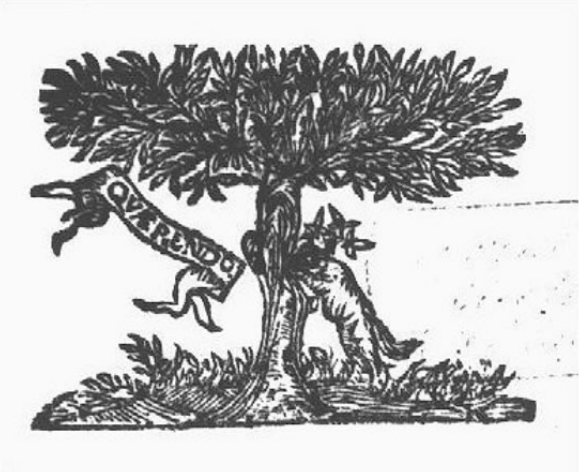
Met SPARQL zoeken in de STCN

Versie 0.1



Peter Boot, Marieke van Delft, Juliette Lonij, Els Stronks

Koninklijke Bibliotheek, Huygens Instituut voor Nederlandse Geschiedenis, CLARIN en Universiteit Utrecht.

2015

Het drukkersmerk op het omslag met de tekst ‘QVAERENDO’ is afkomstig van www.arkyves.org. Dit merk is gebruikt door verschillende drukkers in Rotterdam en Amsterdam in de periode 1654-1680.

0. Inleiding

STCN
De Short-Title Catalogue Netherlands bevat beschrijvingen van alle bewaard gebleven boeken die vóór 1801 in Nederland of in het Nederlands gedrukt zijn (met uitzondering van Vlaanderen; op de webtoegang worden wel titels uit de STCV doorzocht). Daarmee biedt deze bibliografie een schat aan gegevens voor het onderzoek naar cultuur en maatschappij in de vroegmoderne tijd.

Zoeken in de STCN RDF-dataset
Afgelopen najaar zijn de beschrijvingen van de STCN ook als RDF-dataset beschikbaar gemaakt. Het doel hiervan was om in de STCN data veel complexere zoekacties mogelijk te maken dan in de bestaande STCN-webinterface (http://www.stcn.nl). Er is nu een nieuwe zoektool (zie: http://openvirtuoso.kbresearch.nl/sparql) waarmee meer kan. Zo kunnen data van verschijnen gecombineerd worden met gegevens over genre, om te zien welk genre wanneer populair was. En kan met één zoekactie getoond worden hoe formaten van boeken voor jongeren zich ontwikkelden in de loop der tijd. Of wanneer bepaalde steden geïllustreerde boeken produceerden.

SPARQL
In die nieuwe zoektool moeten zoekcommando's in SPARQL gegeven worden, en dat vergt enige oefening. Deze handleiding biedt een korte introductie in SPARQL, en voorbeelden van zoekvragen met uitleg die zelf zoeken in het SPARQL interface mogelijk maken.
De kracht van het zoeken in SPARQL ligt voornamelijk in de mogelijkheden die deze omgeving biedt voor het sorteren, combineren, berekenen en groeperen van de resultaten van een bepaald zoekresultaat. Waar in de oude interface veelvuldig handmatig geteld moest worden om bepaalde verschijnselen te onderzoeken, kan dat met SPARQL eenvoudiger. Ook kan zonder veel moeite allerlei uitvoer gemaakt worden, terwijl de webversie van de STCN slechts enkele formaten kent.

Opzet van de handleiding
We beginnen met een heel korte uitleg over de inhoud van de STCN, gevolgd door een korte uitleg over RDF en gaan dan naar een uitleg over de basale syntax van SPARQL. Daarna laten we aan de hand van concrete zoekvragen uit lopend onderzoek zien hoe je met die syntax gegevens uit de STCN kunt groeperen, ordenen en tellen. Ook laten we zien hoe je die gegevens in verschillende formaten (.cvs, .xsl) kunt downloaden, om ze vervolgens met een programma als Excel in grafiekvorm te kunnen tonen. In schema:

1. inhoud STCN
2. STCN in RDF
3. basale SPARQL syntax
4. voorbeelden zoekvragen + exportmogelijkheden

Den Haag / Utrecht, april 2015

1. Opbouw STCN

De STCN wordt gemaakt met het GGC – het Gemeenschappelijk Geautomatiseerd Catalogiseer systeem van OCLC. De voertaal van de catalogus is Engels. Van de boeken worden elementen beschreven die normalerwijze in een catalogus te vinden zijn (auteur, titel, impressum). Titels worden verkort opgenomen (*Short-Title Catalogue!).* Speciale aandacht wordt gegeven aan karakteristieke elementen die van belang zijn bij het beschrijven van oude drukken zoals de collatieformule en de fingerprint. De velden worden op gestructureerde wijze gevuld. In het online handboek van de STCN is per kenmerkcode (in de tabel: Pica 3) precies beschreven hoe de velden ingevuld worden (zie: http://www.kb.nl/kbhtml/stcnhandleiding/stcn.html)

Dat wil zeggen dat in de web interface van de STCN gezocht kan worden op:
- *type publicatie (monografie, tijdschrift)*
- *jaar van publicatie*
- *typografische kenmerk: hierin worden specifieke kenmerken van het boek aangegeven*
    - a: illustratie op titelpagina
    - b: illustraties buiten collatie
    - c: andere illustraties
    - d: boekenlijst van auteur
    - e: fondslijst
    - f: assortimentslijst
    - g: boekenlijst diversen
    - h: drukkersmerk
    - i: lettertype romein
    - j: lettertype gotisch
    - k: lettertype cursief
    - l: lettertype civilite
    - m: lettertype Grieks
    - n: lettertype Hebreeuws
    - o: lettertype Arabisch
    - p: lettertype Armeens
    - q: muzieknotaties
    - r: lettertype cyrillisch
    - s: lettertypen overige
    - v: bedrukte omslag
    - w: gegraveerde titelpagina
    - x: typografische titelpagina
    - y: geen titelblad
    - z: titelblad in meer kleuren
    - 3: lijst van intekenaren/oproep tot intekening
    - 4: prijsopgave
- *taal van de uitgave. De meest voorkomende talen zijn:*
    - ned = Nederlands
    - fra = Frans
    - eng = Engels
    - grk = Grieks
    - fri = Fries

spa = Spaans
dui = Duits
lat = Latijn
heb = Hebreeuws
mis = Meer teksten in verschillende talen
mul = Eén tekst in verschillende talen

- *land van uitgave. De landcode bestaat uit twee letters. De meest voorkomende landen zijn*:
  nl = Nederland
  be= België
  ch = Zwitserland
  de= Duitsland
  fr= Frankrijk
  gb= Engeland
  id= Indonesië
  it= Italië
- *vingerafdruk*
  Hiermee kunnen verschillende edities onderscheiden worden door het zetsel te onderscheiden. Voor een uitleg van de vingerafdruk zie het eerder genoemde handboek of de uitleg over de pagina over de zoekresultaten in de STCN (http://www.kb.nl/organisatie/onderzoek-expertise/informatie-infrastructuur-diensten-voor-bibliotheken/short-title-catalogue-netherlands-stcn/uitleg-resultaten-stcn)
- *auteurs en medewerkers:*
  de auteurs worden genormaliseerd opgenomen. Alle auteurs hebben een relatie met de auteursdatabase waarin extra gegevens opgenomen zijn zoals leefjaren, beroep, aanduiding of het om een vrouw gaat. LET OP: deze gegevens zijn niet altijd opgenomen!
- *titels: de titels zijn verkort.*
  Bij standaardwerken (bijv. Reinaert, Bijbel) zijn ze eventueel voorzien van een genormaliseerde titel (zie eerder genoemd STCN-handboek (http://www.kb.nl/kbhtml/stcnhandleiding/stcn.html))
- *editievermelding*
- *impressum (Plaats + drukker/uitgever (genormaliseerd)*
- *drukkers:*
  drukkers/uitgevers/boekhandelaren worden ook nog eens apart opgenomen. Alle drukkers hebben een relatie met een drukkersdatabase waarin extra gegevens opgenomen zijn zoals variante namen, leefjaren, jaren van werkzaamheid, adressen, uithangborden. De jaren van werkzaamheid zijn exact genoteerd zodat ook duidelijk is als een drukker bijvoorbeeld een aantal jaren niet actief was.
- *collatieformule: van elk boek is de opbouw van het boek te zien in de collatieformule.*
  Het aantal pagina's of vellen is NIET in de STCN-beschrijving opgenomen.
- *formaat: van ieder boek wordt het bibliografische formaat gegeven, dus niet in centimers*
  Broadsheet
  1º (Plano)
  2º (Folio)
  4º (Quarto)
  8º (Octavo)
  12º (Duodecimo)
  16º (16mo)

