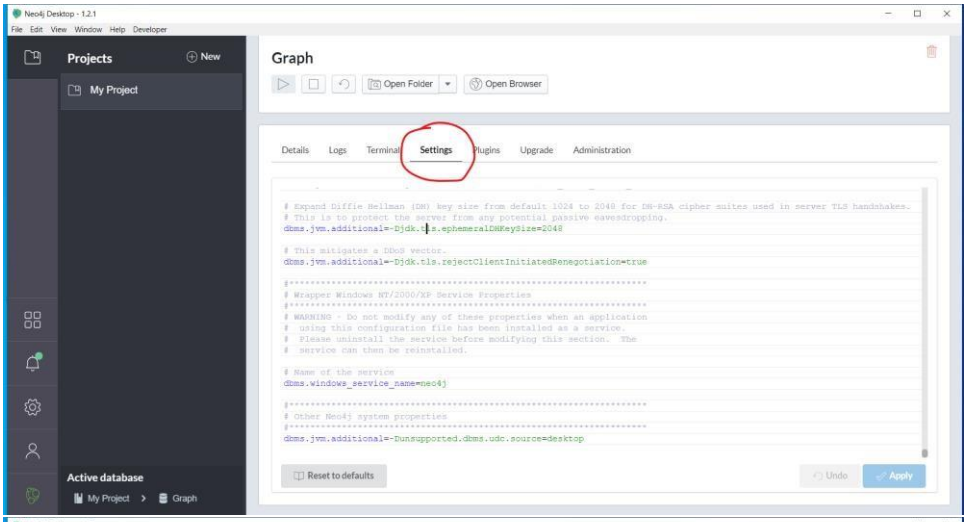

Neo4j Desktop - 1.2.1
File Edit View Window Help Developer
Projects
New
My Project
Graph
Open Folder
Open Browser
Details Logs Terminal Settings Plugins Upgrade Administration
Version 3.5.8 Enterprise
Status STOPPED
Active database
No active database

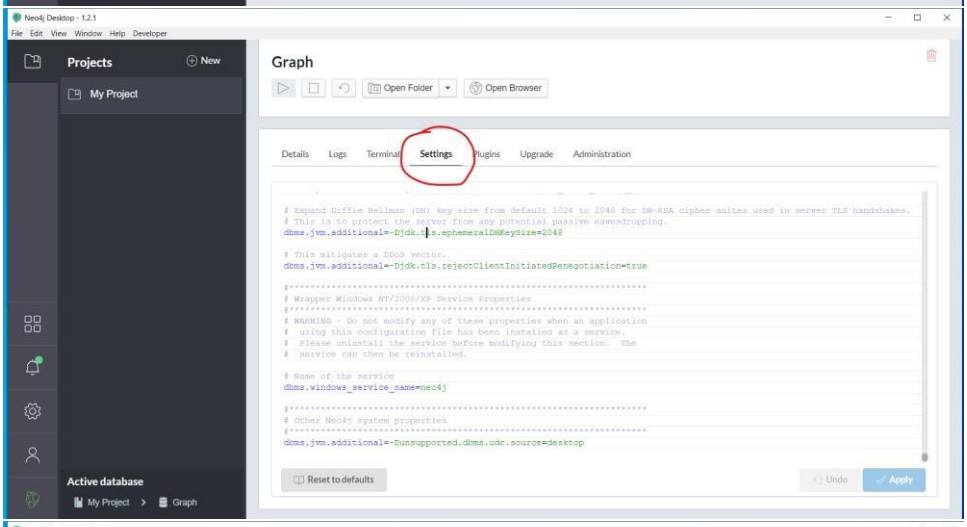

Neo4j Desktop - 1.2.1
File Edit View Window Help Developer
Projects
New
My Project
Graph
Open Folder
Open Browser
Details Logs Terminal Settings Plugins Upgrade Administration
Version 3.5.8 Enterprise
Status STOPPED
Active database
No active database

