# TED[Together]

*Share, learn, discuss.*

<Irene Bravo [1071820]> <Paolo Leidi [1059156]> <Matteo Mariani [1057026]> <Michele Zenoni [1059950]>

# > AWS Glue - Watch Next

TED[Together]

```
68  ## READ WATCH_NEXT DATASET
69  watch_next_dataset_path = "s3://mz-unibg-data-tcm2021/watch_next_dataset.csv"
70  watch_next_dataset = spark.read.option("header","true").csv(watch_next_dataset_path)
71
72  # CREATE THE AGGREGATE MODEL, ADD WATCH_NEXT TO TEDX_DATASET
73  watch_next_dataset_agg = watch_next_dataset.groupBy(col("idx").alias("idx_ref")).agg(collect_list("watch_next_idx").alias("watch_next_idxs"))
74  #watch_next_dataset_agg.printSchema()
75  tedx_dataset_final = tedx_dataset_agg.join(watch_next_dataset_agg, tedx_dataset_agg._id == watch_next_dataset_agg.idx_ref, "left") \
76      .drop("idx_ref") \
77      .select(col("_id"), col("*"))
```

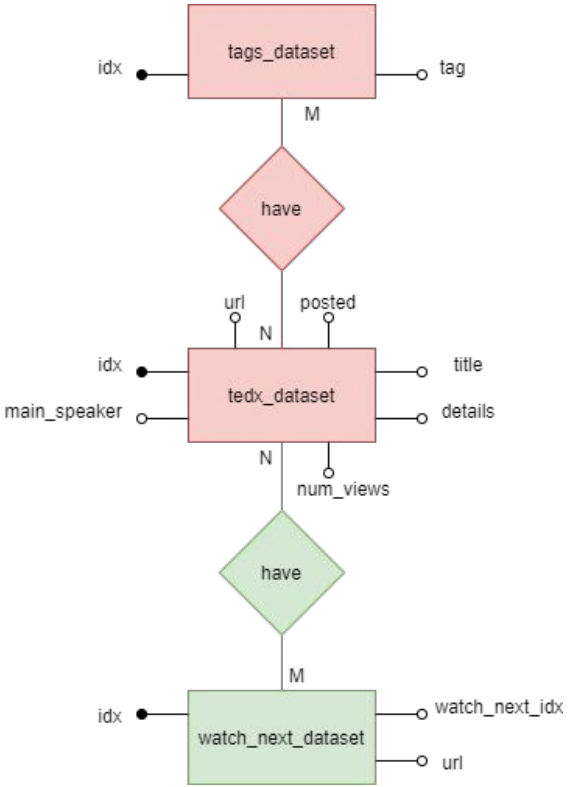
Abbiamo esteso il job PySpark “***CreateDataLake***” per aggiungere la lista degli ID dei talk “watch next” ad ogni video nella collection “***tedx_video_data***” di MongoDB.

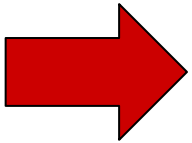
Dati trattati:

- **tedx_dataset.csv**
- **watch_next_dataset.csv**

# > Watch Next: modello E/R e risultato

TED[Together]




```
_id: "8d2005ec35280deb6a438dc87b225f89"
main_speaker: "Alexandra Auer"
title: "The intangible effects of walls"
details: "More barriers exist now than at the end of World War II, says designer..."
posted: "Posted Apr 2020"
url: "https://www.ted.com/talks/alexandra_auer_the_intangible_effects_of_wal..."
› tags: Array
⌄ watch_next_idxs: Array
    0: "5bd34fcc55d9e1267f605fa0c060d54e"
    1: "5bd34fcc55d9e1267f605fa0c060d54e"
    2: "9f7b1654e792011b7e1c6f4288520226"
    3: "fe35edd737282ab3a325f2387cf1b50b"
    4: "fe35edd737282ab3a325f2387cf1b50b"
    5: "9f7b1654e792011b7e1c6f4288520226"
    6: "d9896b41b372ec60cdd3c662e57caad3"
    7: "d9896b41b372ec60cdd3c662e57caad3"
    8: "9f7b1654e792011b7e1c6f4288520226"
    9: "5134ae81a27c94354173f38e84289ad5"
    10: "5134ae81a27c94354173f38e84289ad5"
    11: "9f7b1654e792011b7e1c6f4288520226"
    12: "8576654442b6633b1dc0eb48a989172a"
    13: "8576654442b6633b1dc0eb48a989172a"
    14: "9f7b1654e792011b7e1c6f4288520226"
    15: "078766d6cc461cf71d45dc268b66db95"
    16: "078766d6cc461cf71d45dc268b66db95"
    17: "9f7b1654e792011b7e1c6f4288520226"
```

