POLITECNICO
MILANO 1863

Project Work SMBUD 22/23

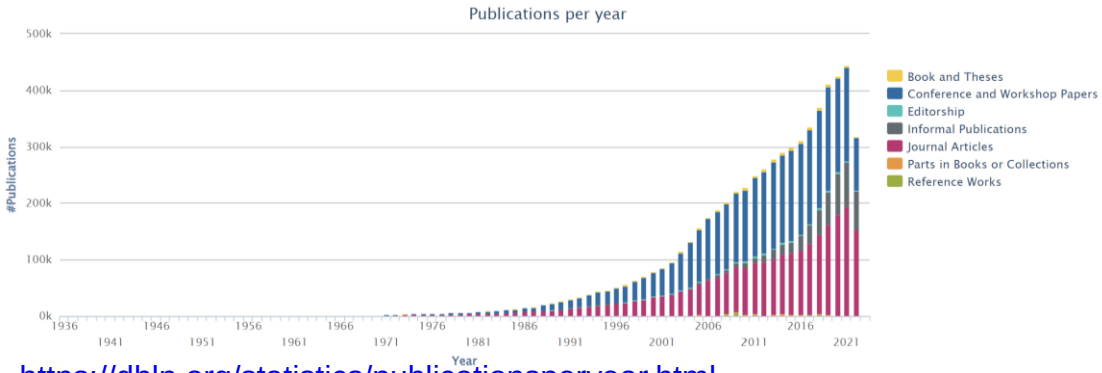
Bibliography Database

Problem Presentation

Assumptions

1. Truthful data
2. People = Authors + Editors
3. Each Author belogs to one University
4. Each Publication has a globally unique DOI
5. Publication types: book, article, incollection, inproceeding, thesis
6. Each Publication can be written by multiple Authors
7. Each Publication can be edited by multiple Editors