18º (18mo)
24º (24mo)
32º (32mo)
48º (48mo)
64º (64mo)
Oblong (breder dan hoog)
Agenda (±3 maal hoger dan breed)

- *trefwoorden:*
  de boeken zijn ontsloten met geografische trefwoorden en onderwerpstrefwoorden. De onderwerpstrefwoorden zijn ontleend aan de Nederlandse Basis Classificatie en zijn op hoofdonderwerp gegeven. Deze trefwoorden worden ontleend aan een gerelateerde database; er is op Nederlandstalig en Engelstalig trefwoord te zoeken
- *instellingen en exemplaren;*
  De instellingen zijn aangeduid met een letter; vervolgens wordt het signatuur van een boek in de desbetreffende instelling gegeven
- *verwijzing naar fulltekst*

Voor meer informatie zie:
- Handboek van de STCN: http://www.kb.nl/kbhtml/stcnhandleiding/stcn.html
- Uitleg van de resultaten van het zoeken in de STCN: www.kb.nl/organisatie/onderzoek-expertise/informatie-infrastructuur-diensten-voor-bibliotheken/short-title-catalogue-netherlands-stcn/uitleg-resultaten-stcn

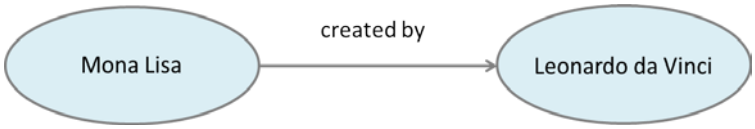
NB1: Niet alle elementen zijn meegenomen in de RDF versie (bijv. land van uitgave, vingerafdruk). Voor een volledig overzicht van de beschikbare velden in de RDF-versie, zie de bijlagen.
NB2: De RDF zoals nu voorligt heeft nog een enkele onvolkomenheden die bepaalde specifieke zoekvragen onmogelijk maakt door de keuze voor bepaalde standaarden (SKOS, Dublin Core), zoals naar de plaats van werkzaamheid van een bepaalde drukker, zijn beroep, zijn specifieke adres of het uithangbord dat hij gebruikte. Deze gegevens plus een algemene annotatie zijn wel meegenomen in de conversie van de drukkersthesaurus maar allemaal in hetzelfde veld editorialNote gezet zonder nadere specificatie (zie hieronder: de gegevens in de grijs gearceerde velden zijn samengevoegd in het veld editorialNote)

033A	$nHalma, François
033C	$aLeeuwarden$b1710-1720
033D	$aprinter$b1711, 1713
033D	$aprinter to the Provincial States$b1710-1719
033D	$auniversity printer$b1716
047A	$aUtrecht 1680-1699, Amsterdam 1700-1711, Franeker 1701-1717, Leeuwarden 1710-1720

2. STCN in RDF

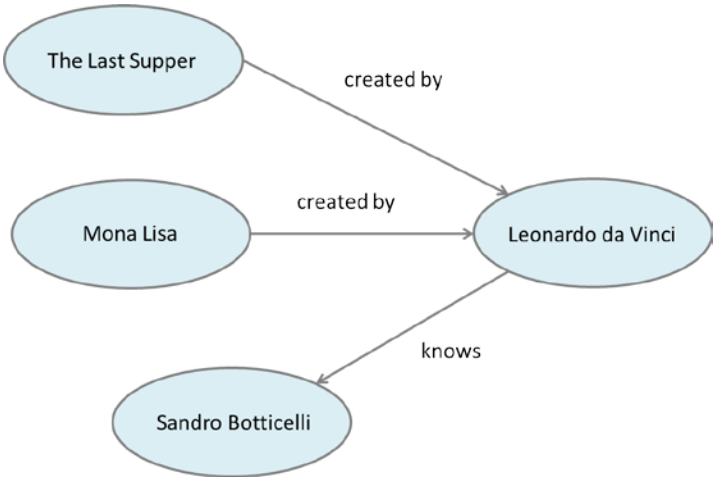
Om het mogelijk te maken de STCN data (gemakkelijker) met complexe(re) queries te bevragen, heeft de KB een voorlopige, experimentele versie van de STCN dataset ontwikkeld in RDF-formaat.

Resource Description Framework (RDF)
Het Resource Description Framework (RDF) is een zeer eenvoudig datamode dat *resources* (onderliggende data) beschrijft in termen van *triples*. Een resource is in principe iedere groep gegevens (ieder 'ding') dat men met een naam zou willen aanduiden. Voorbeelden van resources zijn o.a. mensen, organisaties, webpagina's, foto's, plaatsen, gebouwen, gebeurtenissen etc. Triples zijn uitspraken met behulp van die *resources* opgebouwd die steeds uit drie delen bestaan: een *subject*, een *predicate* en een *object*. Het feit dat Leonardo da Vinci de maker is van de Mona Lisa, ziet er in RDF bijvoorbeeld uit als:

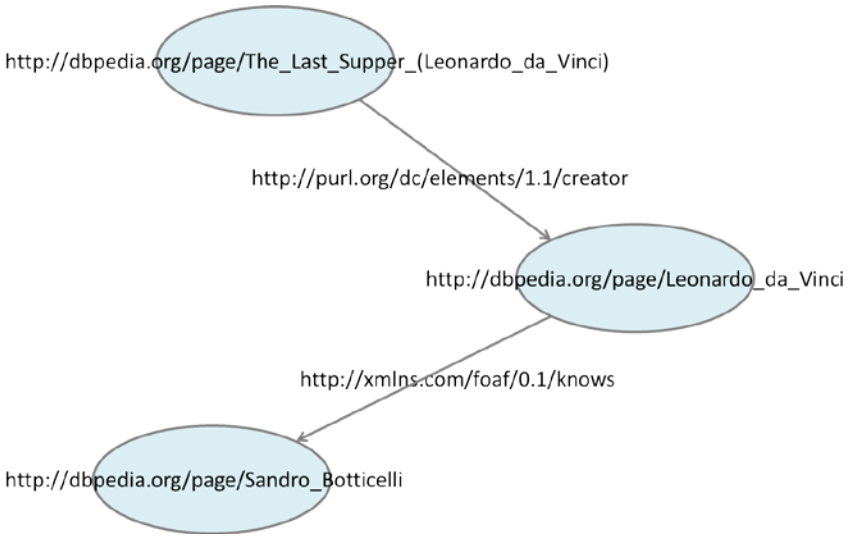


→ hier is de resource <Mona Lisa> het subject dat beschreven wordt
→ de resource <Leonardo da Vinci> het object
→ en <created by> het predicate dat de aard van de relatie tussen beide resources uitdrukt.

Ook Leonardo da Vinci kan op zijn beurt weer beschreven worden door middel van één of meerdere triples. Zo ontstaat een netwerk van via triples met elkaar verbonden resources:



Gebruik van URI's in RDF
In plaats van gewone namen, zoals Leonardo da Vinci, gebruikt RDF zgn. URIs (Unique Resource Identifiers), voor het aanduiden van zowel resources als relaties. Een URI is een unieke naam voor een resource – vaak in de vorm van een URL – die ervoor zorgt dat er nooit ambiguïteit kan optreden. Twee personen kunnen bijvoorbeeld dezelfde naam hebben, maar hun URIs zullen altijd verschillen. Wanneer we zowel resources als relaties aanduiden met URIs, ontstaat een RDF netwerk of RDF *graph*:



De meeste mensen zullen < http://dbpedia.org/page/Leonardo_da_Vinci> niet prettig leesbaar vinden. Daarnaast is niet voor alle resources altijd (direct) een URI voorhanden. Daarom kent RDF naast resources ook zgn. *literals*, letterlijke waarden. Deze maken het onder meer mogelijk om aan te geven dat <http://dbpedia.org/page/Leonardo_da_Vinci> de naam Leonardo da Vinci heeft.