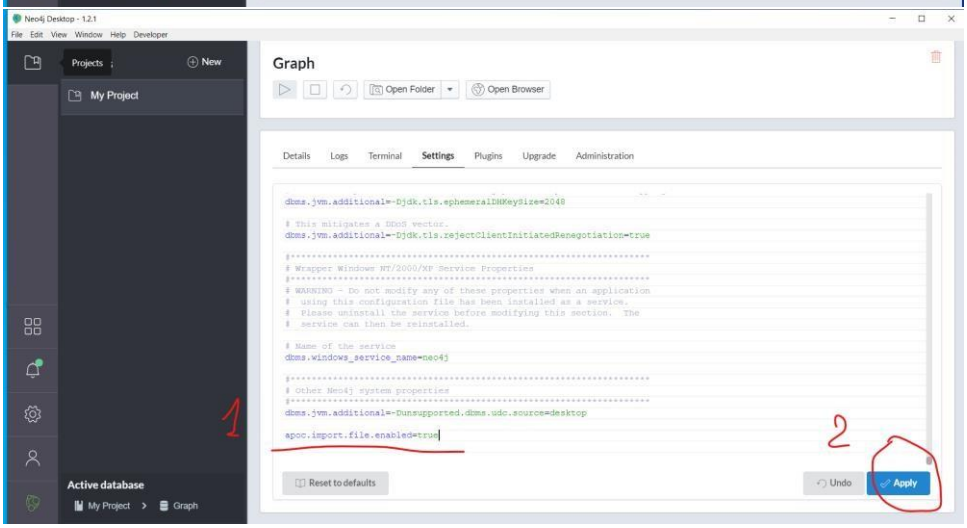

Neo4j Desktop - 1.2.1
File Edit View Window Help Developer
Projects
New
My Project
Graph
Open Folder
Open Browser
Details Logs Terminal Settings Plugins Upgrade Administration
APOC
3.5.0.4
The APOC library consists of many (about 450) procedures and functions to help with many different tasks in areas like data integration, graph algorithms or data conversion.
GitHub
Install
GRAPH ALGORITHMS
3.5.8.1
The goal of this library is to provide efficiently implemented, parallel versions of common graph algorithms for Neo4j 3.x exposed as Cypher procedures.
GitHub
Install
GraphQL
3.5.0.4
This is a GraphQL-Endpoint extension for Neo4j. Based on your GraphQL schema, it translates GraphQL Queries and Mutations into Cypher statements and executes them on Neo4j. It offers both an HTTP API, as well as, Neo4j Cypher Procedures to execute and manage your GraphQL API.
About
GitHub
Install
Active database
My Project
GraphAll


Neo4j Desktop - 1.2.1
File Edit View Window Help Developer
Projects
New
My Project
Graph_test2
Neo4j 3.5.8
30.460.008 nodes (8 labels)
26.824.463 relationships (7 types)
Manage
Start
Graph-all
Neo4j 3.5.8
35.053.792 nodes (9 labels)
75.937.040 relationships (9 types)
Manage
Start
GraphAll
Neo4j 3.5.8
35.053.792 nodes (9 labels)
97.457.188 relationships (9 types)
Manage
Start
Graph
Neo4j 3.5.8
Manage
Start
Add Graph
APOC
GRAPH ALGORITHMS
Add Plugin
Active database
No active database

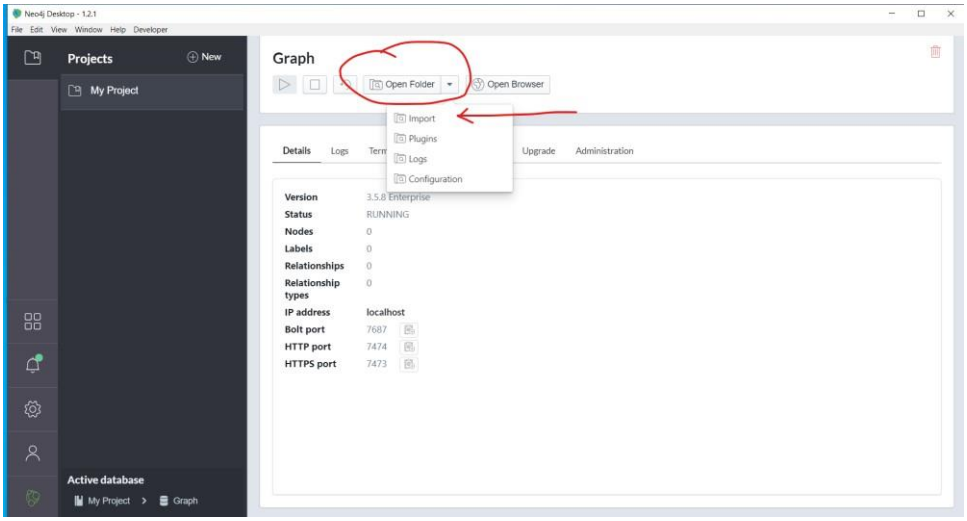
- Zu *Settings* wechseln und **apoc.import.file.enabled=true** hinzufügen. Auf *Apply* drücken.