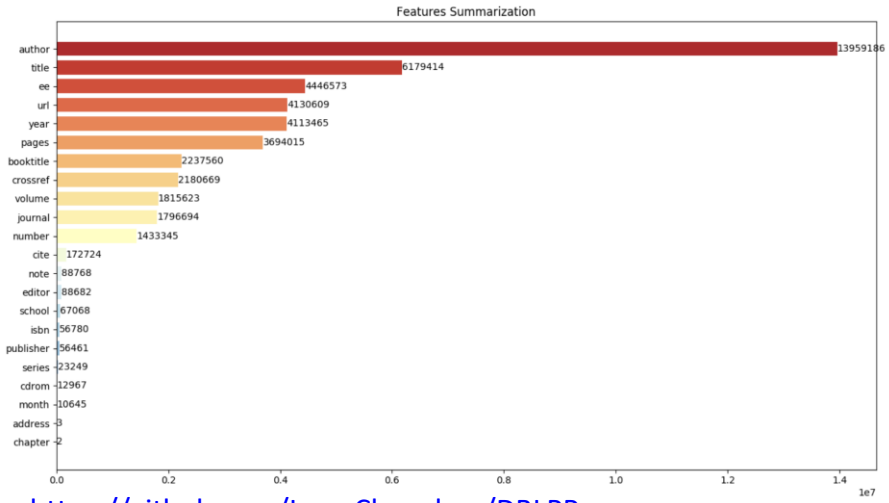
ER Diagram



Publications type stats

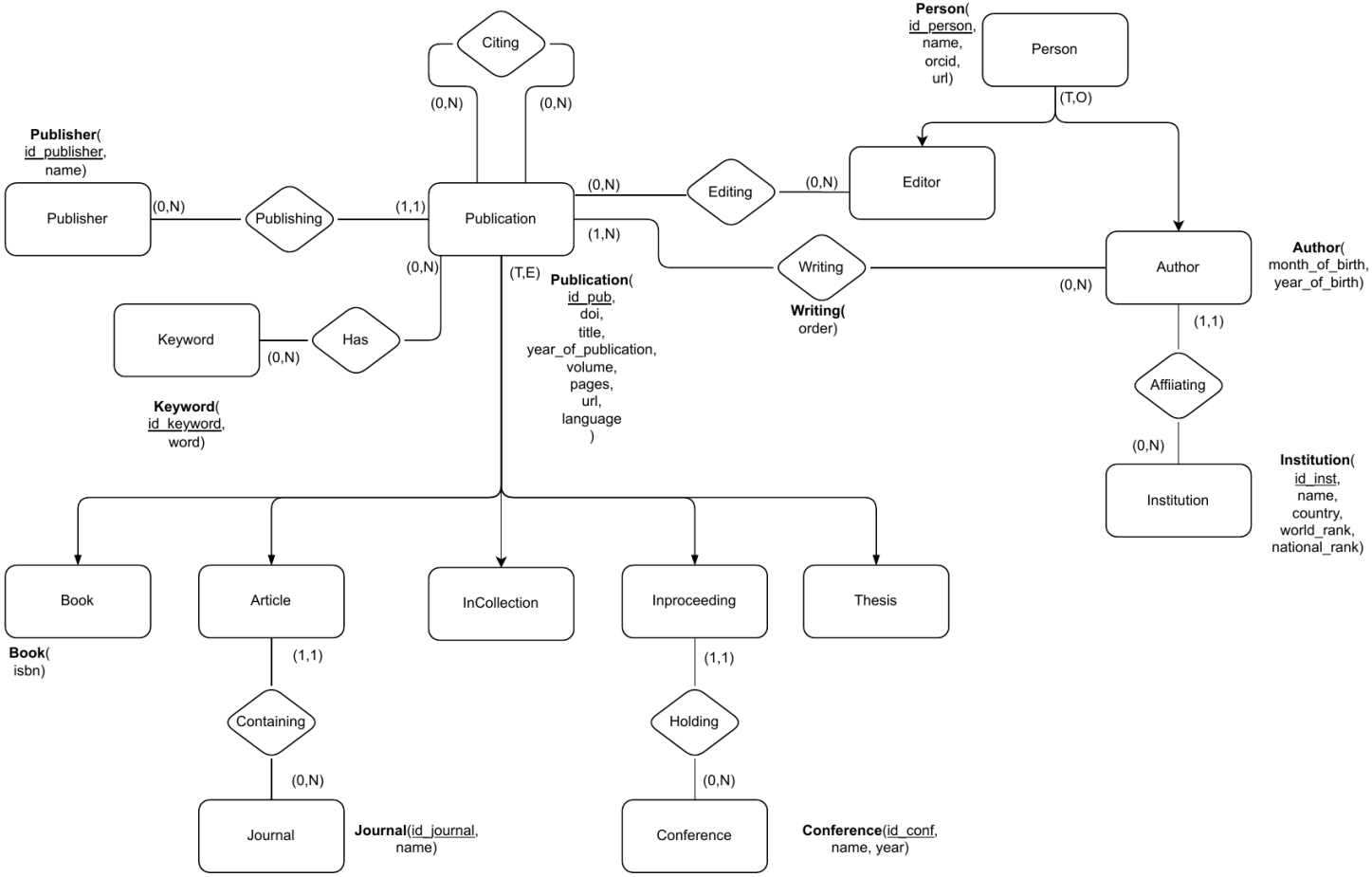
https://dblp.org/statistics/publicationsperyear.html

