https://github.com/ISE-FIZKarlsruhe/bikidata

# bikidata

FIZ-ISE Seminar 16 Aug 2023

Etienne Posthumus
eposthumus@gmail.com https://epoz.org/

bietjie = beetje
“A little bit”

*pronounced:* biki

## Why?

We often use Wikidata in our research.

…for queries

…for extraction

But the WDQS has time-outs, or we want to do something more advanced for analysis

*Can we share efforts?* Instead of all doing custom grep etc. separately

## Own triplestore?

Blazegraph / Virtuoso / QLever etc.

This is a non-trivial and time-consuming exercise.

(never mind resource-intensive)

## TL;DR

- Download Truthy Wikidata .nt dump, split into separate chunks.
- Hash, s-p-o IRIs to 64-bit integers
- Hash IRI/Literal to 64-bit integers
- Store everything in Parquet files
- Query using DuckDB
- …
- 🥳

## Choosing a hash algorithm

Murmurhash, SIP, Cityhash, Spookyhash, xxhash, FNV, MD5, CRC, SHA

And does it really need to be 64-bits? Isn’t 32 bits enough?

## Choosing a hash algorithm

Murmurhash, SIP, Cityhash, Spookyhash, xxhash, FNV, MD5, CRC, SHA

And does it really need to be 64-bits? Isn’t 32 bits enough?

…and the winner is: xxhash
https://github.com/Cyan4973/xxHash

## Looks like

```
|          s           |          p           |          o           |

| 10504872225212057570 | 12580884072737042117 |  1877790563993361007 |
| 10504872225212057570 |  1969734015126054522 |  5587686886852776007 |
| 10504872225212057570 | 15314476376240055071 |  5464438232649063526 |
| 10504872225212057570 |  3176906616339849618 |  8616764692071278374 |
| 10504872225212057570 |  3176906616339849618 |  8661715247485442335 |
| 10504872225212057570 |  3176906616339849618 |  1734327773552895906 |
| 10504872225212057570 |  3176906616339849618 | 11573360774852176845 |
```

## Looks like

```
|          s           |          p           |          o           |

| 10504872225212057570 | 12580884072737042117 |  1877790563993361007 |
| 10504872225212057570 |  1969734015126054522 |  5587686886852776007 |
| 10504872225212057570 | 15314476376240055071 |  5464438232649063526 |
| 10504872225212057570 |  3176906616339849618 |  8616764692071278374 |
| 10504872225212057570 |  3176906616339849618 |  8661715247485442335 |
| 10504872225212057570 |  3176906616339849618 |  1734327773552895906 |
| 10504872225212057570 |  3176906616339849618 | 11573360774852176845 |
```

```
10504872225212057570 | <http://www.wikidata.org/entity/Q42>
```

## Looks like

```
| 10504872225212057570 | 13787943592591698645 | 18303551757152610549 | "4ce2ba117755a"                                                     |
| 10504872225212057570 | 13149575899394925943 |  5620510158999179407 | "39a33dc4-5a81-4d67-91d6-1daecdb854e3"                              |
| 10504872225212057570 | 16872716867785849655 |  9380107508382309212 | "2078791"                                                           |
| 10504872225212057570 | 11643480590671441163 | 16342048489805386123 | <http://www.wikidata.org/entity/Q1860>                              |
| 10504872225212057570 | 17591124141766173371 | 11286963682984007955 | "ncf10168152"                                                       |
| 10504872225212057570 |  2502498544802895930 | 14455623418673618869 | "DouglasAdams"                                                      |
| 10504872225212057570 | 12928713851174965354 | 11645328310747403540 | <http://kbpedia.org/kko/rc/DouglasAdams>                            |
| 10504872225212057570 | 12457161947493164903 | 17795690644028887794 | "26792807"                                                          |
| 10504872225212057570 |  4881929489000864687 |  2501072958756939372 | <http://www.wikidata.org/entity/Q2687578>                           |
| 10504872225212057570 | 17657798667172610268 |  4885215682934094547 | <http://id.loc.gov/authorities/classification/PR6051.D3352>         |
| 10504872225212057570 |  3029897129149013315 |  9225015895394000726 | "215957"                                                            |
| 10504872225212057570 |  4497732400159369300 | 18099947938301090450 | <http://www.wikidata.org/entity/Q6173448>                           |
| 10504872225212057570 |  2002846622502823662 | 13975239749449742978 | <http://www.wikidata.org/entity/Q7066>                              |
| 10504872225212057570 |  9838106236933647348 | 11690699215486955192 | "DNA"@en
```

