

Regestenmodellierung im Graphen

Contents

1	Inhalt	2
2	Regestenmodellierung im Graphen	2
2.1	Wie kommen die Regesten in den Graphen	2
2.1.1	Import mit dem <code>LOAD CSV</code> -Befehl	3
2.1.2	Google-Docs für den CSV-Download	4
2.1.3	Regestenmodellierung im Graphen	4
2.1.4	Indexe Erstellen	7
2.1.5	Erstellen der Regestenknoten	10
2.1.6	Erstellen der Ausstellungsorte	11
2.1.7	Koordinaten der Ausstellungsorte	13
2.1.8	Ausstellungsdatum	13
2.2	Exkurs 1: Herrscherhandeln in den Regesta Imperii	14
2.3	Zitationsnetzwerke in den Regesta Imperii	16
2.4	Import der Registerdaten in die Graphdatenbank	16
2.4.1	Vorbereitung der Registerdaten	16
2.4.2	Import der Registerdaten in die Graphdatenbank	19
2.5	Exkurs 2: Die Hierarchie des Registers der Regesten Kaiser Friedrichs III.	20
2.6	Auswertungsperspektiven	23
2.6.1	Personennetzwerke in den Registern	23
2.6.2	Herrscherhandeln ausgezählt	30
2.6.3	Herrscherhandeln pro Ausstellungsort ausgezählt	31
2.6.4	Herrscherhandeln und Anwesenheit	32
2.6.5	Regesten 200 km rund um Augsburg	33
2.6.6	Welche Literatur wird am meisten zitiert	34

2.6.7	Der Import zusammengefasst	35
2.7	Zusammenfassung	35

1 Inhalt

{:.no_toc}

- Will be replaced with the ToC, excluding the “Contents” header {:.toc}

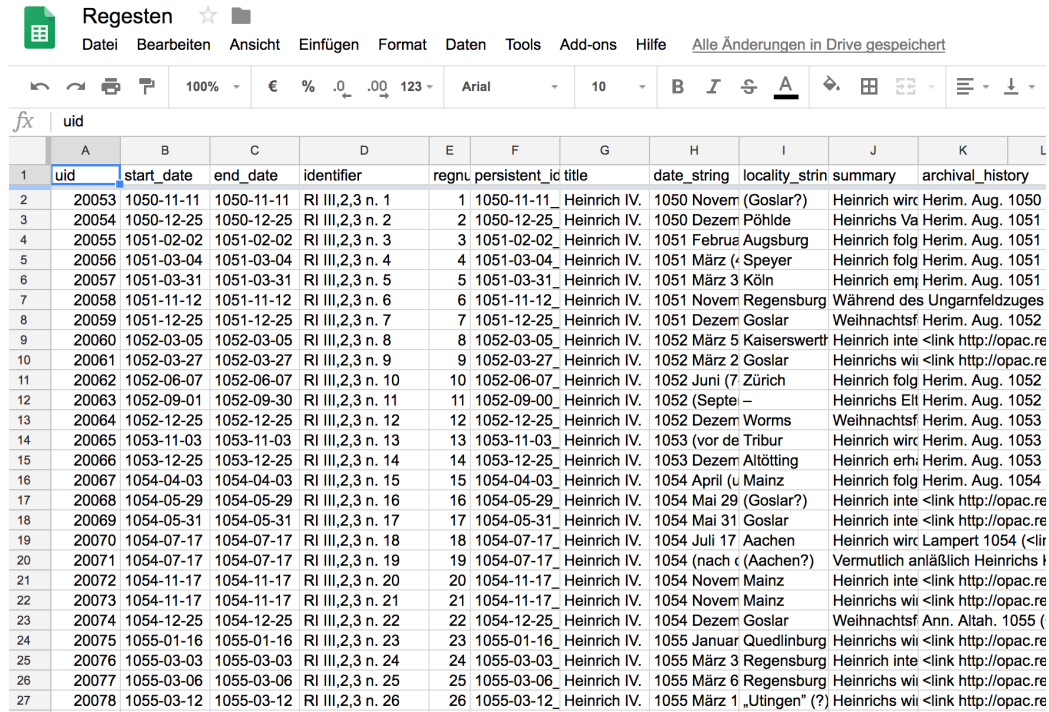
2 Regestenmodellierung im Graphen

2.1 Wie kommen die Regesten in den Graphen

In diesem Abschnitt wird beispielhaft an Hand der Regesten Kaiser Heinrichs IV. der Import der Online-Regesten in die Graphdatenbank neo4j durchgespielt.¹ Die Webseite der Regesta Imperii Online basiert auf dem Content-Management-System typo3, welches auf eine mysql-Datenbank aufbaut. In der Datenbank werden die Regesteninformationen in verschiedenen Tabellen

¹Die den Regesten Kaiser Heinrichs IV. umfassen folgende Bände: Böhmer, J. F., Regesta Imperii III. Salisches Haus 1024-1125. Tl. 2: 1056-1125. 3. Abt.: Die Regesten des Kaiserreichs unter Heinrich IV. 1056 (1050) - 1106. 1. Lief.: 1056 (1050) – 1065, bearb. von Struve, Tilman - Köln (u.a.) (1984). Böhmer, J. F., Regesta Imperii III. Salisches Haus 1024-1125. Tl. 2: 1056-1125. 3. Abt.: Die Regesten des Kaiserreichs unter Heinrich IV. 1056 (1050) - 1106. 2. Lief.: 1065–1075, bearb. von Struve, Tilman unter Mitwirkung von Lubich, Gerhard und Jäckel, Dirk - Köln (u.a.) (2010). Böhmer, J. F., Regesta Imperii III. Salisches Haus 1024-1125. Tl. 2: 1056-1125. 3. Abt.: Die Regesten des Kaiserreichs unter Heinrich IV. 1056 (1050) - 1106. 3. Lief.: 1076–1085, bearb. von Lubich, Gerhard nach Vorarbeiten von Struve, Tilman unter Mitwirkung von Jäckel, Dirk - Köln (u.a.) (2016). Böhmer, J. F., Regesta Imperii III. Salisches Haus 1024-1125. Tl. 2: 1056-1125. 3. Abt.: Die Regesten des Kaiserreichs unter Heinrich IV. 1056 (1050) - 1106. 4. Lief.: 1086–1105/06, bearb. von Lubich, Gerhard nach Vorarbeiten von Brauch, Daniel unter Mitwirkung von Weber, Matthias - Köln (u.a.) (2016). Böhmer, J. F., Regesta Imperii III. Salisches Haus 1024-1125. Tl. 2: 1056-1125. 3. Abt.: Die Regesten des Kaiserreichs unter Heinrich IV. 1056 (1050) - 1106. 5. Lief.: Die Regesten Rudolfs von Rheinfelden, Hermanns von Salm und Konrads (III.). Verzeichnisse, Register, Addenda und Corrigenda, bearbeitet von Lubich, Gerhard unter Mitwirkung von Junker, Cathrin; Klocke, Lisa und Keller, Markus - Köln (u.a.) (2018).

vorgehalten. Die Webseite bietet zum einen die Möglichkeit, die Regesten über eine REST-Schnittstelle im CEI-XML-Format oder als CSV-Dateien herunterzuladen. Für den Import in die Graphdatenbank bietet sich das CSV-Format an.



	A	B	C	D	E	F	G	H	I	J	K	L
	uid	start_date	end_date	identifier	regn	persistent	id	title	date_string	locality_string	summary	archival_history
2	20053	1050-11-11	1050-11-11	RI III,2,3 n. 1	1	1050-11-11		Heinrich IV.	1050 Novem (Goslar?)		Heinrich wir Herim. Aug. 1050	
3	20054	1050-12-25	1050-12-25	RI III,2,3 n. 2	2	1050-12-25		Heinrich IV.	1050 Dezem Pöhlde		Heinrichs Va Herim. Aug. 1051	
4	20055	1051-02-02	1051-02-02	RI III,2,3 n. 3	3	1051-02-02		Heinrich IV.	1051 Februa Augsburg		Heinrich folg Herim. Aug. 1051	
5	20056	1051-03-04	1051-03-04	RI III,2,3 n. 4	4	1051-03-04		Heinrich IV.	1051 März (Speyer		Heinrich folg Herim. Aug. 1051	
6	20057	1051-03-31	1051-03-31	RI III,2,3 n. 5	5	1051-03-31		Heinrich IV.	1051 März 3 Köln		Heinrich emj Herim. Aug. 1051	
7	20058	1051-11-12	1051-11-12	RI III,2,3 n. 6	6	1051-11-12		Heinrich IV.	1051 Novem Regensburg		Während des Ungarnfeldzuges	
8	20059	1051-12-25	1051-12-25	RI III,2,3 n. 7	7	1051-12-25		Heinrich IV.	1051 Dezem Goslar		Weihnachtsf Herim. Aug. 1052	
9	20060	1052-03-05	1052-03-05	RI III,2,3 n. 8	8	1052-03-05		Heinrich IV.	1052 März 5 Kaiserswerth		Heinrich inte <link http://opac.re	
10	20061	1052-03-27	1052-03-27	RI III,2,3 n. 9	9	1052-03-27		Heinrich IV.	1052 März 2 Goslar		Heinrichs wi <link http://opac.re	
11	20062	1052-06-07	1052-06-07	RI III,2,3 n. 10	10	1052-06-07		Heinrich IV.	1052 Juni (7 Zürich		Heinrich folg Herim. Aug. 1052	
12	20063	1052-09-01	1052-09-30	RI III,2,3 n. 11	11	1052-09-00		Heinrich IV.	1052 (Septe –		Heinrichs Elt Herim. Aug. 1052	
13	20064	1052-12-25	1052-12-25	RI III,2,3 n. 12	12	1052-12-25		Heinrich IV.	1052 Dezem Worms		Weihnachtsf Herim. Aug. 1053	
14	20065	1053-11-03	1053-11-03	RI III,2,3 n. 13	13	1053-11-03		Heinrich IV.	1053 (vor de Tribur		Heinrich wir Herim. Aug. 1053	
15	20066	1053-12-25	1053-12-25	RI III,2,3 n. 14	14	1053-12-25		Heinrich IV.	1053 Dezem Altötting		Heinrich erh: Herim. Aug. 1053	
16	20067	1054-04-03	1054-04-03	RI III,2,3 n. 15	15	1054-04-03		Heinrich IV.	1054 April (u Mainz		Heinrich folg Herim. Aug. 1054	
17	20068	1054-05-29	1054-05-29	RI III,2,3 n. 16	16	1054-05-29		Heinrich IV.	1054 Mai 29 (Goslar?)		Heinrich inte <link http://opac.re	
18	20069	1054-05-31	1054-05-31	RI III,2,3 n. 17	17	1054-05-31		Heinrich IV.	1054 Mai 31 Goslar		Heinrich inte <link http://opac.re	
19	20070	1054-07-17	1054-07-17	RI III,2,3 n. 18	18	1054-07-17		Heinrich IV.	1054 Juli 17 Aachen		Heinrich wir Lampert 1054 (<li	
20	20071	1054-07-17	1054-07-17	RI III,2,3 n. 19	19	1054-07-17		Heinrich IV.	1054 (nach c (Aachen?)		Vermutlich anlässlich Heinrichs i	
21	20072	1054-11-17	1054-11-17	RI III,2,3 n. 20	20	1054-11-17		Heinrich IV.	1054 Novem Mainz		Heinrich inte <link http://opac.re	
22	20073	1054-11-17	1054-11-17	RI III,2,3 n. 21	21	1054-11-17		Heinrich IV.	1054 Novem Mainz		Heinrichs wi <link http://opac.re	
23	20074	1054-12-25	1054-12-25	RI III,2,3 n. 22	22	1054-12-25		Heinrich IV.	1054 Dezem Goslar		Weihnachtsf Ann. Altah. 1055 (
24	20075	1055-01-16	1055-01-16	RI III,2,3 n. 23	23	1055-01-16		Heinrich IV.	1055 Januar Quedlinburg		Heinrichs wi <link http://opac.re	
25	20076	1055-03-03	1055-03-03	RI III,2,3 n. 24	24	1055-03-03		Heinrich IV.	1055 März 3 Regensburg		Heinrich inte <link http://opac.re	
26	20077	1055-03-06	1055-03-06	RI III,2,3 n. 25	25	1055-03-06		Heinrich IV.	1055 März 6 Regensburg		Heinrichs wi <link http://opac.re	
27	20078	1055-03-12	1055-03-12	RI III,2,3 n. 26	26	1055-03-12		Heinrich IV.	1055 März 1 „Utingen“ (?)		Heinrichs wi <link http://opac.re	