Welke URI'S?
Een voor de hand liggende vraag is welke URIs te gebruiken bij het beschrijven van resources. Veel bekende personen, plaatsen, etc. hebben al een URI. Belangrijke verstrekkers zijn bijvoorbeeld DBpedia, Freebase en de Virtual Internet Authority File (VIAF). Rond het uitdrukken van relaties is in enkele jaren een consensus gegroeid over een aantal centrale vocabulaires, waaronder Friend of a Friend (FOAF), waarmee personen en relaties tussen personen beschreven kunnen worden, Dublin Core (DC) voor de beschrijving van documenten en Simple Knowledge Organisation System (SKOS) voor thesauri, taxonomieën en classificaties. Schema.org is een algemeen vocabulaire waarvan het gebruik, ook in de bibliotheeksector, steeds meer ingang vindt.

Prefixes
Het is gebruikelijk om voor veel voorkomende vocabulaires zgn. *prefixes* te definiëren, zodat de lange URIs niet steeds opnieuw uitgeschreven hoeven worden. Een eigenschap als de auteur van een publicatie in het Dublin Core vocabulaire kan dan bijvoorbeeld worden aangeduid met de benaming <dc:author>, in plaats van het volledige <http://purl.org/dc/elements/1.1/author>. Hierover in het volgende hoofdstuk meer.

Conversie STCN naar RDF
Voor de transformatie van de STCN dataset vanuit het oorspronkelijke Pica / Pica+ formaat naar RDF is uitgegaan van de STCN bibliografische beschrijvingen en de daarbij behorende thesauri van auteurs, drukkers en trefwoorden zoals die in de STCN aanwezig waren op 23 oktober 2014.

Voor resources in de STCN zijn allereerst URIs opgesteld: voor titels hebben deze de vorm http://data.kb.nl/catalogus/<ppn> gekregen, waarbij <ppn> staat voor het Pica production number, de unieke identificatie van objecten in de STCN; voor de thesaurusingangen voor auteurs, drukkers en onderwerpen zijn deze URIs van de vorm http://data.kb.nl/thesaurus/<ppn>. Waar mogelijk is gezocht naar bestaande URIs, bijvoorbeeld voor de aanduiding van de taal van een uitgave. Hier wordt gebruik gemaakt van de taalcodes uit de ISO 639-2 standaard, waarvoor de Library of Congress URIs uitgeeft. Relaties zijn voor gegevens uit bibliografische records zo veel mogelijk omgezet naar elementen uit het Dublin Core vocabulaire en bij de thesaurusrecords naar het voor dit type gegevens veel gebruikte SKOS vocabulaire.

Bij het uitvoeren van de conversie is er in eerste instantie voor gekozen alleen de meest relevante informatie om te zetten. Enkele gegevens uit de oorspronkelijke data, zoals vingerafdruk en signatuur, zijn hierdoor niet aanwezig in de resulterende RDF. Ook is informatie toegevoegd: bij de auteurs bijvoorbeeld zijn – waar mogelijk – links aangebracht naar VIAF. Deze zijn te vinden als waarde van de eigenschap <skos:exactMatch> (<http://www.w3.org/2004/02/skos/core#exactMatch>).

Om één en ander wat concreter te maken, volgt aan het einde van dit hoofdstuk een lijst met een aantal voorbeelden van triples uit de STCN. In bijlage 1 is een volledig overzicht opgenomen van hoe de verschillende gegevens uit de STCN zijn omgezet naar RDF voor de vier typen beschrijvingen, resp. titels, auteurs, drukkers en trefwoorden.

Toegang
De STCN data in RDF-formaat is vervolgens geladen in een zogenaamde triplestore, een database specifiek ontwikkeld voor het opslaan en bevragen van RDF-triples. Deze triplestore is via het web toegankelijk op het adres http://openvirtuoso.kbresearch.nl/sparql. Dit adres biedt in de eerste plaats een grafische user interface en is daarnaast vanuit scripts of vanaf de *command line* te benaderen als *endpoint* (dit laatste type gebruik valt buiten het bestek van deze handleiding).

De STCN triples vormen samen een netwerk of *graph* met de naam http://data.kb.nl/stcn. Deze dient in de zoekinterface ingevoerd te worden in het veld *Default Data Set Name* als het netwerk dat doorzocht wordt. Vervolgens kan een query ingegeven worden in het veld *Query Text*. Deze query dient te worden geformuleerd in SPARQL, de voor zoektaal voor RDF. Hoe dit in zijn werk gaat, zal in het volgende hoofdstuk worden uitgelegd.

Een deel van de STCN graph

subject	predicate	object
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/terms/extent	"8°"
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/terms/extent	"A-B`SUP`8`LO`"
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/type	http://schema.org/Book
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/type	http://purl.org/dc/dcmitype/Text
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/date	"1769"
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/language	http://id.loc.gov/vocabulary/iso639-2/dut
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/publisher	http://data.kb.nl/thesaurus/137121865
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/subject	http://data.kb.nl/thesaurus/155445642
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/subject	http://data.kb.nl/thesaurus/155444883
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/subject	http://data.kb.nl/thesaurus/155446843
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/title	"Vertoog van een beroemd schryver over de vryheid der drukpers. / `IT`[By John Wilkes]. ; Translated from the English`LO`"@nl
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/terms/isPartOf	"Algemene Catalogus KB"
http://data.kb.nl/catalogus/163434107	http://krait.kb.nl/coop/tel/handbook/telterms.html#annotation	"romein"
http://data.kb.nl/catalogus/163434107	http://krait.kb.nl/coop/tel/handbook/telterms.html#annotation	"typografische titelpagina"
http://data.kb.nl/catalogus/163434107	http://purl.org/dc/elements/1.1/creator	http://data.kb.nl/thesaurus/072309008
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2004/02/skos/core#Concept
http://data.kb.nl/thesaurus/137121865	http://purl.org/dc/elements/1.1/type	"drukker"@nl
http://data.kb.nl/thesaurus/137121865	http://purl.org/dc/elements/1.1/type	"printer"@en
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#altLabel	"lijst"
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#altLabel	"Tongerloo, Kornelis van"
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#editorialNote	"Kalverstraat over de Keizers-Kroon (de) 1766-1769"
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#editorialNote	"Amsterdam 1765-1770. Opm.: tot 1769 alleen samen met zijn moeder, de wed. Kornelis (I)"@nl

http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#editorialNote	"Amsterdam 1765-1771"
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#editorialNote	"bookseller 1769"
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#inScheme	http://schemas.kb.nl/thesaurus/
http://data.kb.nl/thesaurus/137121865	http://www.w3.org/2004/02/skos/core#prefLabel	"Tongerlo, Kornelis van (II)"
http://data.kb.nl/thesaurus/155445642	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2004/02/skos/core#Concept
http://data.kb.nl/thesaurus/155445642	http://purl.org/dc/elements/1.1/type	"keyword"@en
http://data.kb.nl/thesaurus/155445642	http://purl.org/dc/elements/1.1/type	"trefwoord"@nl
http://data.kb.nl/thesaurus/155445642	http://www.w3.org/2004/02/skos/core#altLabel	"Tijdsgeschriften"
http://data.kb.nl/thesaurus/155445642	http://www.w3.org/2004/02/skos/core#altLabel	"Topical texts"
http://data.kb.nl/thesaurus/155445642	http://www.w3.org/2004/02/skos/core#inScheme	http://schemas.kb.nl/thesaurus/
http://data.kb.nl/thesaurus/155445642	http://www.w3.org/2004/02/skos/core#prefLabel	"Period documents"

We zien hier een uittreksel uit de STCN graph. We beginnen met alle triples waarvan het boek met identificatie ‘http://data.kb.nl/catalogus/163434107’ het subject is. We zien onder andere formaat, collatieformule, subject (trefwoorden) en publisher (http://data.kb.nl/thesaurus/137121865). De titel van het boek is ‘Vertoog van een beroemd schryver over de vryheid der drukpers’. Het uittreksel vervolgt met de gegevens over de drukker, Kornelis van Tongerloo. Vervolgens zien we nog wat gegevens over het trefwoord ‘http://data.kb.nl/thesaurus/155445642’, met als preferred label ‘Period Documents’.