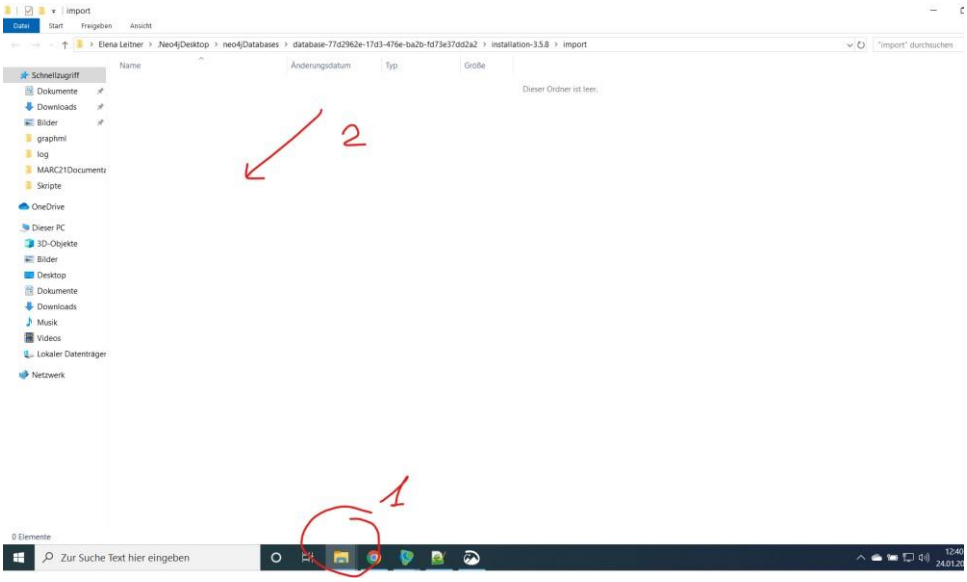




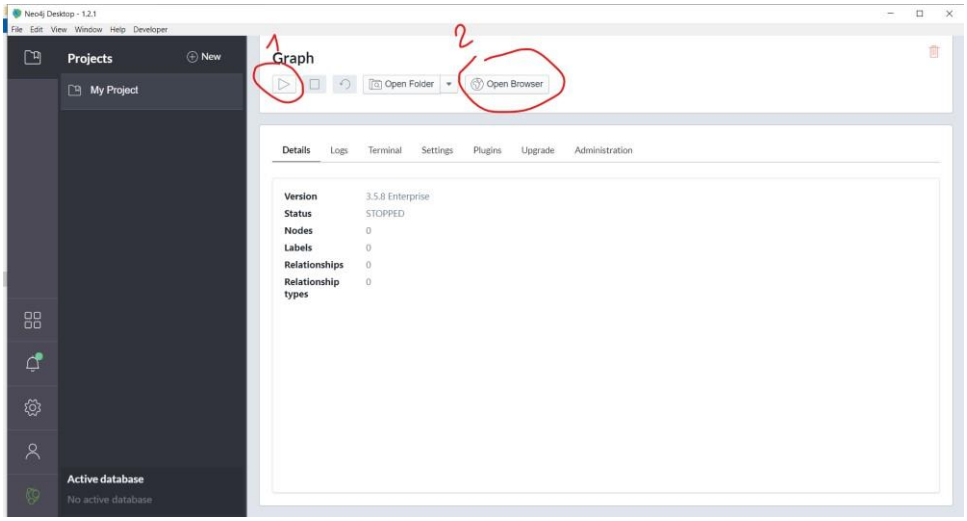


- Zu *Open Folder* gehen und *Import* auswählen. Jetzt öffnet sich ein Ordner. Zu diesem Ordner entpackten Daten in GRAPHML einfügen.
- Die Daten findet man auf **OneDrive**, in *Datenpakete (04)/Datenpakete (Transformation)/GRAPHML/AlleDaten.zip.* Daten runterladen und entpacken!