Figure 1: Regesten als CSV-Datei

In der CSV-Datei finden sich die oben erläuterten einzelnen Elemente der Regesten in jeweils eigenen Spalten. Die Spaltenüberschrift gibt Auskunft zum Inhalt der jeweiligen Spalte.

2.1.1 Import mit dem LOAD CSV-Befehl

Mit dem Befehl `LOAD CSV` können die CSV-Dateien mit den Regesten in die Graphdatenbank neo4j importiert werden.² Hierfür muss die Datenbank

²Zu Installation und ersten Schritten von neo4j vgl. in der Einleitung den Abschnitt zu Installation und Start.

aber Zugriff auf die CSV-Daten haben. Dies ist einerseits über den im Datenbankverzeichnis vorhandene Ordner **import** oder über eine URL, unter der die CSV-Datei abrufbar ist, möglich. Da sich die einzelnen Zugriffswege auf den **import**-Ordner von Betriebssystem zu Betriebssystem unterscheiden, wird hier beispielhaft der Import über eine URL vorgestellt. Hierfür wird ein Webserver benötigt, auf den man die CSV-Datei hochlädt und sich anschließend die Webadresse für den Download der Datei notiert.

2.1.2 Google-Docs für den CSV-Download

Es ist aber auch möglich, CSV-Daten in Google-spreadsheets zu speichern und dort eine URL für den Download der Daten zu erstellen. Zunächst benötigt man hierfür einen Google-Account. Anschließend öffnet man Google-Drive und erstellt dort eine leere Google-Tabellen-Datei (Google-Spreadsheet) in der man dann die CSV-Datei kopieren kann.

Wichtig ist nun, die Datei zur Ansicht freizugeben (Klick auf **Freigeben** oben rechts im Fenster dann **Link zum Freigeben** abrufen und anschließend **Fertig** bestätigen). Jetzt ist die CSV-Datei in Google-Docs gespeichert und kann auch von anderen Personen über den Freigabelink angesehen werden. Für den Import in die Graphdatenbank benötigen wir aber einen Download im CSV-Format. Diesen findet man unter **Datei/Herunterladen als/Kommagetrennte Werte.csv** aktuelles Tabellenblatt.

Damit erhält man das aktuelle Tabellenblatt als CSV-Download. Anschließend muss nun im Browser unter Downloads der Download-Link der Datei gesucht und kopiert werden.

2.1.3 Regestenmodellierung im Graphen

Mit dem **LOAD CSV**-Befehl stehen die Informationen der Regestentabelle nun für die weitere Verarbeitung zur Verfügung. Nun muss festgelegt werden, wie diese Informationen im Graphen modelliert werden sollen. Daher wird im nächsten Schritt das Modell der Regesten im Graphen vorgestellt (siehe Abbildung).

In den Abbildungen finden sich beispielhaft das Regest RI III,2,3 Nr. 1487, einmal in der Ansicht der Onlineregesten und in der zweiten Abbildung als

Regesten ☆

Datei Bearbeiten Ansehen Einfügen Format Daten Tools Add-ons Hilfe Letzte Änderung am 26. Januar

150% € % .0 .00 123 Arial 10 B I S A

fx uid

	A	B	C	D	E	F	G	H
1	uid	start_date	end_date	identifizier	regnum	persistent ic	title	date st
2	20053	1050-11-11	1050-11-11	RI III,2,3 n. 1	1	1050-11-11	Heinrich IV.	1050 N
3	20054	1050-12-25	1050-12-25	RI III,2,3 n. 2	2	1050-12-25	Heinrich IV.	1050 D
4	20055	1051-01-01	1051-01-01	RI III,2,3 n. 3	3	1051-01-01	Heinrich IV.	1051 F
5	20056	1051-01-01	1051-01-01	RI III,2,3 n. 4	4	1051-01-01	Heinrich IV.	1051 M
6	20057	1051-01-01	1051-01-01	RI III,2,3 n. 5	5	1051-01-01	Heinrich IV.	1051 M
7	20058	1051-01-01	1051-01-01	RI III,2,3 n. 6	6	1051-01-01	Heinrich IV.	1051 N
8	20059	1051-01-01	1051-01-01	RI III,2,3 n. 7	7	1051-01-01	Heinrich IV.	1051 D
9	20060	1052-01-01	1052-01-01	RI III,2,3 n. 8	8	1052-01-01	Heinrich IV.	1052 M
10	20061	1052-01-01	1052-01-01	RI III,2,3 n. 9	9	1052-01-01	Heinrich IV.	1052 M
11	20062	1052-01-01	1052-01-01	RI III,2,3 n. 10	10	1052-01-01	Heinrich IV.	1052 J
12	20063	1052-01-01	1052-01-01	RI III,2,3 n. 11	11	1052-01-01	Heinrich IV.	1052 S
13	20064	1052-01-01	1052-01-01	RI III,2,3 n. 12	12	1052-01-01	Heinrich IV.	1052 D
14	20065	1053-01-01	1053-01-01	RI III,2,3 n. 13	13	1053-01-01	Heinrich IV.	1053 (v
15	20066	1053-01-01	1053-01-01	RI III,2,3 n. 14	14	1053-01-01	Heinrich IV.	1053 D
16	20067	1054-01-01	1054-01-01	RI III,2,3 n. 15	15	1054-01-01	Heinrich IV.	1054 A
17	20068	1054-01-01	1054-01-01	RI III,2,3 n. 16	16	1054-01-01	Heinrich IV.	1054 M
18	20069	1054-01-01	1054-01-01	RI III,2,3 n. 17	17	1054-01-01	Heinrich IV.	1054 M
19	20070	1054-01-01	1054-01-01	RI III,2,3 n. 18	18	1054-01-01	Heinrich IV.	1054 J
20	20071	1054-07-17	1054-07-17	RI III,2,3 n. 19	19	1054-07-17	Heinrich IV.	1054 (r
21	20072	1054-11-17	1054-11-17	RI III,2,3 n. 20	20	1054-11-17	Heinrich IV.	1054 N
22	20073	1054-11-17	1054-11-17	RI III,2,3 n. 21	21	1054-11-17	Heinrich IV.	1054 N
23	20074	1054-12-25	1054-12-25	RI III,2,3 n. 22	22	1054-12-25	Heinrich IV.	1054 D
24	20075	1055-01-16	1055-01-16	RI III,2,3 n. 23	23	1055-01-16	Heinrich IV.	1055 J
25	20076	1055-03-03	1055-03-03	RI III,2,3 n. 24	24	1055-03-03	Heinrich IV.	1055 M
26	20077	1055-03-03	1055-03-03	RI III,2,3 n. 25	25	1055-03-03	Heinrich IV.	1055 M

tx_hisodat_domain_model_dateranges

Für andere freigeben

Link zum Freigeben abrufen

Link freigabe ist aktiviert. Weitere Informationen

Jeder mit dem Link darf die Datei ansehen

Link kopieren

https://docs.google.com/spreadsheets/d/1GLQIH9LA5btZc-VCRd-8f9BjyIDvwu29Fv

Personen

Namen oder E-Mail-Adressen eingeben...

Fertig

Erweitert

Figure 2: Freigabe der Datei zum Ansehen für Dritte!