## DuckDB

https://duckdb.org/

An in-process SQL OLAP database management system

## Parquet files

https://parquet.apache.org/

Apache Parquet is an open source, column-oriented data file format designed for efficient data storage and retrieval. It provides efficient data compression and encoding schemes with enhanced performance to handle complex data in bulk.

## Put it all together and we can…

```
select count(*) from 'tri.parquet'

where p=1969734015126054522 and o = 5587686886852776007
```

```
10 646 247
```

## Put it all together and we can…

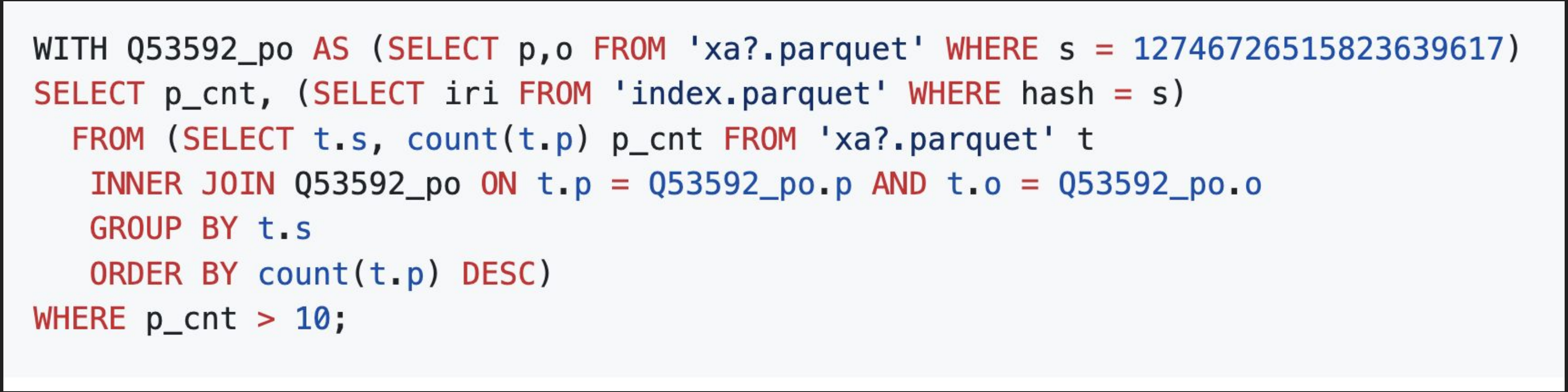
```
select count(*) from 'tri.parquet'

where p=1969734015126054522 and o = 5587686886852776007
```

```
  10 646 247
```

```
select hash, literal from 'map.parquet' where hash in
(1969734015126054522, 5587686886852776007)
```

```
 1969734015126054522 | <http://www.wikidata.org/prop/direct/P31>
 5587686886852776007 | <http://www.wikidata.org/entity/Q5>
```