Publications attributes stats



https://github.com/IsaacChanghau/DBLPParser

ER Diagram

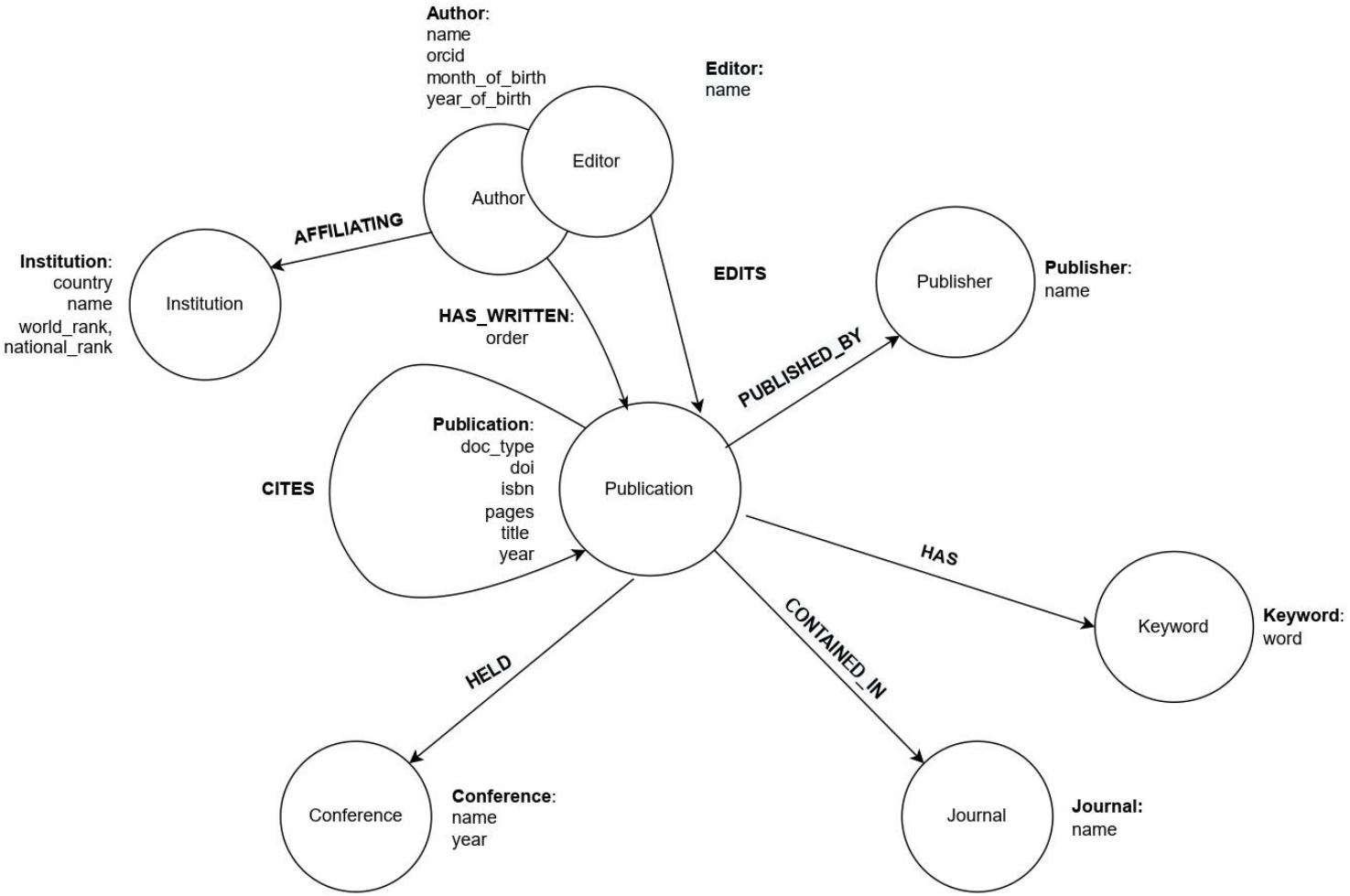

Citing
(0,N)
(0,N)
Person(
id_person,
name,
orcid,
url)
Person
(T,O)
Publisher(
id_publisher,
name)
Publisher
(0,N)
Publishing
(1,1)
Publication
(0,N)
Editing
(0,N)
Editor
(1,N)
(0,N)
(T,E)
Publication(
id_pub,
doi,
title,
year_of_publication,
volume,
pages,
url,
language
)
Writing
Writing(
order)
(0,N)
Author
Author(
month_of_birth,
year_of_birth)
(1,1)
Keyword
(0,N)
Has
Keyword(
id_keyword,
word)
Affiliating
(0,N)
Institution
Institution(
id_inst,
name,
country,
world_rank,
national_rank)
Book
Article
InCollection
Inproceeding
Thesis
Book(
isbn)
(1,1)
(1,1)
Containing
Holding
(0,N)
(0,N)
Journal
Journal(id_journal,
name)
Conference
Conference(id_conf,
name, year)