# > AWS Glue - GetTagDescription

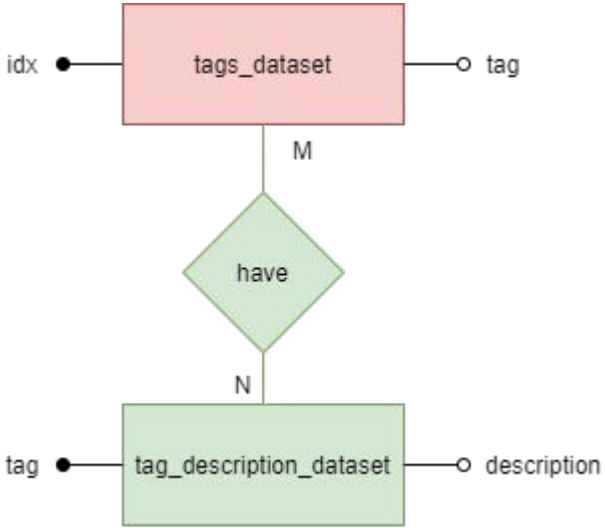
TED[Together]

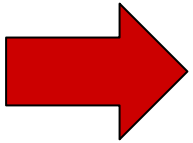
Il nuovo job PySpark ***"GetTagDescription"*** ci permette di creare una nuova collection in MongoDB (***"tedx_tag_data"***) che associa ad ogni tag una breve descrizione del suo significato estratta da Wikipedia e la lista degli ID dei talks che usano quel tag.

```
## READ TAGS DATASET
tags_dataset_path = "s3://mz-unibg-data-tcm2021/tags_dataset.csv"
tags_dataset = spark.read.option("header","true").csv(tags_dataset_path)

print("tags_dataset ready!\n")

tag_description_dataset_path = "s3://mz-unibg-data-tcm2021/tag_description_dataset.csv"
tag_description_dataset = spark.read.option("header","true").csv(tag_description_dataset_path)

tag_description_dataset = tag_description_dataset.groupBy(col("tag").alias("tag_ref")).agg(first("description").alias("description"))

print("tag_description_dataset ready!\n")

# CREATE THE AGGREGATE MODEL, ADD TAGS TO TEDX_DATASET
tags_dataset_agg = tags_dataset.groupBy(col("tag")).agg(collect_list("idx").alias("idxs"))

print("tags_dataset_agg ready!\n")

tags_dataset_final = tags_dataset_agg.join(tag_description_dataset, tags_dataset_agg.tag == tag_description_dataset.tag_ref, "left") \
    .drop("tag_ref") \
    .select(col("tag"), col("*"))
```

# > GetTagDescription: modello E/R e risultato

TED[Together]




```
_id: ObjectId("60a2ccfe25167721d06ed983")
tag: "crowdsourcing"
idxs: Array
  0: "76acb47b8da0314196ef7985500b8f52"
  1: "1f43ebbd6d5f2284d38aceb18e4f701a"
  2: "309945f52273c5a157020e0613cd637f"
  3: "3015b593bf0b568ab4b486df8d31c9be"
  4: "1c4c031d8c45d543da84b22337cabff6"
  5: "f600dfe87de34c1034b602480af6404e"
  6: "86fdde954c266abb824ea9c3d3d9cae0"
  7: "0e9e1ae2484bfd687a597fe22cc20a7f"
  8: "f761dc1f0293c2ed7ddf95f6add11646"
  9: "02a9e121ff52aa0a7192fa8689a14c5a"
  10: "06e3134ec06f6e4628d2558af114be29"
  11: "12072cf21860342ace16f12d1a56eeea"
  12: "5f29ec60adfa33e4602bf0bd7f639291"
description: "Crowdsourcing is a  sourcing ..."
```

# > Criticità

TED[Together]

- Incongruenza dei dataset:
  - ricalcolo dopo aggiornamento dei tag
  - alcuni tag potrebbero non avere una descrizione
- bisogna controllare che le descrizioni dei tag corrispondano effettivamente all’argomento trattato.

# > Possibili sviluppi

TED[Together]

- Estendere le descrizioni anche agli autori (main speaker)
- ricerca dei tag correlati
- migliorare l’algoritmo di acquisizione delle descrizioni
- creare una Lambda function che permetta di aggiornare in automatico le descrizioni dei tag anche al variare del dataset