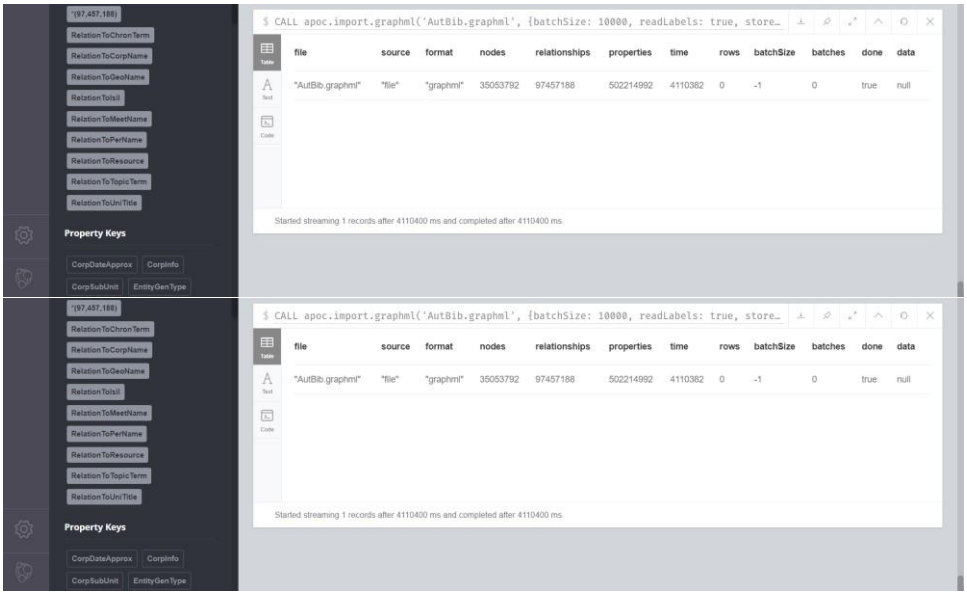


- Graph starten und Browser öffnen



- In Neo4j Browser den Befehl **CALL apoc.import.graphml('AutBib.graphml', {batchSize: 10000, readLabels: true, storeNodeIds: true})** einfügen und auf *Play* drücken. Die Daten werden hochgeladen. Bei mir hat es 69 min gedauert.

- Anschließlich bekommt man eine Meldung, dass Daten erfolgreich hochgeladen wurden.



Mögliche Befehle

- Knoten zählen: *CALL db.labels() YIELD label CALL apoc.cypher.run('MATCH (:`'+label+'`) RETURN count(*) as count',{}) YIELD value RETURN label as Entitaetentyp, value.count as Anzahl*
- Kanten zählen: *CALL db.relationshipTypes() YIELD relationshipType as type CALL apoc.cypher.run('MATCH ()-[:`'+type+'`]->() RETURN count(*) as count',{}) YIELD value RETURN type as Relationentyp, value.count as Anzahl*
- Mögliche Verknüpfungen zählen: *CALL apoc.meta.stats() YIELD relTypes UNWIND keys(relTypes) AS key RETURN key as Relationenart, relTypes[key] as Anzahl ORDER BY relTypes[key]*
- 25 Relationen zu geographischen Namen ausgeben: *MATCH r=()-[]->(:GeoName) RETURN r LIMIT 25*

Weitere Informationen findet ihr z.B. hier: https://neo4j.com/developer/cypher-basics-i/

Anhang (D) – Cypher Guidelines

Import/Export via GRAPHML

! Import/Export in conf einschalten

- apoc.export.file.enabled=true
- apoc.import.file.enabled=true

Import

- Knoten und Kanten aus einer Datei hochladen, ggf. kann man `batchSize` kleiner machen

```
CALL apoc.import.graphml("IMPORT_FILENAME.graphml", {batchSize:
10000, readL abels: true, storeNodeIds: true})
```

Export

- alle Knoten und Kanten exportieren

```
CALL apoc.export.graphml.all("IMPORT_FILENAME.graphml",
{useTypes:true, stor eNodeIds:false})
```

- eine Abfrage exportieren

```
CALL apoc.export.graphml.query("MATCH a=(l)-[*2..]->(k) WHERE
EXISTS(l.Entit yId) AND EXISTS(k.EntityId) RETURN a LIMIT
300","EXPORT_FILENAME.graphml",{u seTypes:true, storeNodeIds:false})
```

Löschen

Falls Import nicht komplett stattgefunden hat, erstmal alle Knoten und Kanten löschen, sonst bekommt man Duplikate in einem Graphen mit unterschiedlichen internen (zugewiesen von neo4j) Identifikatoren.

```
CALL apoc.periodic.iterate("MATCH (n) RETURN n", "DETACH DELETE n",
{batchSize:1
000})
yield batches, total return batches, total
```

Weitere Informationen findet man unter:

- https://neo4j.com/blog/apoc-database-integration-import-export-cypher/ (https://neo4j.com/blog/apocdatabase-integration-import-export-cypher/)
- https://neo4j.com/developer/guide-performance-tuning/ (https://neo4j.com/developer/guide-performancetuning/)