3. Basale SPARQL syntax

SPARQL
SPARQL is een zoektaal voor RDF graphs. De formele definitie is te vinden op http://www.w3.org/TR/sparql11-overview/. SPARQL biedt ook mogelijkheden om RDF inhoud bij te werken. In deze handleiding gaan we alleen in op de mogelijkheid gegevens op te halen uit RDF graphs. De syntax van SPARQL herinnert in veel opzichten aan de Structured Query Language (SQL) die wordt gebruikt voor het doorzoeken van databases.

SPARQL queries hebben globaal de volgende vorm:

```
SELECT [iets]
WHERE [voorwaarden]
BIND [nieuwe inhoud]
VALUES [nieuwe inhoud]
ORDER BY [iets]
GROUP BY [iets]
```

In het volgende zullen van de verschillende onderdelen van zo'n query voorbeelden zien. Op de voorwaarden in de WHERE-clausule gaan we hier nog wat nader in. De algemene vorm van deze voorwaarden is:

```
subject predicaat object.
```

Dat is dus precies de vorm van een RDF triple (zie hoofdstuk 2). Elk van deze drie termen kan worden ingevuld met een vaste waarde of met een variabele. Zo selecteert

```
?subject dc:type schema:Book.
```

alle triples waarvan predicaat en object de waarde hebben van respectievelijk dc:type en sch:Book (dus: alle boeken). Door er een voorwaarde aan toe te voegen kunnen we andere gegevens ophalen of nadere selecties uitvoeren. We vragen de titel van de boeken op als volgt:

```
?subject dc:type schema:Book.
?subject dc:title ?title.
```

Termen die met een vraagteken beginnen zijn zogenaamde variabelen. Variabelen worden gebruikt om de verschillende voorwaarden aan elkaar te knopen (hier ?subject) en komen standaard terecht in het resultaat van de query. Een tweede voorwaarde kan ook het resultaat van de eerste voorwaarde inperken. Bijvoorbeeld:

```
?subject dc:type schema:Book.
?subject dc:date "1769".
```

Hier vragen we om alle boeken die zijn gepubliceerd in 1769.

Een handleiding SPARQL is beschikbaar op http://en.wikibooks.org/wiki/XQuery/SPARQL_Tutorial. Meer informatie bijvoorbeeld ook op http://www.linkeddatatools.com/querying-semantic-data en https://code.google.com/p/tdwg-rdf/wiki/Beginners6SPARQL.

SPARQL uitvoer
De uitvoer van SPARQL kan op verschillende manieren aan de gebruiker worden getoond. Standaard toont Virtuoso het resultaat als een tabel op een webpagina. Er zijn ook andere uitvoermogelijkheden. De belangrijkste alternatieven zijn een spreadsheet, CSV (comma-separated values) en TSV (tab-separated values). Alle drie zijn geschikt voor weergave in een spreadsheet. Het verschil is dat bij de keuze 'spreadsheet' direct een spreadsheet-programma (meestal Excel) wordt geopend. Bij de andere formaten ontstaat alleen een bestand dat in het spreadsheet kan worden geopend. Daarbij kun je eventueel aangeven dat bij het lezen van het bestand van de encoding UTF-8 gebruik gemaakt moet worden. Daarmee wordt voorkomen dat lettertekens met diakrieten gecorrumpeerd raken.

Een paar afspraken
Elke zoekactie in SPARQL bestaat uit een of meerdere commando's die zijn opgebouwd uit SPARQL syntax en variabelen (waarmee je specifieke delen van STCN selecteert/aanroept).

Er zijn een paar regelmatigheden in de commando's die die je bij elk zoekcommando dat je schrijft in de gaten moet houden:

1. de **SPARQL syntax** staat in **hoofdletters**: dat is een conventie, zodat je die syntax goed kunt herkennen en kunt onderscheiden van andere onderdelen van een commando.

2. de **variabelen** beginnen met **vraagteken**;
In eenvoudige zoekacties staat vooraan het subject dat je zoekt, in het midden het predikaat en achteraan het object, aangeduid met vrij te kiezen naam (letter, woord).

3. Verschillende onderdelen worden omgeven door **open en sluit-accolades.** Let er op dat elke geopende accolade ook wordt afgesloten.

4. In deze handleiding maken we waar mogelijk gebruik van **prefixes** in plaats van volledige uri's. Zie voor een overzicht bijlage 2.

3.1 Eenvoudigste zoekopdracht
Maak een filter op de triplestore

- geeft alle triples die een type aangeven
- select *: alle beschikbare velden (?s en ?o) komen terug in het resultaat

```
SELECT *
WHERE {
?s dc:type ?o
}
```

3.2 Select van een genoemde variabele

- geeft alleen nog maar de typen (?o)

```
SELECT ?o
WHERE {
?s dc:type ?o
}
```

3.3 Select DISTINCT waarde

- geeft alleen nog unieke waarden

```
SELECT DISTINCT ?o
WHERE {
?s dc:type ?o
}
```

Het resultaat van deze query is te vinden in bijlage 2.

3. 5 Met behulp van 'AS' een herkenbare naam geven aan de uitvoer

```
SELECT DISTINCT ?o AS ?typen
WHERE {
?s dc:type ?o
}
```

3.6. Alle triples van een bepaald type opvragen

- hier: alle drukkers

```
SELECT *
WHERE {
?s dc:type "drukker"@nl
}
```

3.7 Alle gegevens van één drukker

- willekeurig predikaat en object

```
SELECT *
WHERE {
<http://data.kb.nl/thesaurus/075540614> ?p ?o
}
```

Toelichting: Hierin is http://data.kb.nl/thesaurus/075540614 de identificatie van de drukker.

3.8 Alle drukkers met hun namen

- let op de punt aan het eind van de condities!

```
SELECT *
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
}
```

Toelichting: Hier, zoals elders, is de naam van de variabele vrij. We hadden ook ?drukker in plaats van ?s kunnen gebruiken.

3.9 Verbinding tussen twee objecten: drukkers en hun boeken

- de verbinding wordt gelegd door middel van '?b dc:publisher ?s.'

```
SELECT *
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?b dc:publisher ?s.
?b dc:title ?t
}
```

3.10 Resultaten ordenen en tellen

- gebruik van COUNT(*): tel het aantal triples (hier: boeken)
- gebruik van ORDER BY: op volgorde van naam drukker

```
SELECT ?name, COUNT(*) AS ?aantal_boeken
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?b dc:publisher ?s.
}
ORDER BY ?name
```

- gebruik van ORDER BY DESC: op aflopende volgorde

```
SELECT ?name, COUNT(*) AS ?aantal_boeken
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?b dc:publisher ?s.
}
ORDER BY DESC(?aantal_boeken)
```

- beperk het aantal rijen in het resultaat: LIMIT 100

```
SELECT ?name, COUNT(*) AS ?aantal_boeken
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?b dc:publisher ?s.
}
ORDER BY DESC(?aantal_boeken)
LIMIT 100
```

3.11 Welke trefwoorden bestaan er?

```
SELECT DISTINCT ?l
WHERE {
?s dc:type "trefwoord"@nl.
?s skos:prefLabel ?l
}
ORDER BY ?l
```

3.12 Alle titels met een bepaald trefwoord

```
SELECT *
WHERE {
?b dc:title ?t.
?b dc:subject ?s.
?s skos:prefLabel "Hebrew language and literature"
}
```

3.13 Titels met het ene of het andere trefwoord

- OR wordt uitgedrukt met een UNION:

```
SELECT count(*)
WHERE {
?b dc:title ?t.
?b dc:subject ?s.
{       {?s skos:prefLabel "Hebrew language and literature"}
        UNION
        {?s skos:prefLabel "Greek language and literature"} }
}
```

- introduceer een nieuw waarde in het resultaat: VALUES optie

```
SELECT *
WHERE {
?b dc:title ?t.
{       ?b dc:subject ?s.
        ?s skos:prefLabel "Hebrew language and literature"
        VALUES ?label {"H"}}
UNION
{       ?b dc:subject ?s.
        ?s skos:prefLabel "Greek language and literature"
        VALUES ?label {"G"}}
}
ORDER BY ?label
```

