

SISTEMAS INFORMÁTICOS I

PRÁCTICA 3

AUTORES

CARLOS GARCÍA SANTA
EDUARDO JUNOY ORTEGA

SISTEMAS INFORMÁTICOS
GRUPO1321: PAREJA 01

INGENIERÍA INFORMÁTICA
ESCUELA POLITÉCNICA SUPERIOR
UNIVERSIDAD AUTÓNOMA DE MADRID



17/12/2023

Índice

Índice.....	2
Introducción.....	2
1.1) Base de datos documental.....	3
1.2) BBDD basadas en Grafos.....	4
2) Uso de tecnología caché de acceso rápido.....	5
Transacciones.....	5

Introducción

Este documento resume los conocimientos adquiridos y aplicados durante la tercera práctica, enfocándose en examinar las diversas opciones ofrecidas por los distintos tipos de sistemas de bases de datos, incluyendo las bases de datos documentales y las basadas en grafos y de acceso rápido. **IMPORTANTE:** se adjunta un fichero sql de las consultas ejecutadas con las que se ha comprobado la corrección de los resultados.

1.1) Base de datos documental

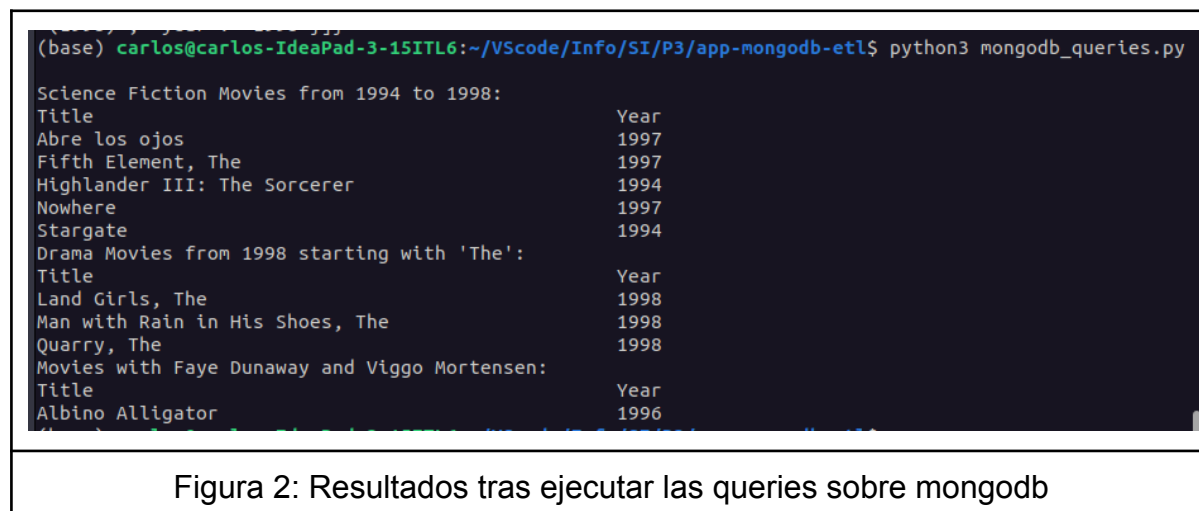
Hemos ejecutado el siguiente comando para poder comprobar que se han guardado los documentos en la colección correctamente.

```
france
france2
france3
si1> db.france3.find()
[
  {
    _id: ObjectId('657f616d8123ce8ef54c7f49'),
    title: 'Ran',
    genres: [ 'Action', 'Drama' ],
    year: 1985,
    directors: [ 'Kurosawa, Akira' ],
    actors: [
      'Amada, Masuo',      'Igawa, Hisashi',
      'Ito, Toshiya',      'Kato, Kazuo (I)',
      'Katô, Takeshi (I)', 'Kawai, Hanbei',
      'Kimura, Sakae',     'Kodama, Kenji (I)',
      'Matsui, Norio',     'Nagasawa, Ryo',
      'Nakadai, Tatsuya',  'Nezu, Jinpachi',
      'Nomura, Mansai',    'Oki, Ryojiro',
      'Peter (I)',         'Ryu, Daisuke',
      'Sakurai, Masaru',   'Sasaki, Masaaki',
      'Sugizaki, Akihiko', 'Suzuki, Heihachiro',
      'Tazaki, Jun',       'Terao, Akira',
      'Ueki, Hitoshi',     'Watanabe, Takashi (V)',
      'Yamaguchi, Yoshimitsu', 'Yamashita, Tetsuo',
      'Yui, Masayuki',     'Zushi, Takao',
      'Zushi, Yoshitaka',  'Harada, Mieko',
      'Kanda, Tokie',      'Kochi, Sawako',
      'Miyazaki, Yoshiko', 'Nanjo, Reiko',
      'Otowa, Kumeko',     'Togo, Haruko'
    ],
    most_related_movies: [ { movietitle: 'Boiling Point (1993/I)', year: '1993' } ],
    related_movies: [
      {
        movietitle: "2 ou 3 choses que je sais d'elle (1967)",
        year: '1967'
      },
      { movietitle: 'Alice et Martin (1998)', year: '1998' },
      { movietitle: 'American Virgin (2000)', year: '2000' },
      { movietitle: 'Beefcake (1998)', year: '1998' },
      { movietitle: 'Before the Rain (1994)', year: '1994' },
      { movietitle: 'Belle de jour (1967)', year: '1967' },
      { movietitle: 'Bitter Moon (1992)', year: '1992' },
      { movietitle: 'Blechtrommel, Die (1979)', year: '1979' },
      { movietitle: 'Blood for Dracula (1974)', year: '1974' },
      { movietitle: 'Bonheur, Le (1965)', year: '1965' },
      { movietitle: 'Bure baruta (1998)', year: '1998' },
      { movietitle: 'Bye-Bye (1995)', year: '1995' },
    ]
  }
]
```

