# Databases Project – Spring 2018

Team No: 7

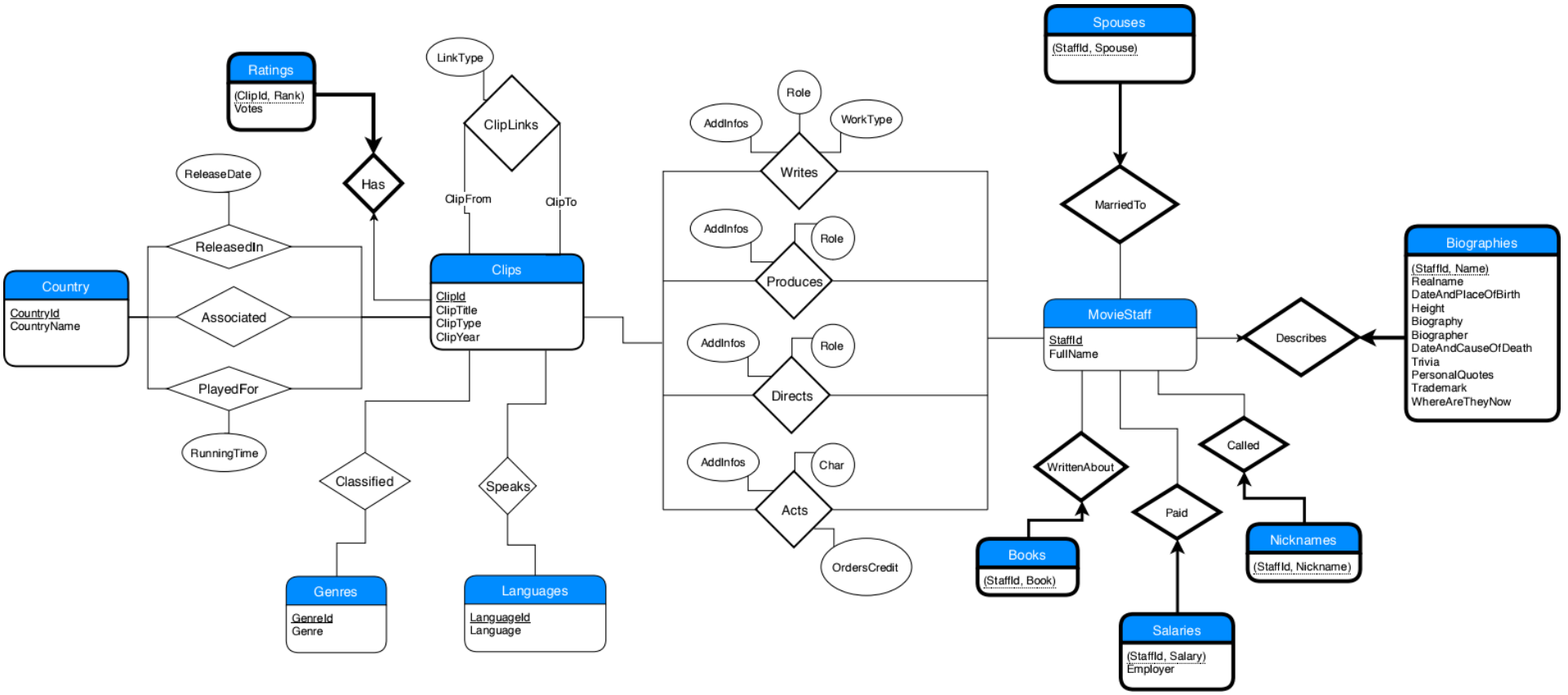
Da Rocha Rodrigues David Joaquim,
Torres Da Cunha Pedro Filipe ,
Justinas Sukaitis

:

# Table of Content

- ER-model
- Cleaning
  - General
  - Creating Ids
- 3rd part Query results
- Implementation
  - Tools
  - Interface
  - Demo

# ER-Model


Ratings
(ClipId, Rank)
Votes
Has
LinkType
ClipLinks
ClipFrom
ClipTo
ReleaseDate
ReleasedIn
Country
CountryId
CountryName
Associated
PlayedFor
RunningTime
Clips
ClipId
ClipTitle
ClipType
ClipYear
Classified
Speaks
Genres
GenreId
Genre
Languages
LanguageId
Language
Role
AddInfos
WorkType
Writes
Produces
Directs
Char
Acts
OrdersCredit
Spouses
(StaffId, Spouse)
MarriedTo
MovieStaff
StaffId
FullName
Describes
Biographies
(StaffId, Name)
Realname
DateAndPlaceOfBirth
Height
Biography
Biographer
DateAndCauseOfDeath
Trivia
PersonalQuotes
Trademark
WhereAreTheyNow
WrittenAbout
Books
(StaffId, Book)
Paid
Salaries
(StaffId, Salary)
Employer
Called
Nicknames
(StaffId, Nickname)

# Cleaning

- Tools:
  - Python, Regex
- Pipeline:
  - Create new csv, and set object “added”
  - For each row
    - Check if key already added before
    - Flatten and filter values
    - Add cleaned row in csv and key in added

# Cleaning Cont.

- Creating Keys for staff
  - Create map (Name → StaffId) and set of names
  - Run through all csv tables for Staff
  - If Name not in set, add (Name → Incr. Value) to map
- Renders adding foreign keys trivial
- No duplicate keys
- Do the same for Languages, Countries and Genres

# Query Results

- 3.a 10 rows in 16.3 s
- 3.b 15 rows in 1.3 s
- 3.c 2086 rows in 830 ms
  - With indexing: 2086 rows in 476 ms
- 3.d 372 rows in 1.3 s
- 3.e 17 rows in 5.4 s
- 3.f 10 rows in 626 ms
- 3.g 17187 rows in 2.4 s
- 3.h 151175 rows in 7s
  - With indexing: 151175 rows in 6.9s
- 3.i 1 row in 495 ms
- 3.j 3 rows in 33 s
- 3.k 1 row in 344 ms
- 3.l 1 row in 697 ms
  - With indexing: 1 row in 415 ms

# Implementation

- PostgreSQL database (locally)

- Web based interface (locally)
  - Front-end (HTML, CSS, JS)
  - Back-end (Node.js)

# Demo

http://localhost:8080/