Dataset description

1. Parsing of XML dump from DBLP with a Python script
2. Each entity and relationship have a separate CSV file
3. Parsed only relevant attributes
4. Citations and keywords are generated

```
<proceedings key="conf/bics/2013" mdate="2021-05-25">
        <editor>Derong Liu 0001</editor>
        <editor>Cesare Alippi</editor>
        <editor>Dongbin Zhao</editor>
        <editor orcid="0000-0002-8080-082X">Amir Hussain 0001</editor>
        <title>Advances in Brain Inspired Cognitive Systems -
            6th International Conference, BICS 2013, Beijing, China,
            June 9-11, 2013. Proceedings</title>
        <year>2013</year>
        <publisher>Springer</publisher>
        <series href="db/series/lncs/index.html">Lecture Notes
            in Computer Science</series>
        <volume>7888</volume>
        <ee>https://doi.org/10.1007/978-3-642-38786-9</ee>
        <isbn>978-3-642-38785-2</isbn>
        <booktitle>BICS</booktitle>
        <url>db/conf/bics/bics2013.html</url>
</proceedings>
```

Graph DB
Neo4j



From Nodes…

Listing 8 Load 1.5 - Publications

```
LOAD CSV WITH HEADERS FROM 'file:///publications.csv'
    AS row FIELDTERMINATOR '|'
CREATE (p:Publication {
    doc_type: row.doc_type,
    doi: row.id,
    isbn: row.isbn,
    pages: row.pages,
    title: row.title,
    year: toIntegerOrNull(row.year)
});
```

…To Relationships

Listing 10 Load 2.1 - Citations

```
LOAD CSV WITH HEADERS FROM "file:///citation.csv"
    AS row FIELDTERMINATOR "|"
MATCH (p1:Publication{doi:row.document})
MATCH (p2:Publication{doi:row.cite})
MERGE (p1)-[:CITES]->(p2)
```

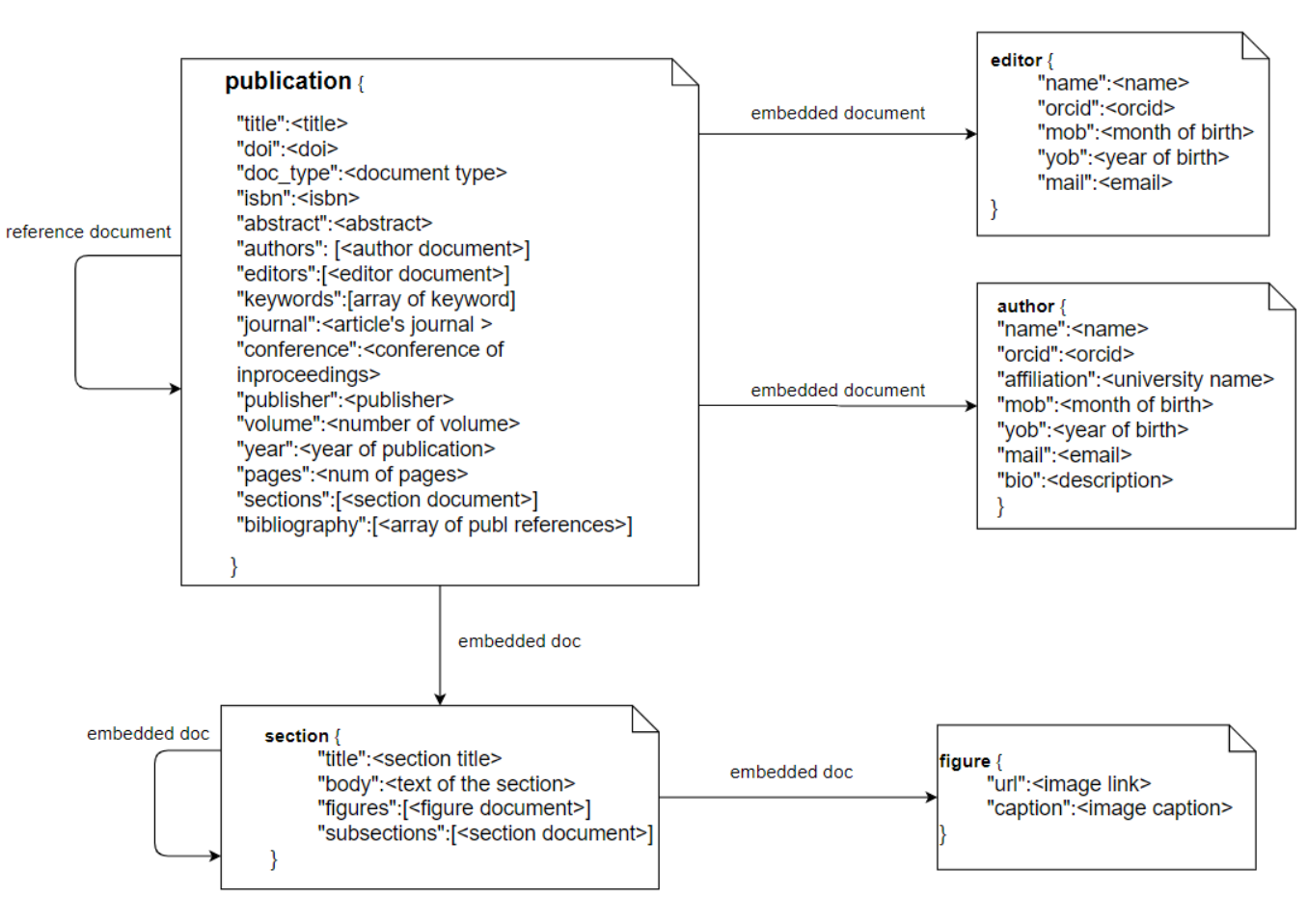
Graph DB
Query

```
MATCH (pub:Publication)-[:CONTAINED_IN]->(j:Journal)
MATCH (pub)<-[:HAS_WRITTEN]-()-[:AFFILIATING]->(inst:Institution)
WITH COUNT(DISTINCT inst.country) AS num_countries, pub
WHERE pub.year > 2010 AND pub.year < 2020
RETURN num_countries, pub.title AS publication, pub.year AS year
ORDER BY num_countries DESC LIMIT 5
```