3.14 bewerk één van de uitvoervelden: BIND

- BIND creëert een nieuwe variabele in het resultaat (hier ?shorttitle)
- gebruik van functies: hier STRBEFORE (string-before):

```
SELECT *
WHERE {
?b dc:title ?t.
?b dc:subject ?s.
?s skos:prefLabel "Hebrew language and literature"
BIND (STRBEFORE(?t,'/') as ?shorttitle)
}
```

- effect: pakt alleen deel van titel voorafgaand aan de slash (‘/’)
- andere functies: http://www.w3.org/TR/2013/REC-sparql11-query-20130321/

3.15 Verfijning door gebruik IF.

```
SELECT *
WHERE {
?b dc:title ?t.
?b dc:subject ?s.
?s skos:prefLabel "Hebrew language and literature"
BIND (IF (CONTAINS(?t, '/'), STRBEFORE(?t, '/'), ?t) as ?shorttitle)
}
```

- pakt alleen deel van titel voor '/', maar als er geen '/' in staat, pak dan de hele titel

3.16 Selecteer op woord in titel

```
SELECT *
WHERE {
?b dc:title ?t.
FILTER (CONTAINS(?t, 'ieught'))
}
```

3.17 Selecteer met behulp van een reguliere expressie die rekening houdt met diverse spellingen

```
SELECT *
WHERE {
?b dc:title ?t.
FILTER REGEX(?t, "(i|j)(eu|ue)(gh?|ch)(t|d)", "i")
}
```

Toelichting: reguliere expressies bieden een mogelijkheid om op basis van zoekpatronen heel precies te zoeken. Het bovenstaande patroon komt overeen met een zoekactie op een *i* of een j, gevolgd door *eu* of *ue*, gevolgd door een *g* en eventueel een *h*, of anders *ch*, en dan een *t* of een *d*. De “i” geeft aan dat hoofdletters als kleine letters worden behandeld. Op basis hiervan worden gevonden: ‘ieught’, ‘jeugd’, ‘JUEGT’, etc.

Bespreking van alle mogelijkheden van reguliere expressies valt buiten het bestek van deze handleiding. Zie http://www.regular-expressions.info/ voor meer informatie.

- OPTIONAL: laat mogelijkheid open dat er geen triple is dat aan de eisen voldoet.
- Titels met hun auteur, of alleen de titel als er geen auteur is:

```
SELECT *
WHERE {
?b dc:title ?t.
OPTIONAL {?b dc:creator ?c.
                    ?c skos:prefLabel ?n}
FILTER (CONTAINS(?t, 'grondslagen'))
}
LIMIT 100
```

3.18 Typografische annotaties

```
SELECT *
WHERE {
?b dc:title ?t.
?b dcx:annotation ?a.
}
LIMIT 100
```

3.19 Typografische annotaties, maar nu gecombineerd voor een titel

```
SELECT ?t, GROUP_CONCAT(?a; separator=", ")
WHERE {
?b dc:title ?t.
FILTER (CONTAINS(?t, 'notarissen'))
?b dcx:annotation ?a.
}
GROUP BY ?b ?t
LIMIT 100
```

3.20 NOT EXISTS: titels zonder typografische annotaties

```
SELECT *
WHERE {
?b dc:title ?t.
FILTER NOT EXISTS {?b dcx:annotation ?a}.
}
LIMIT 100
```

4. Voorbeelden zoekvragen

We geven hieronder drie sets van zoekvragen rond een bepaalde onderzoeksvraag. De eerste deelvragen zijn steeds met vrij eenvoudige SPRQL commando's te bevragen, de latere deelvragen zijn steeds ingewikkelder. Doel is het steeds het zoekresultaat in uiteindelijk één gecombineerd SPARQL commando te krijgen, zodat geen tussenstappen nodig zijn met export van data (waardoor echte analyse buiten SPARQL noodzakelijk wordt).

SET 1 MET VRAGEN OVER OMVANG PRODUCTIE – uit onderzoek Marieke van Delft
Als je in algemene zin kijkt naar de gemiddelde productie van een Leidse en een Alkmaarse drukker, blijkt daar een enorm verschil tussen te bestaan. Leidse drukkers produceerden gemiddeld 44,5 titels terwijl bij een drukker in Alkmaar gemiddeld 13 van de pers rolden. Hiervoor kun je allerlei verklaringen bedenken. Een mogelijke verklaring zou kunnen zijn dat in Alkmaar dikkere boeken gedrukt werden. Om hier iets over te zeggen kan het nuttig zijn om de genres die in de respectieve steden gedrukt werden te vergelijken. Dat is wat met onderstaande vragen gedaan wordt.

Vraag
1. Vergelijk welke genres er in Alkmaar respectievelijk Leiden gedrukt werd tussen 1550-1800

Deelvraag 1A: kijk welke drukkers actief waren in Alkmaar

```
SELECT *
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?s skos:editorialNote ?note.
FILTER (CONTAINS(?note, 'Alkmaar'))
}
```

Toelichting: Bij deze vraag lopen we direct tegen een probleem aan dat veroorzaakt wordt doordat bij de RDF-conversie verschillende velden van de STCN bijeen elkaar gevoegd zijn in het veld editorialNote vanwege het vasthouden aan bepaalde standaarden (SKOS, Dublin Core), die eigenlijk uit elkaar gehouden hadden moeten worden. Er zijn in principe twee manieren om via de STCN te bepalen of een boek in Alkmaar is gedrukt. De eerste ingang is het impressum van het boek. Het impressum bevat echter niet noodzakelijkerwijs een gestandaardiseerde plaatsnaam. Bovendien is het in de RDF-versie van de STCN niet beschikbaar. Het alternatief is om te kijken naar de vestigingsplaats van de drukker. Ook die is echter niet gemakkelijk uit de beschikbare gegevens te bepalen. Wat we hier als eerste benadering doen, is kijken naar alle drukkers bij wie de term 'Alkmaar' voorkomt in de editorialNote. Bij drukkers die in meerdere plaatsen actief zijn geweest, loopt dit dus niet goed. (Er zijn ook drukkers met een editorialNote met de waarde 'Z.pl. 1606-1610, Amsterdam 1614-1615, 1619, Dordrecht 1617, Middelburg 1617, Alkmaar 1619, Delft 1619-1623, Utrecht 1621. Var.: A. Lenertsz, A. Leendertsz, A.L. Camer').

We selecteren dus alle triples uit de triplestore met type ‘drukker’. Voor de gevonden subjecten vragen we hun naam (‘skos:prefLabel’) op, en de beschikbare noten; van de noten kijken we vervolgens of er het woord ‘Alkmaar’ in voorkomt.

Deelvraag 1B: kijk welke genres er bestaan

```
SELECT DISTINCT ?l
WHERE {
?b dc:subject ?t.
?t skos:prefLabel ?l
}
```

Toelichting: De volgende vraag is: hoe bepalen we genre? Genre-aanduidingen komen soms voor in de titel van de werken (*Kluchtighe comedie van Ardelia en Flavioos vryagie*), maar daar kun je moeilijk van uitgaan. Hier kiezen we voor een selectie via trefwoorden. In de RDF-versie van de STCN zijn die te benaderen via dc:subject. Lang niet alle trefwoorden hebben echter met genre te maken. Om te beginnen vragen we hier alle trefwoorden op (= alles wat voorkomt als dc:subject), en daarvan de labels. Het trefwoord zelf is namelijk alleen een URI.

Uit de hier getoonde lijst selecteren we voor de volgende deelvraag de genres ‘Drama’, ‘Poetry’, ‘Occasional writings’ en ‘Songbooks’.

Deelvraag 1C: kijk naar de boeken die in bepaalde genres zijn gepubliceerd

```
SELECT *
WHERE {
?b dc:subject ?t.
{
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Drama')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Songbooks')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Poetry')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Occasional writings')).}
}
}
```

Toelichting: we maken een selectie op basis van de trefwoorden. Omdat we meerdere trefwoorden willen combineren, gebruiken we een UNION (we ‘verenigen’ de resultaten voor de verschillende trefwoorden).