## Or more usefully:



```sql
WITH Q53592_po AS (SELECT p,o FROM 'xa?.parquet' WHERE s = 12746726515823639617)
SELECT p_cnt, (SELECT iri FROM 'index.parquet' WHERE hash = s)
  FROM (SELECT t.s, count(t.p) p_cnt FROM 'xa?.parquet' t
   INNER JOIN Q53592_po ON t.p = Q53592_po.p AND t.o = Q53592_po.o
   GROUP BY t.s
   ORDER BY count(t.p) DESC)
WHERE p_cnt > 10;
```

```
| <http://www.wikidata.org/entity/Q13442814>  | 39371483 |
| <http://www.wikidata.org/entity/Q5>         | 10646247 |
| <http://www.wikidata.org/entity/Q4167836>   |  5135154 |
| <http://www.wikidata.org/entity/Q16521>     |  3552731 |
| <http://www.wikidata.org/entity/Q523>       |  3291341 |
| <http://www.wikidata.org/entity/Q7318358>   |  2096006 |
| <http://www.wikidata.org/entity/Q318>       |  2091076 |
| <http://www.wikidata.org/entity/Q4167410>   |  1436977 |
| <http://www.wikidata.org/entity/Q113145171> |  1252930 |
| <http://www.wikidata.org/entity/Q11173>     |  1249605 |
| <http://www.wikidata.org/entity/Q7187>      |  1213201 |
| <http://www.wikidata.org/entity/Q8054>      |   992097 |
| <http://www.wikidata.org/entity/Q11266439>  |   797866 |
| <http://www.wikidata.org/entity/Q3305213>   |   695859 |
| <http://www.wikidata.org/entity/Q79007>     |   650505 |
| <http://www.wikidata.org/entity/Q13442814>  | 39371483 |
| <http://www.wikidata.org/entity/Q5>         | 10646247 |
| <http://www.wikidata.org/entity/Q4167836>   |  5135154 |
| <http://www.wikidata.org/entity/Q16521>     |  3552731 |
| <http://www.wikidata.org/entity/Q523>       |  3291341 |
| <http://www.wikidata.org/entity/Q7318358>   |  2096006 |
| <http://www.wikidata.org/entity/Q318>       |  2091076 |
| <http://www.wikidata.org/entity/Q4167410>   |  1436977 |
| <http://www.wikidata.org/entity/Q113145171> |  1252930 |
| <http://www.wikidata.org/entity/Q11173>     |  1249605 |
| <http://www.wikidata.org/entity/Q7187>      |  1213201 |
| <http://www.wikidata.org/entity/Q8054>      |   992097 |
| <http://www.wikidata.org/entity/Q11266439>  |   797866 |
| <http://www.wikidata.org/entity/Q3305213>   |   695859 |
| <http://www.wikidata.org/entity/Q79007>     |   650505 |
```

```
create temp table tipes as select o,
count(s) t from 'tri.parquet' where
p=1969734015126054522 group by o;

select m.literal, tipes.t from tipes join
'map.parquet' m on tipes.o = m.hash
order by t desc limit 15;
```

## Try it

teach02:/home/wikidata/

42M duckdb (the executable)

17G map.parquet

41G tri.parquet

## Try it

teach02:/home/wikidata/

pip install duckdb

https://github.com/ISE-FIZKarlsruhe/bikidata

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:

  Persons (P31 Q5)

  Person-plus1

  Books

  *Which others?*

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:

  Persons (P31 Q5)

  Person-plus1

  Books

  *Which others?*

- A cookbook of sample queries

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:

  Persons (P31 Q5)

  Person-plus1

  Books

  *Which others?*

- A cookbook of sample queries
- A “labels” service, similar to the WDQS SERVICE

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:
  Persons (P31 Q5)
  Person-plus1
  Books
  *Which others?*

- A cookbook of sample queries
- A “labels” service, similar to the WDQS SERVICE
- Better Python library, publish to PYPI

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:
  Persons (P31 Q5)
  Person-plus1
  Books
  *Which others?*

- A cookbook of sample queries
- A “labels” service, similar to the WDQS SERVICE
- Better Python library, publish to PYPI

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:
  Persons (P31 Q5)
  Person-plus1
  Books
  *Which others?*

- A cookbook of sample queries
- A “labels” service, similar to the WDQS SERVICE
- Better Python library, publish to PYPI
- Index literals using embeddings and a HNSW (SBERT + FAIS?)

## TODO list

- Want to sort the triple table, but how? Does that improve query speed?
- Make smaller extracts:
  Persons (P31 Q5)
  Person-plus1
  Books
  *Which others?*

Your suggestions?

- A cookbook of sample queries
- A “labels” service, similar to the WDQS SERVICE
- Better Python library, publish to PYPI
- Index literals using embeddings and a HNSW (SBERT + FAIS?)