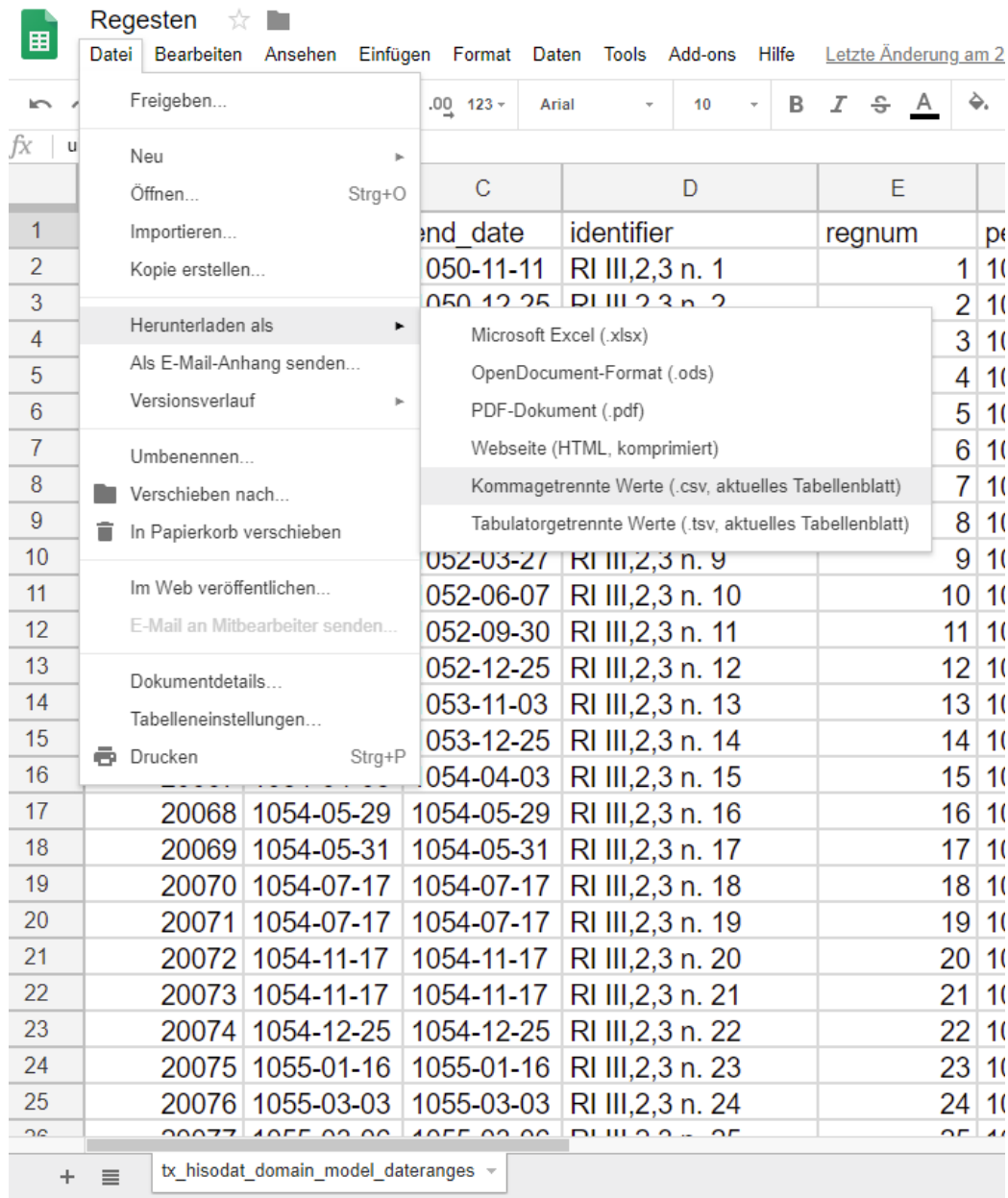


Figure 3: Herunterladen als CSV-Datei

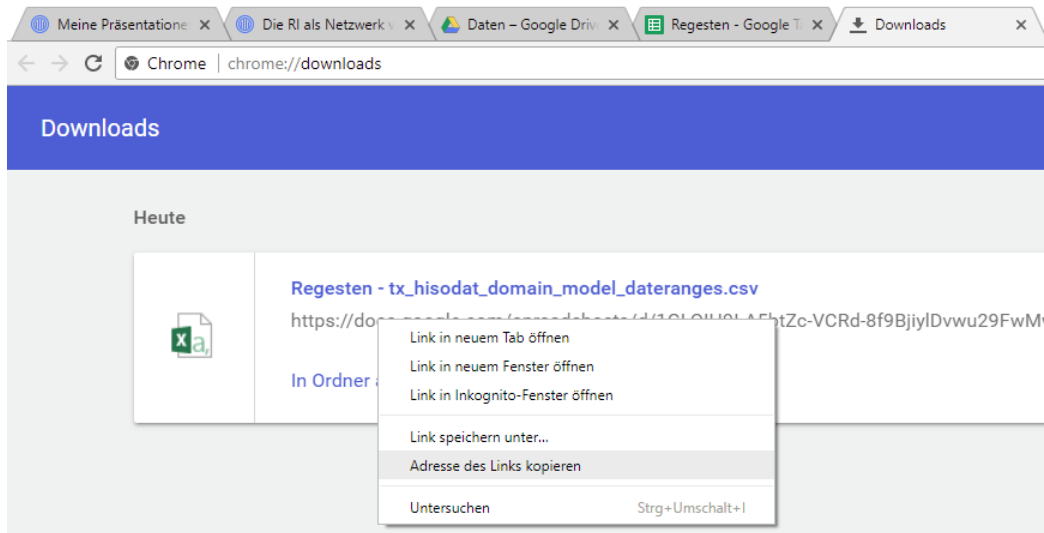


Figure 4: Download-Link der CSV-Datei

Modell im Graphen (neben anderen Regesten).

Die gelben Knoten sind die Regesten. Aus den Angaben des Regests werden mit dem o.a. Befehl noch ein Datums-knoten und ein Ortsknoten erstellt. Mit dem ersten **CREATE**-Befehl werden die Regesten erstellt. Die **MERGE**-Befehle erzeugen ergänzende Knoten für die Datumsangaben und die Ausstellungsorte. Nun ist es aber so, dass Ausstellungsort und Ausstellungsdatum mehrfach vorkommen können. Daher wird hier nicht der **CREATE**-Befehl sondern der **MERGE**-Befehl verwendet. Dieser funktioniert wie der **CREATE**-Befehl, prüft aber vorher, ob in der Datenbank ein solcher Knoten schon existiert. Falls es ihn noch nicht gibt, wird er erzeugt, wenn es ihn schon gibt, wird er der entsprechenden Variable zugeordnet. Anschließend wird die Kante zwischen Regesten-knoten und Ausstellungsorts-knoten und Regesten-knoten und Datums-knoten erstellt. In der folgenden Tabelle werden die einzelnen Befehle dargestellt und kommentiert.

2.1.4 Indexe Erstellen

Bevor nun mit dem Import begonnen wird, ist es für die Beschleunigung des Importprozesses von Vorteil vorher Indexe für häufig genutzte Properties zu

Heinrich IV. - RI III,2,3 n. 1487

URI
Merken

1103 Juni 29, Lüttich

Heinrich feiert das Fest der Apostel, wobei sich Graf Robert von Flandern im Beisein mehrerer Fürsten unterwirft, namentlich der Erzbischöfe Friedrich von Köln und Bruno von Trier, der Bischöfe Otbert von Lüttich, Burchard von Münster, Burchard von Utrecht, Herzog Heinrich von Niederlothringen sowie mehrerer Grafen.

Überlieferung/Literatur

Tagesdatum bei Ann. Patherbr. 1103 (Scheffer-Boichorst 107 f.): *in festo apostolorum Petri et Pauli*; Gesta Galcheri Episcopi Cameracensis (SS 14, 202); Sigeb. Gembl. 1103 (SS 6, 368); Ann. Elnon. maior. 1103 (SS 5, 14); Ann. Leod., Cont. 1103 (SS 4, 29); Ann. Aquens. 1103 (SS 16, 685); Ann. necrol. Prum. 1103 (SS 13, 223).

Kommentar

Zur Lehensterminologie in den Gesta Galcheri (*Facto palam hominio, iurat Robertus Henrico, promittit, miles domino, quia fidelis amodo*) vgl. G a n s h o f, Was ist das Lehnswesen? (1961) 72 f. – Zum Ereigniskontext vgl. Reg. 1475; Heinrich V. führte bereits 1107 wieder einen Feldzug gegen Robert von Flandern; vgl. B o s h o f, Bischofskirchen von Passau und Regensburg (Salier 2, 1991) 148. – Vgl. K i l i a n, Itinerar 127 mit der Vermutung einer weiteren, der Unterwerfung Roberts vorangehenden Heerfahrt Heinrichs nach Flandern; M e y e r v o n K n o n a u, Jbb. 5, 179 f.; T. R e u t e r, Unruhestiftung (Salier 3, 1991) 324–326.

Figure 5: RI III,2,3 n. 1487, in: Regesta Imperii Online, URI: <http://www.regesta-imperii.de/id/cf75356b-bd0d-4a67-8aeb-3ae27d1dcefa>.

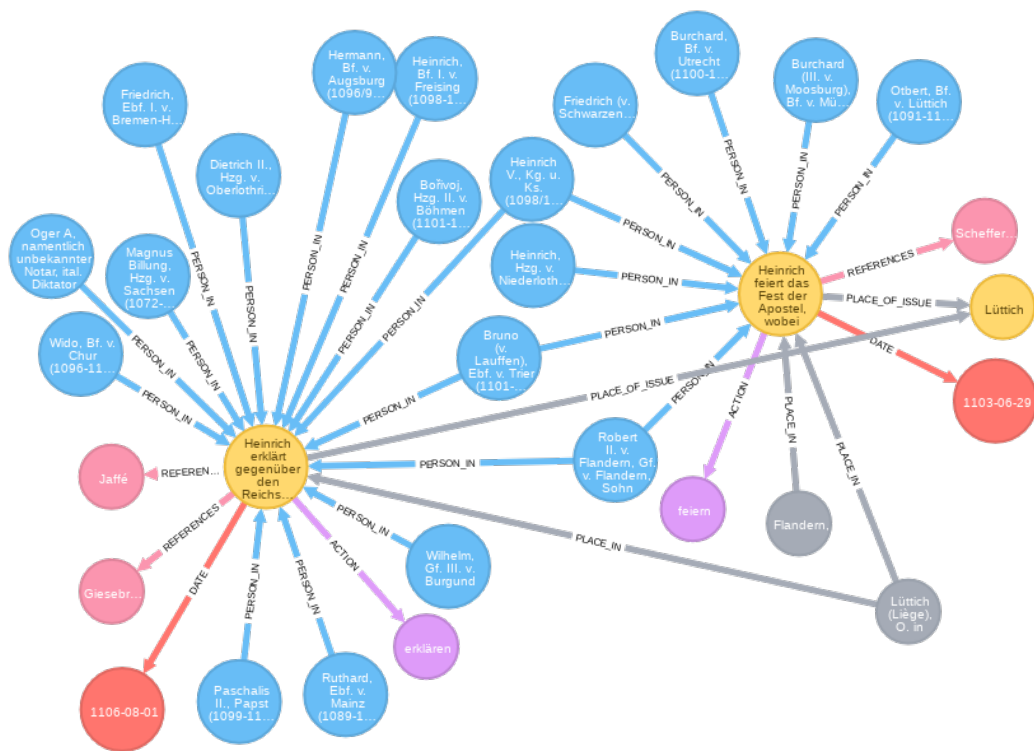


Figure 6: Das Regest im Graphen.