Deelvraag 1D: kijk welke aantallen werken er in deze genres in Alkmaar werden gepubliceerd

```
SELECT ?l, COUNT(*)
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?s skos:editorialNote ?note.
?b dc:publisher ?s.
?b dc:subject ?t.
{
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Drama')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Songbooks')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Poetry')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Occasional writings')).}
}
FILTER (CONTAINS(?note, 'Alkmaar'))
}
GROUP BY ?l
```

Toelichting: Hier combineren we de zoekvragen van 1A en 1C. De boeken (?b) van 1C ‘knopen’ we aan de drukkers (?s) van 1A via het criterium ‘?b dc:publisher ?s’. We bepalen de totalen per trefwoord (?l).

Deelvraag 1E: Vergelijk nu Leiden en Alkmaar

```
SELECT ?place, ?l, COUNT(*)
WHERE {
?s dc:type "drukker"@nl.
?s skos:prefLabel ?name.
?b dc:publisher ?s.
?b dc:subject ?t.
{
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Drama')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Songbooks')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Poetry')).}
UNION
{?t skos:prefLabel ?l.
FILTER (CONTAINS(?l, 'Occasional writings')).}
}
{
{?s skos:editorialNote ?note.
FILTER (CONTAINS(?note, 'Alkmaar')).
VALUES ?place {'Alkmaar'} }
UNION
{?s skos:editorialNote ?note.
FILTER (CONTAINS(?note, 'Leiden')).
VALUES ?place {'Leiden'} }
}
}
GROUP BY ?place ?l
ORDER BY ?place ?l
```

Toelichting: We voegen een tweede UNION in, nu om zowel drukkers uit Alkmaar als uit Leiden mee te kunnen nemen. Met de VALUES-clausule kennen we een label toe aan de gevonden drukkers (respectievelijk ‘Alkmaar’ en ‘Leiden’).

SET 2 MET GENREVRAGEN – uit onderzoek Els Stronks

We kennen in de vroegmoderne boekproductie allerlei genres waar we vrij gemakkelijk genrekenmerken voor vast kunnen stellen (bijvoorbeeld aan de hand van eigentijdse poëtica's, denk bijvoorbeeld aan het treurspel). Er zijn ook genres waarvoor genrekenmerken heel veel lastiger vast te stellen bleken, mogelijk omdat genrekenmerken daarvoor niet in klassieke poëtica's maar op de vroegmoderne boekenmarkt zelf voor het eerst gestalte kregen. Een voorbeeld van zo'n genre is het pamflet. Deze set met zoekvragen gaat over een ander genre dat tot nu toe niet veel aandacht kreeg, maar dat mogelijk ook een vastomlijnd iets werd op de vroegmoderne boekenmarkt.

Het gaat om zogenaamde 'how to'- boeken (zoals onze *Word for Dummies*), hoeveel kunnen we over dit soort boeken via STCN te weten komen, en kunnen we op basis daarvan zien of er sprake was van een soort herkenbaar genre?

Deelvraag 2: Welke boeken hebben in de titel varianten van het woord 'handboek'?

```
SELECT * WHERE {
?b dc:type schema:Book.
?b dc:title ?t.
FILTER REGEX(?t, "han[dts-]{0,3}bo[eu]", "i").
}
```

Toelichting: We vragen om boeken ('?b dc:type schema:Book') en halen daarvoor ook de titel op (dc:title). We filteren de resultaten met behulp van een reguliere expressie (REGEX). We zouden kunnen filteren met 'CONTAINS': dan kunnen we wel kijken of bijvoorbeeld het woord 'handboek' in de titel voorkomt, maar dan missen we spellingsvarianten. Vandaar de keuze om te filteren met een reguliere expressie.

Lees de reguliere expressie als volgt: de zoekterm bevat de letters 'han', in die volgorde, gevolgd door maximaal drie ({0,3}) tekens uit het groepje 'dts-', gevolgd door de letters 'bo', in die volgorde, gevolgd door een 'e' of een 'u'. Resultaten zijn dus bijvoorbeeld: handboek, hantboek, handt-bouck etc. De parameter 'i' zorgt dat in de zoekactie hoofdletters als kleine letters worden behandeld. 'Handboek' wordt dus ook gevonden.

Deelvraag 2B: Welke boeken hebben in de titel varianten van de woorden handleiding, instructie(boek), aanleiding, handboek, enchiridion.

```
SELECT * WHERE {
?b dc:type schema:Book.
?b dc:title ?t.
FILTER REGEX(?t, "(han([dts-]{0,3})leid)|(a[ae]nle[iy])|(ench(e|i|ei)r)|(han([dts-
]{0,3})bo[eu])|(instruc)", "i").
}
```

Toelichting: We breiden onze zoekterm uit met een aantal verwante termen. De termen staan tussen haakjes en worden gescheiden door een verticale streep ('|'). Deze scheidt alternatieven.

Deelvraag 2C: Hoeveel van die boeken bevatten illustraties?

```
SELECT COUNT(DISTINCT ?b) WHERE {
?b dc:type schema:Book.
?b dc:title ?t.
FILTER REGEX(?t, "(han([dts-]{0,3})leid)|(a[ae]nle[iy])|(ench(e|i|ei)r)|(han([dts-]{0,3})bo[eu])|(instruc)", "i").
?b dcx:annotation ?a.
FILTER (CONTAINS(?a, 'illustratie'))
}
```

Toelichting: We breiden de zoekopdracht uit 2B uit met een selectie op typografische kenmerken (‘?b dcx:annotation ?a’). Het kenmerk moet het woord ‘illustratie’ bevatten. We zijn alleen geïnteresseerd in het aantal boeken met illustraties, vandaar ‘COUNT(DISTINCT ?b)’. Daarmee vragen we het aantal boeken met deze eigenschap op. Als we ‘COUNT(?b)’ gebruikt hadden (zonder ‘DISTINCT’) zouden we boeken met meerdere typografische kenmerken dubbel tellen.

Deelvraag 2D: Over welke onderwerpen (trefwoorden als theology, geography) gaan boeken met die titels?

```
SELECT ?l, COUNT(*) AS ?aantal WHERE {
?b dc:type schema:Book.
?b dc:title ?t.
FILTER REGEX(?t, "(han([dts-]{0,3})leid)|(a[ae]nle[iy])|(ench(e|i|ei)r)|(han([dts-]{0,3})bo[eu])|(instruc)", "i").
?b dc:subject ?s.
?s skos:prefLabel ?l.
}
GROUP BY ?l
ORDER BY DESC(?aantal)
```

Toelichting: We breiden de zoekopdracht uit 2B uit met een selectie van trefwoorden (‘?b dc:subject ?s’) en vragen het bijbehorende label op (‘?s skos:prefLabel ?l’).

Deelvraag 2E: Hoe is de productie van boeken met die woorden in titels over de periode 1550-1800 verspreid?

```
SELECT ?period, COUNT(*)  WHERE {
?b dc:type schema:Book.
?b dc:title ?t.
FILTER REGEX(?t, "(han([dts-]{0,3})leid)|(a[ae]nle[iy])|(ench(e|i|ei)r)|(han([dts-]{0,3})bo[eu])|(instruc)", "i").
?b dc:date ?j.
FILTER REGEX(?j, "^[0-9]{3}", "i").
BIND (xsd:decimal(SUBSTR(?j,1,3))*10 AS ?period).
FILTER (?period >= 1550).
FILTER (?period < 1800).
}
GROUP BY ?period
ORDER BY ?period
```

Toelichting: We breiden de query uit met het jaar van publicatie (‘?b dc:date ?j’). Het publicatiejaar kan ook de vorm hebben ‘166X’ of ‘16XX’. Met een reguliere expressie dwingen we af dat de eerste drie posities numeriek zijn (‘^[0-9]{3}’). We bekijken de boekproductie per periode van tien jaar. We bepalen de periode waartoe een boek behoort:
- door eerst de drie posities van het jaar te pakken met behulp van ‘SUBSTR(?j,1,3)’ (‘1666’ wordt ‘166)
- vervolgens het stukje tekst ‘166’ om te zetten naar het getal 166 (functie xsd:decimal)
- en het resultaat met tien te vermenigvuldigen (levert 1660)

Vervolgens eisen we nog dat de periode groter is dan 1550 en kleiner dan 1800. We vragen de totalen per periode op, op volgorde van periode.