Figura 1: Resultados en mongosh tras ejecutar
create_mongodb_from_postgresqldb

Se ha implementado una mejora en cuanto a la impresión de las queries, ya que se requería imprimir todo el documento, pero esto hacía que el resultado fuese muy

complicado de leer, por lo tanto se han establecido dos columnas una que contiene el título de la película y otra que contiene el año. Sin embargo, para los resultados que se piden se ha implementado una versión que cumple con los requisitos exactos.



1.2) BBDD basadas en Grafos

Se ha conseguido satisfactoriamente tanto la base de datos requerida, cuyo resultado podemos ver gracias al comando de APOC, como las consultas solicitadas en cypher.

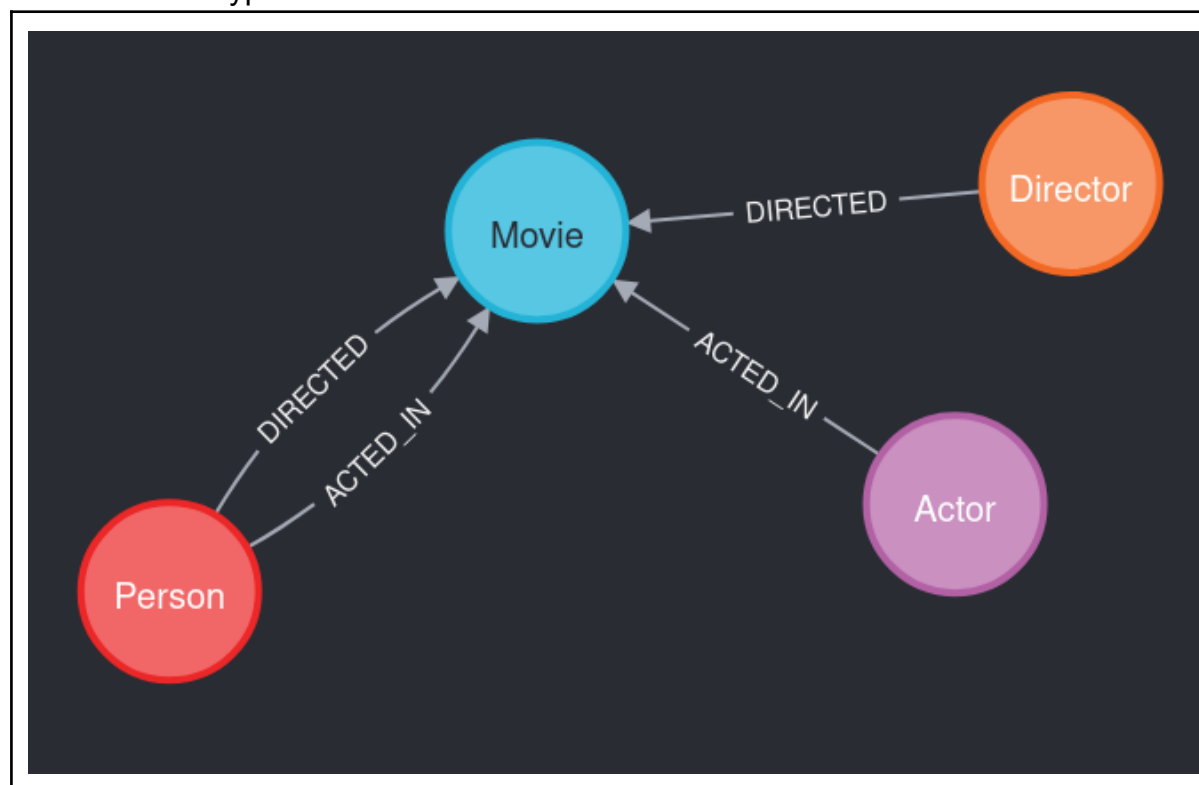


Figura 3: Base de datos en node4j vista con comando APOC en node4j web

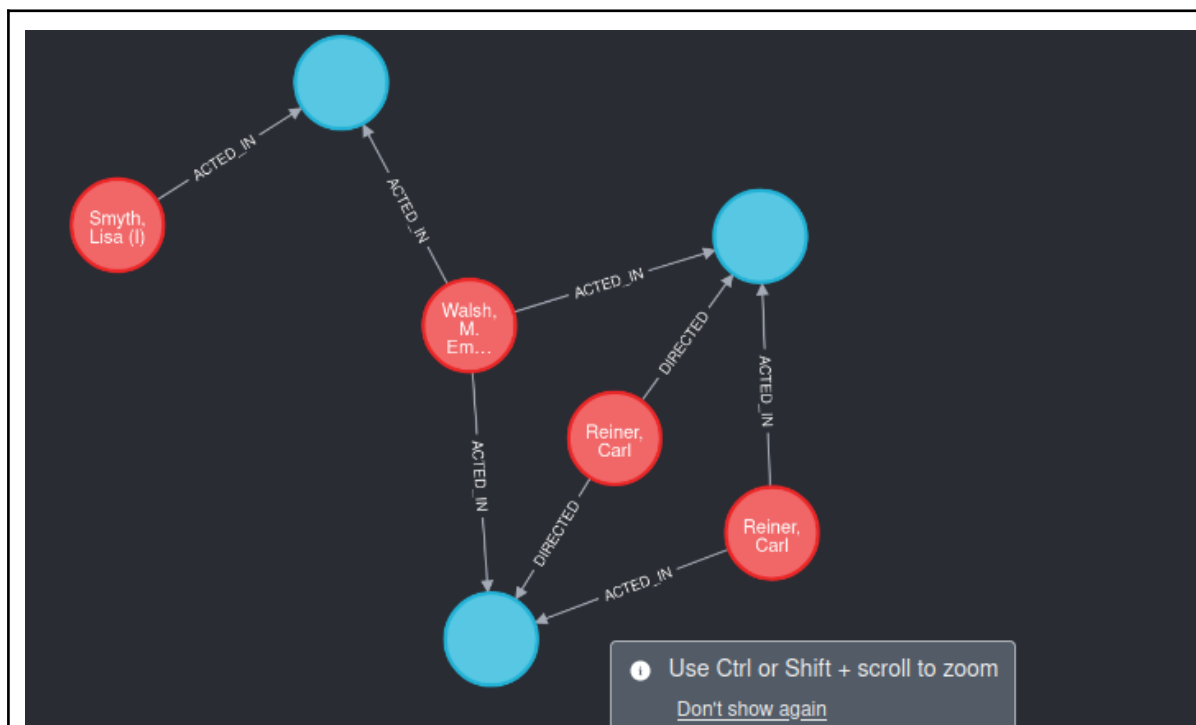


Figura 4: Camino de relaciones obtenido para llegar al objetivo

2) Uso de tecnología caché de acceso rápido

Se crea y maneja correctamente la base de datos Redis, no obstante, cabe notar que se podrían haber usado sets ordenados para una mayor eficiencia en las operaciones, pero por falta de tiempo no se ha llevado a cabo esta implementación.

```
(base) carlos@carlos-IdeaPad-3-15ITL6:~/VScode/Info/SI/P3/app-redis-etl$ python3 create_redis_from_postgresqldb.py
innate.behead@potmail.com
{'name': 'swiss clamor', 'phone': '+5 160438886', 'visits': '85'}
```

Figura 5: Resultado de consultas a base Redis

Transacciones

Una transacción en un sistema de gestión de bases de datos es un conjunto de órdenes que se ejecutan formando una unidad de trabajo, es decir, en forma indivisible o atómica. Cuando por alguna causa el sistema debe cancelar la transacción, empieza a deshacer las órdenes ejecutadas hasta dejar la base de

datos en su estado inicial, como si la orden de la transacción nunca se hubiese realizado.

Para ello existen distintos mecanismos que constituyen una transacción:

- **BEGIN TRANSACTION:** Especifica que va a empezar una transacción.
- **COMMIT TRANSACTION:** Le indica al motor que puede considerar la transacción completada con éxito.
- **ROLLBACK TRANSACTION:** Indica que se ha alcanzado un fallo y que debe restablecer la base al punto de integridad.

En un sistema ideal, las transacciones deberían garantizar todas las propiedades ACID (Atomicity, Consistency, Isolation, Durability).