erstellen.

```
// vorab Index erzeugen -> Import wird schneller
CREATE INDEX ON :Regesta(ident);
CREATE INDEX ON :Regesta(regnum);
CREATE INDEX ON :Regesta(persistentIdentifier);
CREATE INDEX ON :Regesta(registerId);
CREATE INDEX ON :Regesta(heftId);
CREATE INDEX ON :Regesta(placeOfIssue);
CREATE INDEX ON :Regesta(origPlaceOfIssue);
CREATE INDEX ON :Date(startDate);
CREATE INDEX ON :Place(original);
CREATE INDEX ON :Place(normalizedGerman);
CREATE INDEX ON :Lemma(lemma);
CREATE INDEX ON :Literature(literatur);
CREATE INDEX ON :Reference(reference);
CREATE INDEX ON :IndexEntry(registerId);
CREATE INDEX ON :IndexEntry(nodeId);
CREATE INDEX ON :Regesta(latLong);
CREATE INDEX ON :IndexPlace(registerId);
CREATE INDEX ON :IndexEvent(registerId);
CREATE INDEX ON :IndexPerson(registerId);
```

2.1.5 Erstellen der Regestenknoten

Mit dem folgenden cypher-Query werden die Regestenknoten in der Graphdatenbank erstellt:

```
// Regestenknoten erstellen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/1GLQIH9LA5btZc-
CREATE (r:Regesta {regid:line.persistentIdentifier, text:line.summary,
    archivalHistory:line.archival_history, date:line.date_string,
    ident:line.identifier, regnum:line.regnum,
    origPlaceOfIssue:line.locality_string, startDate:line.start_date,
    endDate:line.end_date})
MERGE (d:Date {startDate:line.start_date, endDate:line.end_date})
MERGE (r)-[:DATE]->(d)
RETURN count(r);
```

Im folgenden werden die einzelnen Teile des Import-Befehls erläutert:

Befehl	Variablen	Bemerkungen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/ ..." AS line	line	Import der CSV-Dateien. Es wird jeweils eine Zeile an die Variable line weitergegeben
CREATE(r:Regesta {regid:line.persistentIdentifier, text:line.summary, archivalHis- tory:line.archival_history, date:line.date_string MERGE (d:Date {start- Date:line.start_date, enddate:line.end_date})	line.persistent_ Identifier, line.summary etc.	Erstellung des Regestenknötens. Für die weiteren Befehle steht der neu erstellt Regestenknöt unter der Variable r zur Verfügung.
MERGE (r)- [:HAT_DATUM]->(d)	line.start_date und line.end_date	Es wird geprüft, ob ein Datumsknöt mit der Datumsangabe schon existiert, falls nicht, wird er erstellt. In jedem Fall steht anschließend der Datumsknöt unter der Variable d zur Verfügung.
	(r) ist der Regestenknöt, (d) ist der Datumsknöt	Zwischen Regestenknöt und Datumsknöt wird eine HAT_DATUM-Kante erstellt.

2.1.6 Erstellen der Ausstellungsorte

In den Kopfzeilen der Regesten ist, soweit bekannt, der Ausstellungsort der Urkunde vermerkt. Im Rahmen der Arbeiten an den Regesta Imperii Online wurden diese Angaben zusammengestellt und soweit möglich die Orte identifiziert, so dass diese Angabe nun beim Import der Regesten in den Graphen berücksichtigt werden kann. Insgesamt befinden sich in den Regesta Imperii über 12.000 verschiedene Angaben für Ausstellungsorte, wobei sie sich aber auch teilweise auf den gleichen Ort beziehen können (Wie z.B. Aachen, Aquisgrani, Aquisgradi, Aquisgranum, coram Aquisgrano etc.).

Allein mit der Identifizierung der 1.000 häufigsten Ortsangaben konnte schon die überwiegende Mehrzahl der Ausstellungsorte georeferenziert werden. Die Daten zur Ortsidentifizierung liegen auch in einer Google-Tabelle vor.

Mit dem folgenden cypher-Query werden die Ausstellungsorte in die Graphdatenbank importiert:

```
// RI-Ausstellungsorte-geo erstellen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/
13_f6Vja4HfOpju9RVDubHiMLzS6Uoa7MIOHFeg5V7lw/
export?format=csv&id=13_f6Vja4HfOpju9RVDubHiMLzS6Uoa7MIOHFeg5V7lw
&gid=420547059"
AS line
WITH line
WHERE line.Lat IS NOT NULL
AND line.normalisiertDeutsch IS NOT NULL
MATCH (r:Regesta {origPlaceOfIssue:line.Original})
MERGE (p:Place {normalizedGerman:line.normalisiertDeutsch,
  longitude:line.Long, latitude:line.Lat})
WITH r, p, line
MERGE (r)-[rel:PLACE_OF_ISSUE]->(p)
SET p.wikidataId = line.wikidataId
SET p.name = line.name
SET p.gettyId = line.GettyId
SET p.geonamesId = line.GeonamesId
SET rel.original = line.Original
SET rel.alternativeName = line.Alternativname
SET rel.commentary = line.Kommentar
SET rel.allocation = line.Zuordnung
SET rel.state = line.Lage
SET rel.certainty = line.Sicherheit
SET rel.institutionInCity = line.InstInDerStadt
RETURN count(p);
```

Da Import-Query etwas komplexer ist, wird er im folgenden näher erläutert. Nach dem `LOAD CSV WITH HEADERS FROM`-Befehl wird zunächst überprüft, ob der jeweils eingelesene Eintrag in der Spalte `line.lat` und in der Spalte `line.normalisiertDeutsch` Einträge hat. Ist dies der Fall wird überprüft, ob es einen Regestenknoten gibt, der einen Ausstellungsorteintrag hat, der

der Angabe in der Spalte **Original** entspricht. Diese Auswahl ist notwendig, da in der Tabelle die Ausstellungsorte der gesamten Regesta Imperii enthalten sind. Für diesen Import sollen aber nur jene angelegt werden, die für die Regesten Kaiser Heinrichs IV. relevant sind. Mit dem **MERGE**-Befehl wird der **Place**-Knoten erstellt (falls es ihn nicht schon gibt) und anschließend mit dem Regestenknoten verknüpft. Schließlich werden noch weitere Details der Ortsangabe im **Place**-Knoten und in den **PLACE_OF_ISSUE**-Kanten ergänzt.

2.1.7 Koordinaten der Ausstellungsorte

Mit dem folgenden Query werden die Koordinatenangaben zu Höhen- und Breitengraden der Ausstellungsorte (**Place**-Knoten), die in den Property **latitude** und **longitude** abgespeichert sind, in der neuen Property **LatLong** zusammengefasst und in **point**-Werte umgewandelt. Seit Version 3 kann neo4j mit diesen Werten Abstandsberechnungen durchführen (Mehr dazu siehe unten bei den Auswertungen).

```
// Regesten und Ausstellungsorte mit Koordinaten der Ausstellungsorte versehen
MATCH (r:Regesta)-[:PLACE_OF_ISSUE]->(o:Place)
SET r.latLong = point({latitude: tofloat(o.latitude),
    longitude: tofloat(o.longitude)})
SET o.latLong = point({latitude: tofloat(o.latitude),
    longitude: tofloat(o.longitude)})
SET r.placeOfIssue = o.normalizedGerman
SET r.latitude = o.latitude
SET r.longitude = o.longitude;
```

2.1.8 Ausstellungsdatum

In den Regesta Imperii Online sind die Datumsangaben der Regesten iso-konform im Format **JJJJ-MM-TT** (also Jahr-Monat-Tag) abgespeichert. neo4j behandelt diese Angaben aber als **String**. Um Datumsberechnungen durchführen zu können, müssen die Strings in neo4j-interne Datumswerte umgerechnet werden. Der cypher-Query hierzu sieht wie folgt aus:

```
// Date in neo4j-Datumsformat umwandeln
MATCH (n:Regesta)
```

```

SET n.isoStartDate = date(n.startDate);
MATCH (n:Regesta)
SET n.isoEndDate = date(n.endDate);
MATCH (d:Date)
SET d.isoStartDate = date(d.startDate);
MATCH (d:Date)
SET d.isoEndDate = date(d.endDate);

```

Zunächst werden mit dem `MATCH`-Befehl alle Regestenknoten aufgerufen. Anschließend wird für jeden Regestenknoten aus der String-Property `startDate` die Datumsproperty `isoStartDate` berechnet und im Regestenknoten abgespeichert. Mit Hilfe der Property können dann Datumsangaben und Zeiträume abgefragt werden (Beispiel hierzu unten in der Auswertung).

2.2 Exkurs 1: Herrscherhandeln in den Regesta Imperii

Bisher wurden beim Import der Regesten in den Graphen nur die in den Online-Regesten bereits angelegten Angaben importiert. Im folgenden Schritt werden nun in einem kleinen Exkurs die Regestentexte selbst analysiert und anschließend die Graphdatenbank um eine weitere Informationsebene ergänzt. Regesten sind in ihrer Struktur stark formalisiert. Meist wird mit dem ersten Verb im Regest das Herrscherhandeln beschrieben. Um dies auch digital auswerten zu können, haben wir in einem kleinen Testprojekt mit Hilfe des Stuttgart-München Treetaggers³ aus jedem Regest das erste Verb extrahiert und normalisiert. Die Ergebnisse sind in folgender Tabelle einsehbar. Diese Tabelle wird mit dem folgenden cypher-Query in die Graphdatenbank eingelesen.

```

// ReggH4-Herrscherhandeln
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/1n1bZmQYcT1E3Z5"
AS line FIELDTERMINATOR ','
MATCH (r:Regesta{ident:line.regid})
MERGE (l:Lemma{lemma:line.Lemma})
MERGE (r)-[:ACTION]->(l);

```

³Zum Treetagger vgl. <http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>.

Dabei wird zunächst mit dem **MATCH**-Befehl das jeweilige Regest gesucht, anschließend mit dem **MERGE**-Befehl der **Lemma-Knoten** für das Herrscherhandeln angelegt (falls noch nicht vorhanden) und schließlich der **Regesta-knoten** mit dem **Lemma-Knoten** über eine **ACTION**-Kante verbunden. In der folgenden Abbildung ist ein Ausschnitt mit Regesten und den verknüpften Lemmaknoten dargestellt.