Statistiken zum Graphen

- alle Knoten nach Typ zählen

```
CALL db.labels() YIELD label CALL apoc.cypher.run("MATCH
(:`'+label+'`) RETU RN count(*) as count",{}) YIELD value RETURN
label as Entitaetentyp, value.c ount as Anzahl
```

- alle Kanten nach Typ zählen

```
CALL db.relationshipTypes() YIELD relationshipType as type CALL apoc.cypher.
run("MATCH ()-[:`'+type+'`]->() RETURN count(*) as count",{}) YIELD value RE
TURN type as Relationentyp, value.count as
```

Anzahl

- alle Verknüpfungskombinationen zählen

```
CALL apoc.meta.stats() YIELD relTypes UNWIND keys(relTypes) AS key
RETURN ke y as Relationenart, relTypes[key] as Anzahl ORDER BY
relTypes[key]
```

- Ausgangsknoten zählen

```
MATCH (n) WHERE NOT (n)-->() RETURN COUNT(n)
```

- Zielknoten zählen

```
MATCH (n) WHERE NOT ()-->(n) RETURN COUNT(n)
```

- Knoten ohne Relationen zählen

```
MATCH (n) WHERE (n)--() RETURN COUNT(n)
```

- Knoten, die mit anderen verbunden sind, zählen

```
MATCH (n) WHERE NOT (n)--() RETURN COUNT(n)
```

- Top 15 Relationen zu Personen finden, Namen und Identifikatoren (absteigend) ausgeben

```
MATCH (n)-[]->(o:PerName) RETURN o.EntityName, o.EntityId, count(*)
ORDER by count(*) desc LIMIT 15
```

MATCH-Abfragen

- z.B. Personen, Ortsnamen mit max. Ausgabeanzahl 25 Knoten

```
MATCH (n:PerName) RETURN n LIMIT 25
```

```
MATCH (n:GeoName) RETURN n LIMIT 25
```

- einen Knoten mit

dem internen Identifikator X matchen und einen Knoten ausgeben

```
MATCH (n) WHERE ID(n)=X RETURN n
```
• einen Knoten mit dem internen Identifikator X matchen und EntityName eines Knoten ausgeben

```
MATCH (n) WHERE ID(n)=7389415 RETURN n.EntityName
```
• einen Knoten mit dem internen Identifikator größer als X matchen und id eines Knoten ausgeben

```
MATCH (n) WHERE ID(n)>X RETURN ID(n), n.id
```

• einen Knoten mit CONTAINS matchen, der veraltete Identifikator X enthält

```
MATCH (n) WHERE n.EntityOldId CONTAINS "X" RETURN n
```

Export via CSV

• eine Abfrage exportieren

```
CALL apoc.export.csv.query("MATCH (n)  WHERE n.id CONTAINS 'Aut'
RETURN  n.i  d","EXPORT_FILENAME.csv",{batchSize:  10000})
```
• eine Abfrage exportieren und Delimiter in CSV bestimmen

```
CALL apoc.export.csv.query("MATCH (n)-[r:RelationToCorpName]-
>(o:UniTitle) R
ETURN n.EntityName, n.id, labels(n)[0], TYPE(r),o.EntityName, o.id,
labels
(o)[0] ", "EXPORT_FILENAME.csv", {delim: "\t"})
```

Nachschlagequellen

- https://daten-und-bass.io/blog/graph-data-modelling-with-neo4j-a-short-introduction/ (https://daten-undbass.io/blog/graph-data-modelling-with-neo4j-a-short-introduction/)
- https://neo4j.com/developer/guide-data-modeling/ (https://neo4j.com/developer/guide-data-
- modeling/) https://medium.com/@niazangels/export-and-import-your-neo4j-graph-easily-with-apoc-4ea614f7cbdf (https://medium.com/@niazangels/export-and-import-your-neo4j-
- graph-easily-with-apoc-4ea614f7cbdf) https://www.freecodecamp.org/news/writing-a-command-line-database-client-in-10-minutesaa608536ae4b/ (https://www.freecodecamp.org/news/writing-a-command-line-database-client-in-10minutes-
- aa608536ae4b/) https://kuczera.github.io/Graphentechnologien/25_xml2neo4j-kollatz.html (https://kuczera.github.io/Graphentechnologien/25_xml2neo4j-kollatz.html)
- https://gist.github.com/wjgilmore/8ba5f31ef1435dc04c52 (https://gist github com/wjgilmore/8ba5f31ef1435dc04c52)