Deelvraag 2F: Hoe verhoudt hoe die productie zich verhoudt tot alle andere boeken die in diezelfde periode?

```
SELECT ?groep, ?period, COUNT(*)  WHERE {
?b dc:type schema:Book.
{
{?b dc:title ?t.
FILTER REGEX(?t, "(han([dts-]{0,3})leid)|(a[ae]nle[iy])|(ench(e|i|ei)r)|(han([dts-]{0,3})bo[eu])|(instruc)", "i").
VALUES ?groep {'Instructie'}}
UNION
{?b dc:title ?t.
VALUES ?groep {'Alles'}}
{
?b dc:date ?j.
FILTER REGEX(?j, "^[0-9]{3}", "i").
BIND (xsd:decimal(SUBSTR(?j,1,3))*10 AS ?period).
}
GROUP BY ?groep ?period
ORDER BY ?groep ?period
```

Toelichting: We doen hetzelfde als in 2E, maar nu ook een keer voor alle boeken, in plaats van alleen de selectie met de juiste titelwoorden. Daarvoor gebruiken we weer een UNION-constructie. De VALUES-clausule kent een label toe aan de beide groepen.

SET 3 MET LEEFTIJDSVRAGEN – uit onderzoek Willemijn Zwart

Vondel bracht in zijn *Aenleidingh ter Nederduitsche dichtkunst* het ideaal onder woorden van de schrijver die na jarenlang oefenen excelleerde in zijn vak. Omdat dat ideaalbeeld zo sterk is, hebben we nog nooit onderzoek gedaan naar de jonge auteurs uit de periode 1550-1800. Wat droegen die bij aan de productie van literatuur?

Deelvraag 3A: Geef de leeftijd van een auteur bij publicatie.

```
SELECT *
WHERE {
?p dc:type "person"@en.
?p skos:prefLabel ?o.
?p schema:birthDate  ?pd.
?b dc:creator  ?p.
?b dc:date  ?bd.
FILTER REGEX(?o, "[0-9]{4}(?!.*fl.)", "i").
FILTER REGEX(?pd, "^[0-9]{4}$", "i").
FILTER REGEX(?bd, "^[0-9]{4}$", "i") .
FILTER (xsd:decimal(?pd) > 1550) .
FILTER (xsd:decimal(?pd) < 1800) .
BIND ((xsd:decimal(?bd) - xsd:decimal(?pd)) AS ?age).
}
```

Toelichting: We vragen personen op met naam en geboortedatum. We vragen naar de boeken die de betreffen de persoon heeft geschreven. Vervolgens bepalen we de leeftijd bij publicatie.

Hierbij moeten we rekening houden met een aantal beperkingen van de data: het jaar van publicatie is soms alleen bij benadering bekend, en ook leefjaren van auteurs zijn vaak onbekend. Sommige jaartallen die in de STCN worden gegeven zijn ‘floruit’-jaartallen: ze geven de bloeiperiode van een auteur in plaats van geboorte- en sterfjaar.

In deze zoekopdracht selecteren we dus op publicatiejaar en geboortejaar gelijk aan een combinatie van vier cijfers (dus niet bijvoorbeeld ‘156X’). Bovendien verlangen we dat in de naam van de auteur een jaartal niet wordt voorafgegaan door ‘fl.’ (dus geen ‘floruit’-leefperiode). De leeftijd bepalen we als het verschil tussen publicatiedatum (?bd, bookdate) en geboortedatum (?pd, person date).

Deelvraag 3B: Op welke leeftijd kwam in de periode 1550-1800 het eerste werk van een auteur in gedrukte vorm op de markt?

```
SELECT ?p, ?o, MIN(?age), ?period
WHERE {
?p dc:type "person"@en.
?p skos:prefLabel ?o.
?p schema:birthDate  ?pd.
?b dc:creator  ?p.
?b dc:date  ?bd.
FILTER REGEX(?o, "[0-9]{4}(?!.*fl.)", "i").
FILTER REGEX(?pd, "^[0-9]{4}$", "i").
FILTER REGEX(?bd, "^[0-9]{4}$", "i") .
FILTER (xsd:decimal(?pd) > 1550) .
FILTER (xsd:decimal(?pd) < 1740) .
BIND ((xsd:decimal(?bd) - xsd:decimal(?pd)) AS ?age).
BIND (floor(xsd:decimal(?pd)/25)*25 AS ?period).
FILTER (?age < 80).
FILTER (?age > 5).
}
```

Toelichting: Aan de vorige opdracht hebben we nu de berekening van ?periode (hier per 25 jaar) toegevoegd. We filteren op ?age < 80 om auteurs die pas na hun dood worden gedrukt (denk ook aan buitenlanders) uit te filteren. We filteren ook auteurs uit die publiceren voor hun vijfde jaar (waarschijnlijk databasefouten). De leeftijd bij eerste publicatie bepalen we door het minimum te nemen van de berekende leeftijden.

Deelvraag 3C: Wat is het aantal publicaties van jongeren vs. dat van ouderen (jong is alles beneden de 30)?

```
SELECT ?agegroup, COUNT(*) AS ?n
WHERE {
?p dc:type "person"@en.
?p skos:prefLabel ?o.
?p schema:birthDate  ?pd.
?b dc:creator  ?p.
?b dc:date  ?bd.
FILTER REGEX(?o, "[0-9]{4}(?!.*fl.)", "i").
FILTER REGEX(?pd, "^[0-9]{4}$", "i").
FILTER REGEX(?bd, "^[0-9]{4}$", "i") .
FILTER (xsd:decimal(?pd) > 1550) .
FILTER (xsd:decimal(?pd) < 1740) .
BIND ((xsd:decimal(?bd) - xsd:decimal(?pd)) AS ?age).
FILTER (?age < 80).
FILTER (?age > 5).
BIND (IF (?age >=30, 'OLD', 'YOUNG') as ?agegroup).
}
GROUP BY ?agegroup
```

Toelichting: Ten opzichte van query 3A hebben we hier de berekening van een agegroup toegevoegd. De variabele krijgt de waarde OLD voor iedereen boven de 30, YOUNG voor alle anderen. We kijken naar het totaal aantal publicaties voor beide leeftijdsgroepen.

Deelvraag 3D: schrijven jonge auteurs over andere onderwerpen dan oude auteurs (jong is alles beneden de 30)?

```
SELECT ?agegroup, ?l, COUNT(*) AS ?n
WHERE {
?p dc:type "person"@en.
?p skos:prefLabel ?o.
?p schema:birthDate  ?pd.
?b dc:creator  ?p.
?b dc:date  ?bd.
FILTER REGEX(?o, "[0-9]{4}(?!.*fl.)", "i").
FILTER REGEX(?pd, "^[0-9]{4}$", "i").
FILTER REGEX(?bd, "^[0-9]{4}$", "i") .
FILTER (xsd:decimal(?pd) > 1550) .
FILTER (xsd:decimal(?pd) < 1740) .
BIND ((xsd:decimal(?bd) - xsd:decimal(?pd)) AS ?age).
FILTER (?age < 80).
FILTER (?age > 5).
BIND (IF (?age >=30, 'OLD', 'YOUNG') as ?agegroup).
?b dc:subject ?s.
?s skos:prefLabel ?l.
}
GROUP BY ?l ?agegroup
ORDER BY ?l ?agegroup
```

Toelichting: We breiden het vorige commando uit met een selectie op steekwoorden (subjects), en groeperen mede daarop. Bij het bekijken van het resultaat houden we in gedachte dat van de hele collectie ongeveer 10% is geschreven door jongeren (zie 3C). Houd daarbij rekening met het feit dat de STCN niet weet wanneer een werk een herdruk is: wanneer een boek van een jongere wordt herdrukt als hij ouder dan 30 is, telt het boek als dat van een oudere.

Bijlage 1: De conversie naar RDF

In deze bijlage vind je een overzicht van de conversie van de velden zoals die gebruikt worden bij het maken van STCN-records in het GGC. Bij de conversie naar RDF zijn niet alle elementen meegenomen. Gekozen is voor de velden die nodig waren bij het onderzoek van de KB-fellow. Als de kolom ‘RDF’ leeg is, betekent dit dat dit veld niet aanwezig in de RDF data en er dus niet op gezocht kan worden.