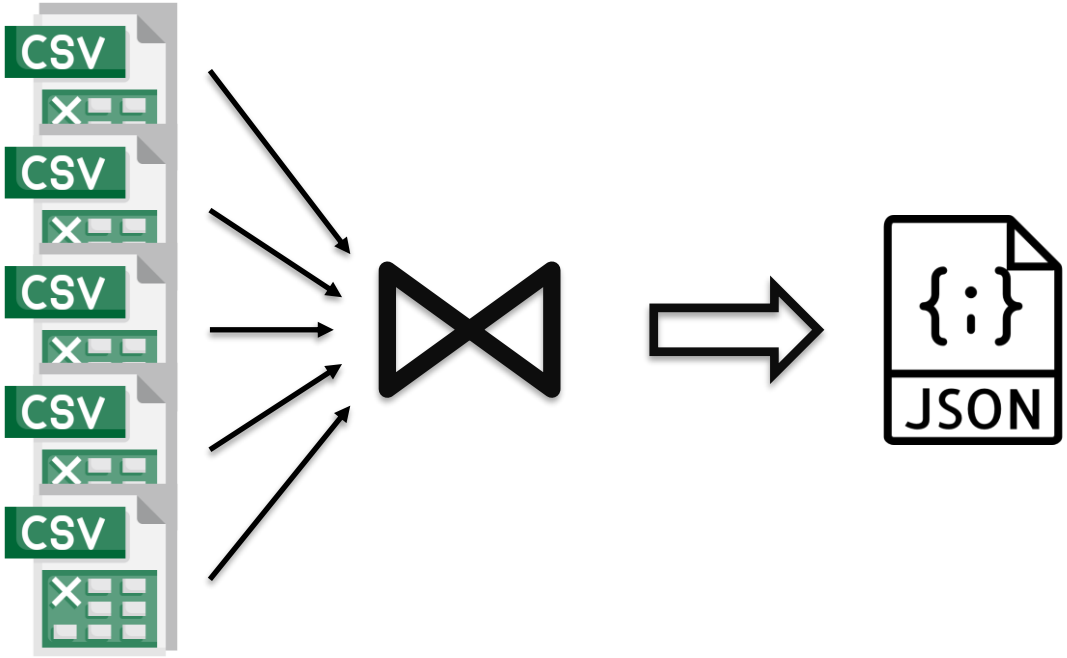
Find the 5 publications contained in a journal and made in a year between 2010 and 2020, that are written by the highest number of authors coming from institutions located in different countries.