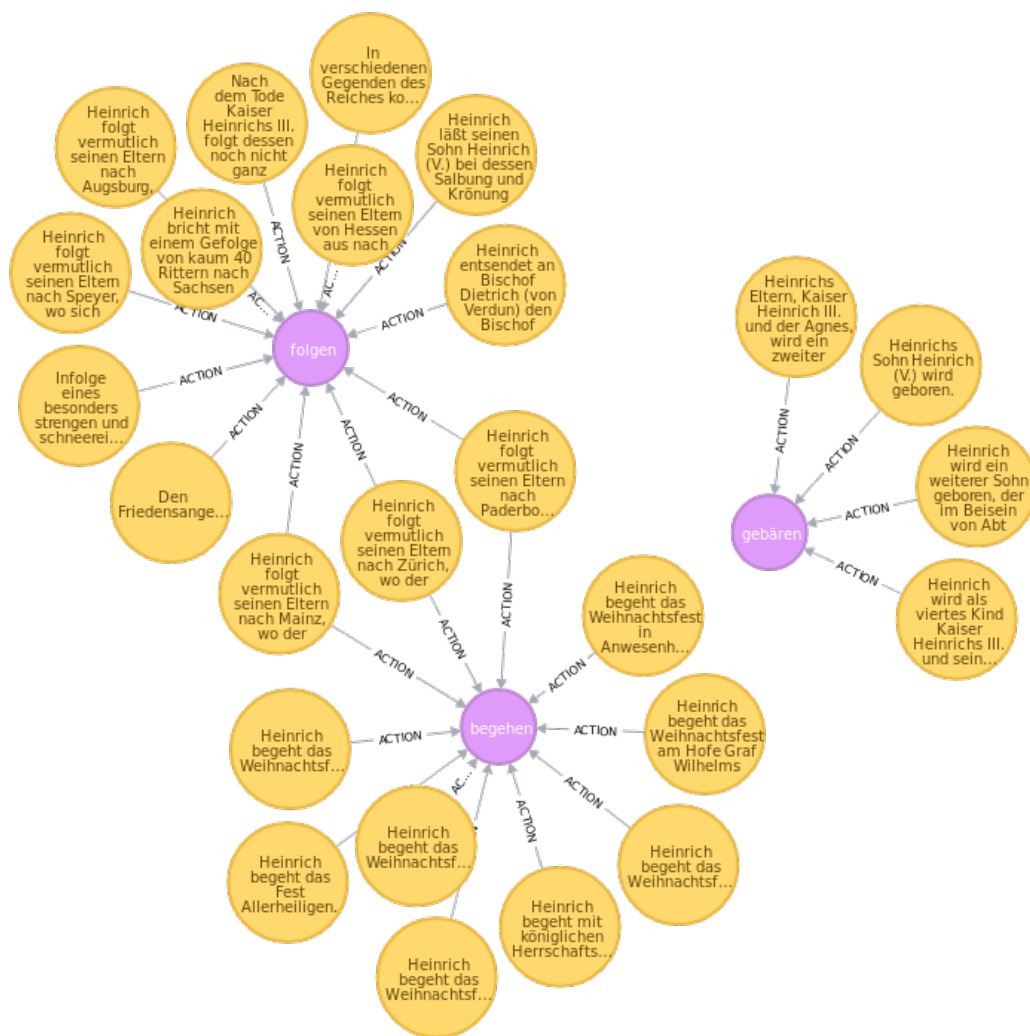


Figure 7: Herrscherhandeln im Graphen.

2.3 Zitationsnetzwerke in den Regesta Imperii

In vielen Online-Regesten ist die zitierte Literatur mit dem Regesta-Imperii-Opac verlinkt. Da es sich um URLs handelt, sind diese Verweise eindeutig. Andererseits lassen sie sich mit regulären Ausdrücken aus den Regesten extrahieren. Mit folgendem Query werden aus den Überlieferungsteilen der Regesten die mit dem Opac verlinkten Literaturangaben extrahiert und jede Literaturangabe als **Reference**-Knoten angelegt.

```
// ReggH4-Literaturnetzwerk erstellen
MATCH (reg:Regesta)
WHERE reg.archivalHistory CONTAINS "link"
UNWIND apoc.text.regexGroups(reg.archivalHistory,
"<link (\\S+)>(\\S+)</link>") as link
MERGE (ref:Reference {url:link[1]})
ON CREATE SET ref.title=link[2]
MERGE (reg)-[:REFERENCES]->(ref);
```

Da dies mit dem **MERGE**-Befehl geschieht, wird in der Graphdatenbank jeder Literaturtitel nur einmal angelegt. Anschließend werden die **Reference**-Knoten mit den Regesten über **REFERENCES**-Kanten verbunden. Zu den Auswertungsmöglichkeiten vgl. unten den Abschnitt zu den Auswertungsperspektiven.

2.4 Import der Registerdaten in die Graphdatenbank

2.4.1 Vorbereitung der Registerdaten

Register spielen für die Erschließung von gedrucktem Wissen eine zentrale Rolle, da dort in alphabetischer Ordnung die im Werk vorkommenden Entitäten (z.B. Personen und Orte) hierarchisch gegliedert aufgeschlüsselt werden. Für die digitale Erschließung der Regesta Imperii sind Register von zentraler Bedeutung, da mit ihnen die in den Regesten vorkommenden Personen und Orte bereits identifiziert vorliegen. Für den Import in die Graphdatenbank wird allerdings eine digitalisierte Fassung des Registers benötigt. Im Digitalisierungsprojekt Regesta Imperii Online wurden Anfang der 2000er Jahre auch die gedruckt vorliegenden Register digitalisiert. Sie dienen nun als Grundlage für die digitale Registererschließung der Regesta Imperii. Im

hier gezeigten Beispiel werden die Regesten Kaiser Heinrichs IV. und das dazugehörige Register importiert. Da der letzte Regestenband der Regesten Kaiser Heinrichs IV. mit dem Gesamtregister erst vor kurzem gedruckt wurde, liegen hier aktuelle digitale Fassung von Registern und Regesten vor. Die für den Druck in Word erstellte Registerfassung wird hierfür zunächst in eine hierarchisch gegliederte XML-Fassung konvertiert, damit die Registerhierarchie auch maschinenlesbar abgelegt ist.

```
<Stufe0 id="H4P00005">
  <Inhalt>Abiram, biblische Gestalt, Sohn Eliabs, Bruder Dathans, Auflehner gegen
    Moses</Inhalt>
  <Regestennummer>
    <r>762</r>
  </Regestennummer>
</Stufe0>
<Stufe0 id="H4P00006">
  <Inhalt>AC → Gottschalk v. Aachen</Inhalt>
</Stufe0>
<Stufe0 id="H4P00007">
  <Inhalt>Achalmer, Adelsgeschlecht aus Schwaben</Inhalt>
  <Regestennummer>
    <r>363</r>
  </Regestennummer>
  <Stufe1>
    <Inhalt><vw/>→ Liutold, Gf. v. Achalm</Inhalt>
  </Stufe1>
  <Stufe1>
    <Inhalt><vw/>→ Werner (v. Achalm), Bf. II. v. Straßburg</Inhalt>
  </Stufe1>
  <Stufe1>
    <Inhalt><vw/>→ Williberga</Inhalt>
  </Stufe1>
</Stufe0>
```

Figure 8: Ausschnitt aus dem XML-Register der Regesten Heinrichs IV.

In der XML-Fassung sind die inhaltlichen Bereiche und die Abschnitte für die Regestennummern jeweils extra in die Tags `<Inhalt>` und `<Regestennummer>` eingefasst. Innerhalb des Elements `<Regestennummer>` ist dann nochmal jede einzelne Regestennummer in `<r>`-Tags eingefasst. Die aus dem gedruckten Register übernommenen Verweise sind durch ein leeres `<vw/>`-Element gekennzeichnet.

Die in XML vorliegenden Registerdaten werden anschließend mit Hilfe von TuStep in einzelne CSV-Tabellen zerlegt.

In einer Tabelle werden alle Entitäten aufgelistet und jeweils mit einer ID versehen.

6	H4P00005	Abiram, biblische Gestalt, Sohn Eliabs, Bruder Dathans, Auflehner gegen Moses			
7	H4P00006	AC → Gottschalk v. Aachen			
8	H4P00007	Achalmer, Adelsgeschlecht aus Schwaben			
9	H4P00008	Aczo, Sohn Rudolf Rubias, Bruder Attos			
10	H4P00009	Adalbero, Mgf. d. karantanischen Mark			
11	H4P00010	Adalbero, Gf. v. Ebersberg, Gem. Richildes			
12	H4P00011	Adalbero, Edelfreier			
13	H4P00012	Adalbero A (AA), namentlich unbekannter Bamberger Diktator			

Figure 9: Ausschnitt der Entitätentabelle des Registers der Regesten Heinrichs IV.

ID	regnum	regnum2	name1	name2					
H4P00001	805	805		<i>A.</i>,</td></tr><tr><td>H4P00002</td><td>1517</td><td>1517</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00003</td><td>1519</td><td>1519</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00005</td><td>762</td><td>762</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00007</td><td>363</td><td>363</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00008</td><td>1056</td><td>1056</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00009</td><td>714</td><td>714</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00010</td><td>78</td><td>78</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00011</td><td>331</td><td>331</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>666</td><td>666</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>717</td><td>717</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>959</td><td>959</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>1400</td><td>1400</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>1403</td><td>1403</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>1420</td><td>1420</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>1481</td><td>1481</td><td></td><td><i>A.</i>,</td></tr><tr><td>H4P00012</td><td>1485</td><td>1485</td><td></td><td><i>A.</i>,</td></tr></table>					

Figure 10: Ausschnitt der Verknüpfungstabelle des Registers der Regesten Heinrichs IV.

In der anderen Tabelle werden die Verknüpfungen zwischen Registereinträgen und den Regesten aufgelistet. Der Registereintrag Adalbero kommt also in mehreren Regesten vor. Da das Register der Regesten Heinrichs IV. nur zwei Hierarchiestufen enthält, in denen beispielsweise verschiedene Amtsphasen ein und derselben Person unterschieden werden, wurden diese beim Import zusammengefasst.⁴ Damit gibt es pro Person jeweils nur einen Knoten.