BIBLIOGRAFISCHE RECORDS

Pica3	Pica+	RDF	Toelichting	Kenmerken
797	003@			
0500	002@	<dc:type>	waar in het GGC 'Abv' wordt gebruikt, vermeld men in de STCN 'Acv' en andersom	verplicht; niet herhaalbaar
1100	011@	<dc:date>		verplicht; niet herhaalbaar
1200	014@	<dcx:annotation>	typografisch kenmerk	verplicht; herhaalbaar
1500	010@	<dc:language>	/1, /2 of /3	verplicht; niet herhaalbaar (subvelden /1/2/3 binnen KMC herhaalbaar)
1700	019@		/1 of /2	verplicht; niet herhaalbaar (/2 herhaalbaar)
2275	007P		vingerafdruk	verplicht; herhaalbaar
3000	028A of 028 B	<dc:creator>	primair auteur	niet verplicht; niet herhaalbaar (028B herhaalbaar)
301x	028C (met numerus currens)	<dc:contributor>	secundaire auteur	niet verplicht; herhaalbaar met vervolgnummer (3012, 3013, etc.)
3210, 3211, 3220	025@		sorteertitels	niet verplicht; niet herhaalbaar
3261	027A/0X		extra zoekingang	niet verplicht; herhaalbaar met vervolgnummer (3262, 3263, etc.)
3400	150C		lokaal hoofdwoord; in gebruik bij voor de echte naam bij pseudoniemen of mystificaties in KMC 3000	niet verplicht; niet herhaalbaar
4000	021A	<dc:title>	titel	verplicht; niet herhaalbaar
4020	032@		editievermelding	niet verplicht; niet herhaalbaar
4040	033D		impressum	verplicht; niet herhaalbaar
4043	033J	<dc:publisher>	boekverkoper/drukker	verplicht; herhaalbaar
4060	034D	<dcterms:extent>	collatieformule	verplicht; niet herhaalbaar
4062	034I	<dcterms:extent>	formaat	verplicht; niet herhaalbaar
4160	036D		koepeltitel	niet verplicht; herhaalbaar
4201	037A		annotatie	niet verplicht; herhaalbaar
4400			wordt alleen gebruikt bij een anoniem hoofdwoord in afwijkende spelling	niet verplicht; niet herhaalbaar
4701	147A		werkaantekening	niet verplicht; herhaalbaar
4711	147B/01		beschrijver	verplicht; niet herhaalbaar

4712	147B/02		collatieformule bij meerdelige werken	niet verplicht; niet herhaalbaar
6501	144Z/01	<dc:subject>	geografisch trefwoord	niet verplicht; herhaalbaar: (onderscheid d.m.v. subvelden)
651X	144Z/1X	<dc:subject>	trefwoorden; vormdescriptoren	verplicht; herhaalbaar (tot en met 6519)
700X	208@/0X		selectiesleutel	verplicht; herhaalbaar als volgnummer (7002, 7003, etc.)
7100	209A/0X		signatuur	verplicht; niet herhaalbaar (KMC blijft 7100, de pica+-code verandert met het aantal exemplaren)
7134	209S/0X	<dc:identifier>	illustratie(s); fondslijst(en); full-tekst in de RDF alleen verwijzingen naar de full-tekst	niet verplicht; herhaalbaar (0X is het nummer van het exemplaarblok in kwestie)

AUTEURSNAMEN

Pica3	**Pica+**	**RDF**	**Toelichting**	**Kenmerken**
	002-	<dc:type>	soort en status	verplicht; niet herhaalbaar
	003-	<skos:Concept>	PPN	
	003-	<skos:exactMatch>	PPN	
009	009B		code	niet verplicht; niet herhaalbaar
008	010-	<skos:prefLabel>	taalcode	niet verplicht, niet herhaalbaar (subveld is wel herhaalbaar)
006	013A		indicator	vooralsnog alleen gebruikt bij brieven; niet herhaalbaar (subveld wel herhaalbaar)
007	019-		landcode	niet verplicht; niet herhaalbaar (subveld wel herhaalbaar)
100	028A	<skos:prefLabel>	naam zoals in de publicatie	niet verplicht; niet herhaalbaar
200	028-	<skos:altLabel>	naamsvariant	niet verplicht; herhaalbaar
110	028B	<skos:altLabel>	meest volledige naam	niet verplicht; niet herhaalbaar
300	032A	<schema:birthDate>	leefjaren	niet verplicht; niet herhaalbaar
300	032A	<schema:deathDate>	leefjaren	
300	032A	<skos:prefLabel>	leefjaren	
310	032B	<skos:prefLabel>	identificerende gegevens	niet verplicht; niet herhaalbaar
710	032C		beroepsaanduiding	niet verplicht; herhaalbaar
720	032D		naam als vrije tekst; bij vrouwen: naam van de echtgenoot	niet verplicht; herhaalbaar
400	038A	<skos:scopeNote>	verwijzing van pseudoniem of persoonsaanduiding naar eigennaam	niet verplicht; herhaalbaar
410	038B	<skos:scopeNote>	verwijzing van eigennaam naar pseudoniem of persoonsaanduiding	niet verplicht; herhaalbaar
9XX	047A	<skos:editorialNote>	scope note of toelichting	niet verplicht; herhaalbaar
9XX	047A	<skos:scopeNote>	scope note of toelichting	

DRUKKERS

Pica3	Pica+	RDF	Toelichting	Kenmerken
005	002@			verplicht; niet herhaalbaar
009	009B		indicator voor eeuw, plaats en functie	niet verplicht; niet herhaalbaar
180	033A	<skos:prefLabel>	naam	verplicht; niet herhaalbaar
280	033@	<skos:altLabel>	naamsvariant	niet verplicht; herhaalbaar
740	033C	<skos:editorialNote>	plaats plus jaren van vestiging	niet verplicht; herhaalbaar
741	033D	<skos:editorialNote>	beroepsaanduiding plus jaren van uitoefening	niet verplicht; herhaalbaar
742	033E	<skos:editorialNote>	adres plus jaren van vestiging	niet verplicht; herhaalbaar
743	033F	<skos:editorialNote>	uithangbord plus jaren	niet verplicht; herhaalbaar
900	047A	<skos:editorialNote>	werkaantekeningen: plaats, jaartallen, naamsvarianten	niet verplicht; herhaalbaar
910	047A/10	<skos:scopeNote>	annotatie: vooral m.b.t. fictieve gegevens	

TREFWOORDEN

Pica3	Pica+	RDF	Toelichting	Kenmerken
005	002@			verplicht; niet herhaalbaar
009	009B		indicator	niet verplicht; niet herhaalbaar
160	044A	<skos:prefLabel>	trefwoord; zonder onderscheid naar soort	verplicht; niet herhaalbaar
260	044B	<skos:altLabel>	trefwoord; extra ingang (synoniemen van 160)	niet verplicht; herhaalbaar

Bijlage 2: Beschikbare namespaces

Prefix	URI
dc	http://purl.org/dc/elements/1.1/
dcmi	http://purl.org/dc/dcmitype/
dcterms	http://purl.org/dc/terms/
dcx	http://krait.kb.nl/coop/tel/handbook/telterms.html#
foaf	http://xmlns.com/foaf/0.1/
geo	http://www.w3.org/2003/01/geo/wgs84_pos#
kbc	http://data.kb.nl/catalogus/
kbt	http://data.kb.nl/thesaurus/
owl	http://www.w3.org/2002/07/owl#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfa	http://www.w3.org/ns/rdfa#
rdfs	http://www.w3.org/2000/01/rdf-schema#
schema	http://schema.org/
skos	http://www.w3.org/2004/02/skos/core#
void	http://rdfs.org/ns/void#
xml	http://www.w3.org/XML/1998/namespace
xsd	http://www.w3.org/2001/XMLSchema#

Bijlage 3: Typen objecten in de triplestore

De triplestore bevat de volgende soorten objecten:

http://schema.org/Book
http://purl.org/dc/dcmitype/Text
http://schema.org/Periodical
"person"@en
"persoon"@nl
"keyword"@en
"trefwoord"@nl
"drukker"@nl
"printer"@en