"num_countries"	"publication"	"year"
9	"ALACRITY: Analytics-Driven Lossless Data Compression for Rapid In-Situ Indexing, Storing, and Querying."	2013
8	"Contribution of the Living Lab approach to the development, assessment and provision of assistive technologies for supporting older adults with cognitive disorders."	2013
7	"Opening Up Data Analysis for Medical Health Services: Data Integration and Analysis in Cancer Registries with CARESS."	2016
7	"Horizontal Business Process Model Integration."	2015
7	"Triage Support Algorithm for Patients Classification at Urgency Care Area in a Hospital."	2013

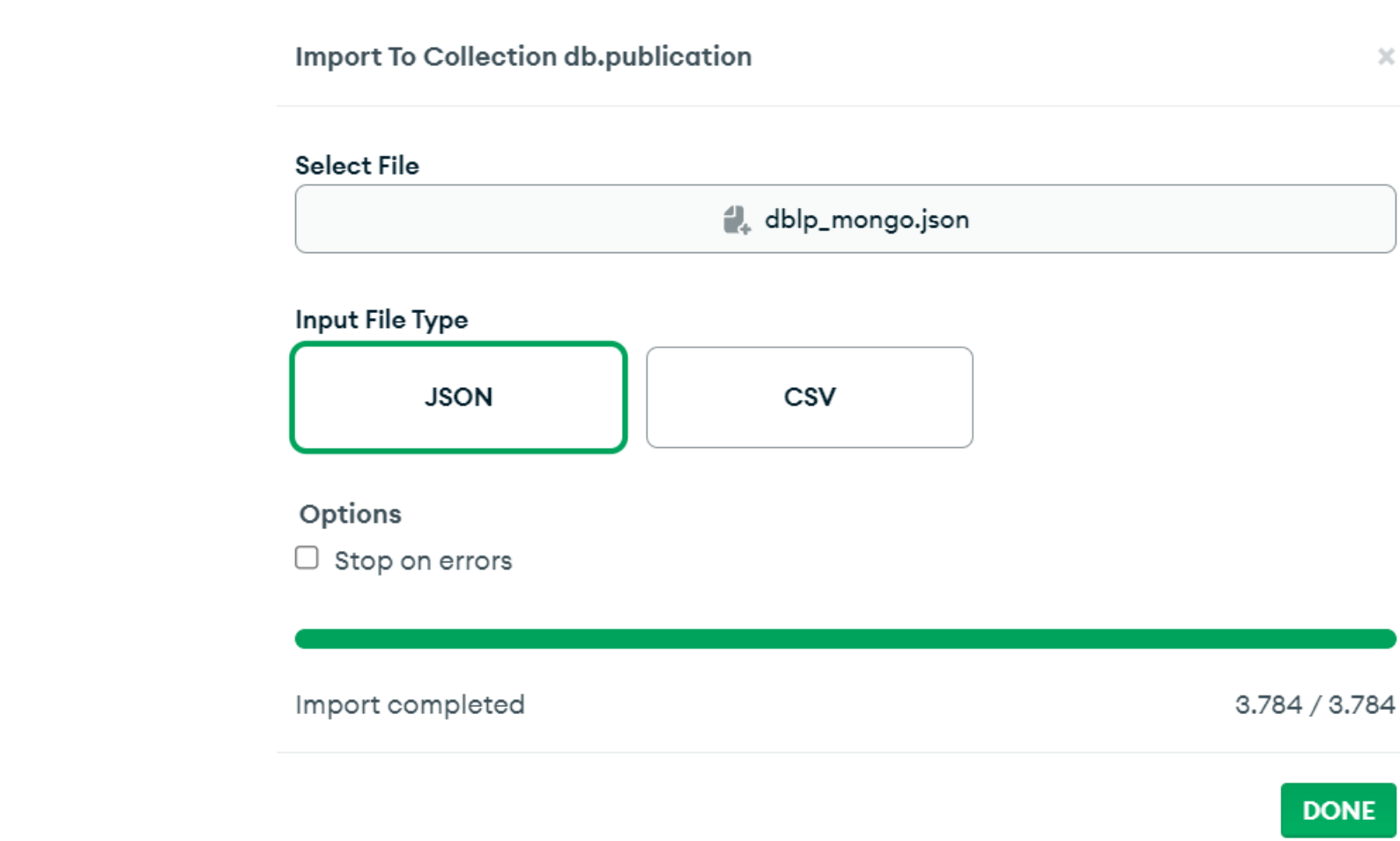
Document DB
MongoDB



Document DB
Data Upload



Document DB
Data Upload



Find the top 3 most cited inproceedings (title, doi, year) written by an author(s) affiliated to University of Milan.

```
db.publications.aggregate([
  { $match: {"authors.affiliation": "University of Milan",
                    doc_type: "inproceedings"  } },
  { $lookup: {
      from: "publications",
      localField: "doi",
      foreignField: "bibliography",
      as: "ref_by"
    } },
  { $addFields: { ref_by_count: {$size: "$ref_by" } } },
  { $project: { title: 1, doi: 1, year: 1, ref_by_count: 1 },
  { $sort: {ref_by_count: -1} },
  { $limit: 3 }
])
```

Execution time with no indexes: 50ms.

Execution time with with indexes on doi and authors.affiliation: 25ms.

Document DB
Query

Find the top 3 most cited inproceedings (title, doi, year) written by an author(s) affiliated to University of Milan.

```
{ _id: ObjectId("63790d7815acad8c496ec240"),
  doi: 'https://doi.org/10.1109/COOPIS.1997.613816',
  title: 'Query Modification in Object-Oriented Database Federations.',
  year: 1997,
  ref_by_count: 4 }
{ _id: ObjectId("63790d7815acad8c496ec102"),
  doi: 'https://doi.org/10.1007/3-540-36124-3_50',
  title: 'Reconciling Replication and Transactions for the End-to-End Reliability of CORBA Applications.',
  year: 2002,
  ref_by_count: 2 }
{ _id: ObjectId("63790d7815acad8c496ec26c"),
  doi: 'https://doi.org/10.1109/NOTERE.2010.5536814',
  title: 'Risk Characterization and Prototyping.',
  year: 2010,
  ref_by_count: 1 }
```

```