2.4.2 Import der Registerdaten in die Graphdatenbank

Im Gegensatz zu den Regesten Kaiser Friedrichs III., bei denen Orte und Personen in einem Register zusammengefasst sind, haben die Regesten Kaiser Heinrich IV. getrennte Orts- und Personenregister. Die digitalisierten Registerdaten können hier eingesehen werden. In dem Tabellendokument befinden sich insgesamt drei Tabellen. In der Tabelle Personen sind die Einträge des Personenregisters aufgelistet und in der Tabelle Orte befindet sich die Liste aller Einträge des Ortsregisters. Schließlich enthält die Tabelle `APPEARS_IN` Information dazu, welche Personen oder Orte in welchen Regesten genannt sind. Der folgende cypher-Query importiert die Einträge der Personentabelle in die Graphdatenbank und erstellt für jeden Eintrag einen Knoten vom Typ `:IndexPerson`:

```
// Registereinträge Personen erstellen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/12T-RD1Ct4aAUMN
AS line
CREATE (:IndexPerson {registerId:line.ID, name1:line.name1});
```

Mit dem folgenden cypher-Query werden nach dem gleichen Muster aus der Tabelle `Orte` die Ortseinträge in die Graphdatenbank importiert.

```
// Registereinträge Orte erstellen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/12T-RD1Ct4aAUMN
AS line
CREATE (:IndexPlace {registerId:line.ID, name1:line.name1});
```

⁴Vgl. die Vorbemerkung zum Register in Böhmer, J. F., *Regesta Imperii III. Salisches Haus 1024-1125. Tl. 2: 1056-1125. 3. Abt.: Die Regesten des Kaiserreichs unter Heinrich IV. 1056 (1050) - 1106. 5. Lief.: Die Regesten Rudolfs von Rheinfelden, Hermanns von Salm und Konrads (III.). Verzeichnisse, Register, Addenda und Corrigenda*, bearbeitet von Lubich, Gerhard unter Mitwirkung von Junker, Cathrin; Klocke, Lisa und Keller, Markus - Köln (u.a.) (2018), S. 291.

Die beiden Befehle greifen also auf verschiedene Tabellenblätter des gleichen Google-Tabellendokuments zu, laden es als CSV-Daten und übergeben die Daten zeilenweise an die weiteren Befehle (Hier an den `MATCH`- und den `CREATE`-Befehl). Im nächsten Schritt werden nun mit den Daten der `APPEARS_IN`-Tabelle die Verknüpfungen zwischen den Registereinträgen und den Regesten erstellt.

```
// PLACE_IN-Kanten für Orte erstellen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/12T-RD1Ct4aAUMN"
AS line
MATCH (from:IndexPlace {registerId:line.ID})
MATCH (to:Regesta {regnum:line.regnum2})
CREATE (from)-[:PLACE_IN {regnum:line.regnum,
    name1:line.name1, name2:line.name2}]->(to);
```

Mit zwei `MATCH`-Befehlen wird jeweils das Regest und der Registereintrag aufgerufen und mit dem `CREATE`-Befehl eine `PLACE_IN`-Kante zwischen den beiden Knoten angelegt, die als Attribute den Inhalt der Spalten `name1` und `name2` erhält. Analog werden die Verknüpfungen zwischen Regestenknoten und Personenknoten angelegt:

```
// PERSON_IN-Kanten für Person erstellen
LOAD CSV WITH HEADERS FROM "https://docs.google.com/spreadsheets/d/12T-RD1Ct4aAUMN"
AS line
MATCH (from:IndexPerson {registerId:line.ID}),
(to:Regesta {regnum:line.regnum2})
CREATE (from)-[:PERSON_IN {regnum:line.regnum, name1:line.name1,
    name2:line.name2}]->(to);
```

2.5 Exkurs 2: Die Hierarchie des Registers der Regesten Kaiser Friedrichs III.

In anderen Registern der Regesta Imperii, wie beispielsweise den Regesten Kaiser Friedrichs III., sind teilweise fünf oder mehr Hierarchiestufen vorhanden, die jeweils auch Entitäten repräsentieren.

In diesen Fällen müssen die Hierarchien auch in der Graphdatenbank abgebildet werden, was durch zusätzliche Verweise auf die ggf. vorhandenen

- Bamberg (Bayern), Stadt
- Bürger und Einwohner s. Herr
 - Bischof 178
 - – Anton (von Rotenhan) (1431-1459) **29, 35, 45, 46, 177, 216, 238, 564**
 - Domkapitel **35**
 - – Domdekan s. Rotenhan
 - – – Gericht des Domdekans 29
 - – Dompropst 375, **376, 377**
 - – – Georg von Schaumberg **45, 46, 410, 411-413, 426, 430**
 - – – Lehengericht des Dompropstes 410-413, 426, 430
 - – – – Lehenrichter s. Truchseß
 - Tag 20, 30, 31, 102

Figure 11: Ausschnitt aus dem Register des Heftes 19 der Regesten Kaiser Friedrichs III.

übergeordneten Registereinträge möglich wird.

nodeID	xmlID	topnodeID	name1	name3
1	A00000001		Aa, Johann von	Aa, Johann von ~
2	A00000002	1	Sophie von ~, To	Aa, Johann von ~ // Sophie von ~, Tochter Johanns, Bürgerin zu Köln
3	A00000003		Aach	Aach (Fluß durch Aach, n. Singen, Baden-Württemberg)
4	A00000004		Aach	Aach (n. Singen, Baden-Württemberg), Stadt
5	A00000005		Aache s. Aacher	Aache s. Aachen
6	A00000006		Aachen	Aachen (Aache; Nordrhein-Westfalen), Stadt
7	A00000007	6	Einwohner und B	Aachen (Aache; Nordrhein-Westfalen), Stadt // Einwohner und Bürger ; s. Col
8	A00000008	6	Fischmarkt	Aachen (Aache; Nordrhein-Westfalen), Stadt // Fischmarkt (Parwisch)
9	A00000009	6	Gerichte	Aachen (Aache; Nordrhein-Westfalen), Stadt // Gerichte
10	A00000010	6	Kurgericht	Aachen (Aache; Nordrhein-Westfalen), Stadt // Gerichte // Kurgericht
11	A00000011	6	Schöffensteinuhl, k	Aachen (Aache; Nordrhein-Westfalen), Stadt // Gerichte // Schöffensteinuhl, kgl.
12	A00000012	6	Richter und Schö	Aachen (Aache; Nordrhein-Westfalen), Stadt // Gerichte // Schöffensteinuhl, kgl.
13	A00000013	6	Schöffen	Aachen (Aache; Nordrhein-Westfalen), Stadt // Gerichte // Schöffensteinuhl, kgl.
14	A00000014	6	„Grafschaften“	Aachen (Aache; Nordrhein-Westfalen), Stadt // „Grafschaften“
15	A00000015	6	„Grasgebot“	Aachen (Aache; Nordrhein-Westfalen), Stadt // „Grasgebot“
17	A00000017	6	Meierei	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei
19	A00000019	6	Brothaus	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // Brothaus
21	A00000021	6	Gewandhaus	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // Gewandhaus
22	A00000022	6	Grashaus	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // Grashaus
23	A00000023	6	Plankenhaus	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // Plankenhaus
24	A00000024	6	Tuchhaus	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // Tuchhaus
25	A00000025	6	Haus zum Haner	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // Haus zum Haner (zu
26	A00000026	6	zo der Geiss	Aachen (Aache; Nordrhein-Westfalen), Stadt // Meierei // zo der Geiss
27	A00000027	6	Rentmeister	Aachen (Aache; Nordrhein-Westfalen), Stadt // Rentmeister
29	A00000029	6	„Stadtbücher“	Aachen (Aache; Nordrhein-Westfalen), Stadt // „Stadtbücher“
30	A00000030	6	Vogtei	Aachen (Aache; Nordrhein-Westfalen), Stadt // Vogtei

Figure 12: Ausschnitt der Entitätentabelle des Registers der Regesten Friedrichs III.

Im Tabellenausschnitt wird jedem Registereintrag in der ersten Spalte eine **nodeID** als eindeutige Kennung zugewiesen. Bei Registereinträgen, die kein Hauptlemma sind, enthält die dritte Spalte **topnodeID** den Verweis auf die eindeutige Kennung **nodeID** des übergeordneten Eintrages. Beim Import in die Graphdatenbank wird diese Hierarchie über **CHILD_OF**-Kanten abgebildet, die vom untergeordneten Eintrag auf das übergeordnete Lemma verweisen. Damit ist die komplette Registerhierarchie im Graphen abgebildet. In der Spalte **name1** ist das Lemma angegeben. In der Spalte **name3** ist zusätzlich zum Lemma noch der gesamte Pfad vom Hauptlemma bis zum Registereintrag, jeweils mit Doppelslashes (//) getrennt. Bei tiefer gestaffelten Registern ist teilweise ohne Kenntnis der übergeordneten Einträge eine eindeutige Identifizierung eines Eintrages nicht möglich. So wird in Zeile 17 der o.a. Abbildung allein mit der Angabe aus der Spalte **name1** nicht klar, um welche **Meierei** es sich handelt. Mit dem kompletten Pfad des Registereintrages in der Spalte **name3** wird dagegen deutlich, dass die Aachener **Meierei** gemeint ist.

2.6 Auswertungsperspektiven

2.6.1 Personennetzwerke in den Registern

2.6.1.1 Graf Robert II. von Flandern in seinem Netzwerk

Nach dem Import können nun die Online-Regesten und die Informationen aus den Registern der Regesten Kaiser Heinrichs IV. in einer Graphdatenbank aus einer Vernetzungsperspektive abgefragt werden.⁵

Ausgangspunkt ist der Registereintrag von Graf Robert II. von Flandern. Diesen Knoten finden wir mit folgendem Query.

```
// Robert II. von Flandern
MATCH (n:IndexPerson) WHERE n.registerId = 'H4P01822'
RETURN *;
```

Mit einem Doppelklick auf den `IndexPerson`-Knoten öffnen sich alle `Regesta`-Knoten, in denen Robert genannt ist. Klickt man nun wiederum alle Regesten-knoten doppelt an, sieht man alle Personen und Orte, mit denen Robert gemeinsam in den Regesten genannt ist.

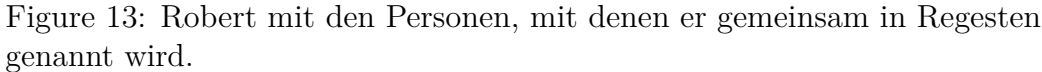
Dies kann auch in einem cypher-Query zusammengefasst werden.

```
// Robert II. von Flandern mit Netzwerk
MATCH (n:IndexPerson)-[:PERSON_IN]->
(r:Regesta)<-[:PERSON_IN]-
(m:IndexPerson)
WHERE n.registerId = 'H4P01822'
RETURN *;
```

In der folgenden Abb. wird das Ergebnis dargestellt.

Hier wird der `MATCH`-Befehl um einen Pfad über `PERSON_IN`-Kanten zu `Regesta`-Knoten ergänzt, von denen jeweils eine `PERSON_IN`-Kante zu den anderen, in den Regesten genannten `IndexPerson`-Knoten führt.

⁵Die nun folgenden Abfragen sind zum Teil einer Präsentation entnommen, die für die Summerschool der Digitalen Akademie im Rahmen des Mainzed entwickelt wurden. Die Präsentation findet sich unter der URL <https://digitale-methodik.adwmainz.net/mod5/5c/slides/graphentechnologien/RI.html>.



Nimmt man noch eine weitere Ebene hinzu, wächst die Ergebnismenge stark an. Der folgende Query kann daher je nach Rechnerleistung etwas länger dauern.

```
// Robert II. von Flandern mit Netzwerk und Herrscherhandeln (viel)
MATCH
(n1:IndexPerson)-[:PERSON_IN]->(r1:Regesta)<-[:PERSON_IN]-
(n2:IndexPerson)-[:PERSON_IN]->(r2:Regesta)<-[:PERSON_IN]-
(n3:IndexPerson)
WHERE n1.registerId = 'H4P01822'
RETURN *;
```

2.6.1.2 Graf Robert II. von Flandern und Herzog Heinrich von Niederlothringen

In der Graphdatenbank ist es über die Exploration der Beziehungen einer Person hinaus möglich, explizit die Verbindungen von zwei Personen abzufragen. In unserem nächsten Beispiel suchen wir jene Regesten, in denen Graf Robert II. von Flandern und Herzog Heinrich von Niederlothringen gemeinsam genannt sind.

```
// Robert II. von Flandern und Herzog Heinrich von Niederlothringen mit Netzwerk
MATCH
(n:IndexPerson)-[:PERSON_IN]->
(r:Regesta)<-[:PERSON_IN]-(m:IndexPerson)
WHERE n.registerId = 'H4P01822'
AND m.registerId = 'H4P00926'
RETURN *;
```

Es zeigt sich, dass Robert und Heinrich in einem Regest gemeinsam genannt sind.

Und dieses Regest berichtet von der Unterwerfung Roberts unter und Heinrich IV.⁶

Heinrich feiert das Fest der Apostel, wobei sich Graf Robert von Flandern im Beisein mehrerer Fürsten unterwirft, namentlich der Erzbischöfe Friedrich von Köln und Bruno von Trier, der Bischöfe

⁶Vgl. RI III,2,3 n. 1487.

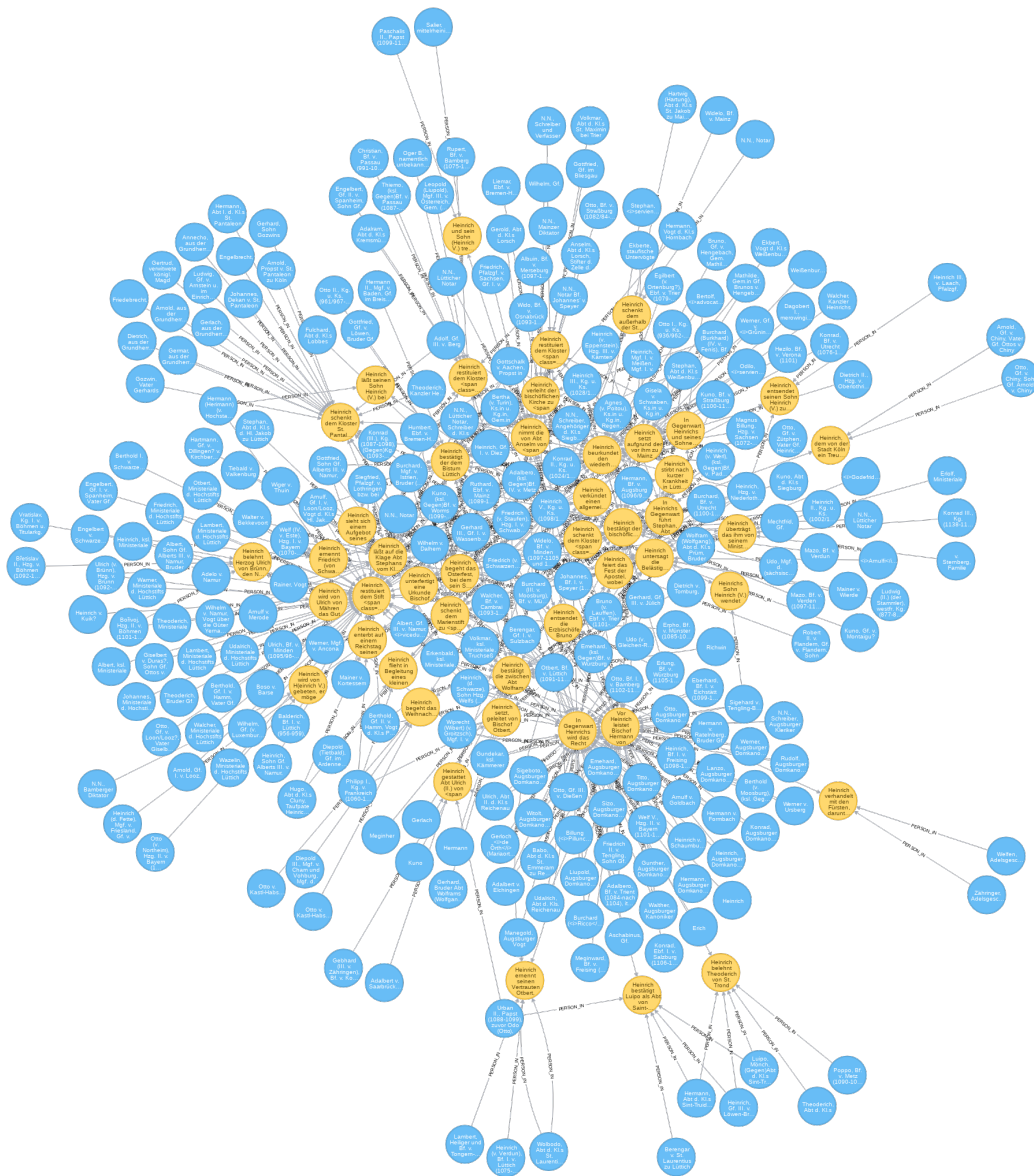


Figure 14: Robert mit Personen, die wiederum mit Personen gemeinsam in Regesten genannt sind.



Figure 15: Robert und Heinrich sind in einem Regest gemeinsam genannt.

Otbert von Lüttich, Burchard von Münster, Burchard von Utrecht,
Herzog Heinrich von Niederlothringen sowie mehrerer Grafen.

Möglicherweise haben beide aber gemeinsame Bekannte, also Personen mit denen sowohl Heinrich als auch Robert in unterschiedlichen Regesten gemeinsam genannt sind. Hierfür wird der cypher-Query um eine Ebene erweitert.

```
// Robert und Heinrich mit allen gemeinsamen Personen und Regesten
MATCH (n1:IndexPerson)
-[:PERSON_IN]->(r1:Regesta)<-[:PERSON_IN]-
(n2:IndexPerson)-[:PERSON_IN]->(r2:Regesta)
<-[:PERSON_IN]-(n3:IndexPerson)
WHERE n1.registerId = 'H4P00926'
AND n3.registerId = 'H4P01822'
RETURN *;
```

Ein erster Blick auf das Ergebnis zeigt, dass Heinrich allgemein besser vernetzt ist. Für die weitere Analyse ihres Verhältnisses ist nun die Lektüre der angegebenen Regesten notwendig. Hierfür lässt sich das Ergebnis noch etwas weiter aufbereiten, indem die zwischen den Personen liegenden Regesten in KNOWS-Kanten umgewandelt werden, die als zusätzliche Information die Angaben zu den Regesten enthalten.

```
// Rausrechnen der dazwischenliegenden Knoten
MATCH
(startPerson:IndexPerson)-[:PERSON_IN]->
(regest:Regesta)<-[:PERSON_IN]-(endPerson:IndexPerson)
WHERE startPerson.registerId in ['H4P01822', 'H4P00926']
WITH startPerson, endPerson, count(regest) as anzahl,
collect(regest.ident) as idents
CALL apoc.create.vRelationship(startPerson, "KNOWS",
```

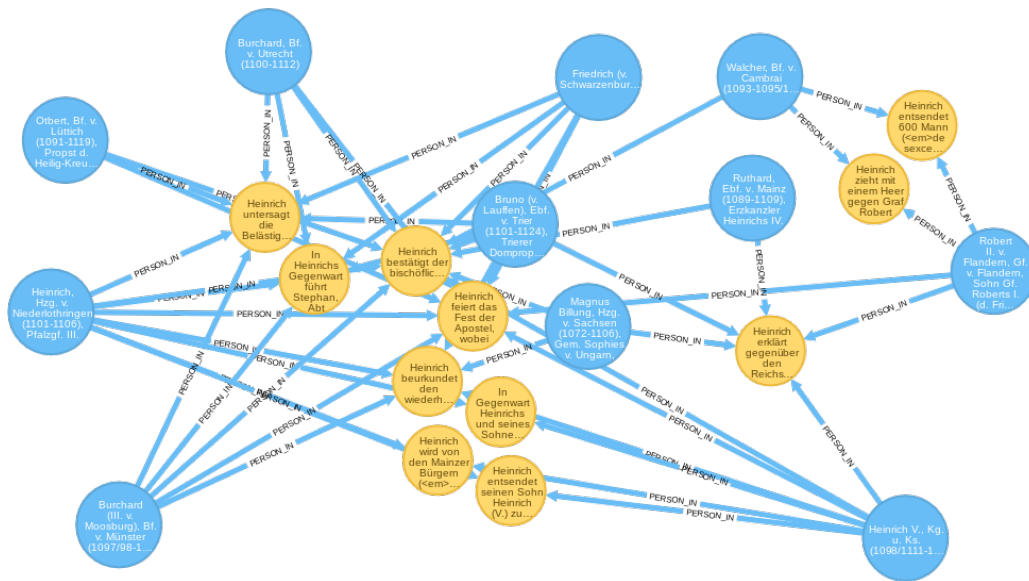


Figure 16: Robert und Heinrich mit den gemeinsamen Bekanntschaften.

```
{anzahl:anzahl, regesten:idents}, endPerson) YIELD rel
RETURN startPerson, endPerson, rel;
```

In der Abbildung sind die zwei Ego-Netzwerke von Heinrich (links) und Robert (rechts) mit den dazwischen liegenden gemeinsamen Bekanntschaften dargestellt. Es zeigt sich, dass Heinrich stärker sowohl mit Geistlichen als auch Weltlichen vernetzt war, während Robert insgesamt weniger Kontakte aber mit einem Schwerpunkt in der Geistlichkeit hatte.