# Path of the directory with all the csv files
path_of_the_directory = "...\dataset"

dataset = {}

for filename in os.listdir(path_of_the_directory):
    f = os.path.join(path_of_the_directory, filename)
    if os.path.isfile(f):

        # Load a DataFrame for each csv file
        df = spark.read.option("header", True).option("delimiter", "|") \
            .option("inferSchema",True).csv(f)

        # Create a dictionary where key=filename and value=dataframe
        dataset[filename.split(".")[0]] = df
```

NAME OF THE KEY: **authors**
root
 |-- author_name: string (nullable = true)
 |-- orcid: string (nullable = true)
 |-- month_of_birth: integer (nullable = true)
 |-- year_of_birth: integer (nullable = true)
 |-- mail: string (nullable = true)

NAME OF THE KEY: **publications**
root
|-- id: string (nullable = true)
|-- title: string (nullable = true)
|-- year: integer (nullable = true)
|-- pages: integer (nullable = true)
|-- isbn: string (nullable = true)
|-- doc_type: string (nullable = true)

NAME OF THE KEY: **write_relationship**
root
 |-- author_name: string (nullable = true)
 |-- pub_id: string (nullable = true)
 |-- author_order: string (nullable = true)

NAME OF THE KEY: **work_relationship**
root
 |-- author_name: string (nullable = true)
 |-- university: string (nullable = true)

The number of publications written by authors with a Polimi email grouped by year, starting from 2010.

```
# Collect into an array all the authors with a Polimi email
polimi_authors = dataset["authors"].filter(col("mail").rlike("polimi")) \
                                   .select(collect_set("author_name")) \
                                   .collect()[0][0]

dataset["publications"].join(dataset["write_relationship"],
    dataset["publications"].id == dataset["write_relationship"].pub_id) \
    .filter(col("author_name").isin(polimi_authors) & (col("year") >= "2010")) \
    .groupBy("year") \
    .agg(countDistinct("title").alias("Number of publications")) \
    .sort(col("Number of publications").desc()) \
    .show()
```

year	Number of publications
2018	167
2015	132
2016	80
2020	78
2013	71
2019	55
2010	47
2011	44
2017	44
2014	36
2021	35
2022	35
2012	32
2023	2

THE END