Für den Historiker ist aber vor allem interessant, was in den Regesten steht, die Robert und Heinrich über die Mittelsmänner verbinden. Hierfür wird der cypher-Query angepasst und sowohl Personen als auch die Regestentexte ausgegeben.

```
// Liste der Regesten als Ergebnis
MATCH
(startPerson:IndexPerson)-[:PERSON_IN]->
(regest1:Regesta)-[:PERSON_IN]-(middlePerson:IndexPerson)
-[:PERSON_IN]->(regest2:Regesta)
<-[:PERSON_IN]-(endPerson:IndexPerson)
WHERE startPerson.registerId in ['H4P00926']
```



```

AND endPerson.registerId in ['H4P01822']
RETURN DISTINCT startPerson.name1,
regist1.ident, regist1.text,
middlePerson.name1, regist2.ident,
regist2.text, endPerson.name1;

```

In der folgenden Abbildung wird ein Ausschnitt der Ergebnistabelle gezeigt. In der ersten Spalte der Tabelle finden sich Robert, anschließend die Angaben zum Regest, mit dem er mit der mittleren Person (middlePerson.name1) verknüpft ist. Dem folgen schließlich die Angaben zum Regest, mit den die mittlere Person mit Robert in der letzten Spalte verbunden ist. Die Tabelle bietet einen Überblick zur Überlieferungssituation aus der Perspektive der Regesta Imperii.

startPerson.name1	regist1.ident	regist1.text	middlePerson.name1	regist2.ident	regist2.text	endPerson.name1
"Heinrich, Hzg. v. Niederlothringen (1101-1106), Pfalzgf. III. v. Lothringen, Gf. v. Limburg"	"RI III,2,3 n. 1489"	"Heinrich bestätigt der bischöflichen Kirche zu Bamberg unter Bischof Otto zum Gedenken an seine Großeltern, Kaiser Konrad (II.) und Kaiserin Gisela, seine Eltern, Kaiser Heinrich (III.) und Agnes, seine Gemahlin, die Kaiserin Bertha, sowie besonders an seinen Verwandten, Kaiser Heinrich (II.), den Gründer der Bamberger Kirche, aufgrund der Intervention seines Sohnes, König Heinrichs V., der Erzbischöfe Friedrich von Köln, Bruno von Trier und Humbert von Bremen, der Bischöfe Otbert von	"Bruno (v. Lauffen), Ebf. v. Trier (1101- 1124), Trierer Dompropst"	"RI III,2,3 n. 1487"	"Heinrich feiert das Fest der Apostel, wobei sich Graf Robert von Flandern im Beisein mehrerer Fürsten unterwirft, namentlich der Erzbischöfe Friedrich von Köln und Bruno von Trier, der Bischöfe Otbert von Lüttich, Burchard von	"Robert II. v. Flandern, Gf. v. Flandern, Sohn Gf. Roberts I. (d. Friesen) v. Flandern, Neffe Gf. Balduins VI. v. Flandern"

Figure 18: Robert und Heinrich mit den gemeinsamen Bekanntschaften.

2.6.2 Herrscherhandeln ausgezählt

Wie bereits oben erwähnt wurde in einem ersten Test jeweils das erste Verb des Regestentextes extrahiert, lemmatisiert und in die Graphdatenbank eingespielt. Im folgenden werden nun einige cypher-Querys vorgestellt, die dies beispielhaft auswerten.

```

// Herrscherhandeln ausgezählt
MATCH (n:Lemma)-[h:ACTION]-(m:Regesta)
RETURN n.lemma, count(h) as ANZAHL ORDER BY ANZAHL desc LIMIT 10;

```

n.lemma	ANZAHL
werden	145
schenken	133
bestätigen	109
begehen	95
verleihen	48
ernennen	36
nehmen	35
treffen	34
empfangen	29
erhalten	26

Die Ergebnisliste zeigt gleich die Einschränkungen, da das Hilfsverb *werden* aus dem textuellen Zusammenhang gerissen ist. Andererseits ergeben sich aber auch interessante Erkenntnisse zur Häufigkeitsverteilung von Herrscherhandeln in Regestentexten. Die Anwendung des Verfahrens auf Regestentexte ist dabei auf der einen Seite positiv, da bei der Erstellung der Regesten sehr stark auf formale Kriterien geachtet wird und so die Zusammenhänge gut zu erfassen sind. Auf der anderen Seite ist die Auswertung aber wiederum einen weiteren Schritt von der ursprünglichen Quelle entfernt.

2.6.3 Herrscherhandeln pro Ausstellungsort ausgezählt

Im folgenden Query kommt eine räumliche Komponente zur Abfrage hinzu, da das Lemma hier jeweils abhängig vom Ausstellungsort der Urkunde abgefragt wird.

```
// Herrscherhandeln pro Ausstellungsort
MATCH (n:Lemma)-[:ACTION]-(:Regesta)-[:PLACE_OF_ISSUE]->(p:Place)
WHERE p.normalizedGerman IS NOT NULL
RETURN p.normalizedGerman, n.lemma, count(h) as ANZAHL ORDER BY ANZAHL desc LIMIT
```

p.normalizedGerman	n.lemma	ANZAHL
Mainz	begehen	15
Mainz	schenken	14
Goslar	schenken	13

p.normalizedGerman	n.lemma	ANZAHL
Rom	werden	12
Regensburg	schenken	12
Goslar	begehen	11
Speyer	schenken	10
Worms	begehen	8
Regensburg	bestätigen	7
Regensburg	werden	7

In der ersten Spalte befindet sich der Ortsname, der aus der Property **normalizedGerman** des **Place**-Knotens stammt. In der zweiten Spalte wird das Lemma angegeben und in der dritten Spalte schließlich die Anzahl der jeweiligen Regesten. Interessant wäre hier auch noch die Ergänzung der zeitlichen Dimension, mit der dann der zeitliche Verlauf in die Auswertung miteinbezogen werden könnte.

2.6.4 Herrscherhandeln und Anwesenheit

Im nächsten Beispiel werden in einem Regest genannten Personen in die Auswertung des Herrscherhandelns mit einbezogen.

```
MATCH (p:IndexPerson)-[:PERSON_IN]-(r:Regesta)-[:ACTION]-(l:Lemma)
RETURN p.name1, l.lemma, count(1) AS Anzahl ORDER BY p.name1, Anzahl DESC;
```

p.name1	l.lemma	Anzahl
...
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	schenken	21
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	bestätigen	9
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	verleihen	4

p.name1	l.lemma	Anzahl
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	erlassen	2
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	übertragen	2
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	ermäßigen	2
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	gestatten	2
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	vollziehen	1
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	nehmen	1
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	mindern	1
Adalbero, Metzger Domkanoniker, Kanzler Heinrichs IV., Kanzler (Gegen)Kg. Rudolfs v. Rheinfelden	setzen	1
...

Die Ergebnistabelle zeigt den Abschnitt zu Adalbero, einem Metzger Domkanoniker mit der Häufigkeit des jeweiligen Herrscherhandeln-Lemmas.

2.6.5 Regesten 200 km rund um Augsburg

Mit dem folgenden Query werden für den Umkreis von 200 km rund um Augsburg alle Regesten aufgerufen.

// Entfernungen von Orten berechnen lassen

```

MATCH (n:Place)
WHERE n.normalizedGerman = 'Augsburg'
WITH n.latLong as point
MATCH (r:Regesta)
WHERE distance(r.latLong, point) < 200000
AND r.placeOfIssue IS NOT NULL
AND r.placeOfIssue <> 'Augsburg'
RETURN r.ident, r.placeOfIssue,
distance(r.latLong, point) AS Entfernung
ORDER BY Entfernung;

```

Solche Queries lassen sich auch mit zeitlichen Abfragen kombinieren und bieten sehr flexible Abfragemöglichkeiten.

2.6.6 Welche Literatur wird am meisten zitiert

Beim Import der Regesten in die Graphdatenbank werden die mit dem RI-Opac verlinkten Literaturtitel als eigenständige **Reference**-Knoten angelegt und jeweils mit dem **Regesta**-Knoten verknüpft. Diese Verknüpfung wird mit dem folgenden Query abgefragt, ausgezählt und aufgelistet.

```

MATCH (n:Reference)-[r:REFERENCES]-(m:Regesta)
RETURN n.title, count(r) AS Anzahl
ORDER BY Anzahl DESC LIMIT 10;

```

n.title	ANZAHL
Stumpf	215
Böhmer	201
Ldl	101
Jaffé	60
Schmale	56
Buchholz	51
Scheffer-Boichorst	50
Wauters	39
Dobenecker	33
Remling	28

Mit diesen Daten lassen sich Zitationsnetzwerke in den Regesten darstellen mit denen Regesten gefunden werden können, die auf Grund der gemeinsam zitierten Literatur die gleichen inhaltlichen Schwerpunkte aufweisen können.

2.6.7 Der Import zusammengefasst

Den komplette cypher-Code für die Erstellung der Graphdatenbank ist zusammengefasst über ein Textdatei abrufbar. Es ist zu empfehlen, die aktuelle Version von neo4j-Desktop zu installieren, eine Graphdatenbank anzulegen und in der Graphdatenbank die APOC-Bibliothek zu installieren. Inzwischen ist es möglich, in der Befehlszeile des neo4j-Browsers auch mehrere Befehle nacheinander ausführen zu lassen. Alternativ kann man nach dem Start der Graphdatenbank im Reiter **Terminal** mit dem Befehl `bin/cypher-shell` die cypher-shell aufgerufen werden. In diese Shell werden dann alle Befehl gemeinsam reinkopiert und ausgeführt. Alternativ zur Installation von neo4j kann auch auf den Internetseiten von neo4j seine Sandbox erstellt werden.

2.7 Zusammenfassung

In diesem Kapitel wurden die Schritte zum Import der Regesten Kaiser Heinrichs IV. in die Graphdatenbank neo4j erläutert sowie verschiedene Auswertungsbeispiele vorgestellt.