Brainstorm

Se llama ***seis grados de separación*** a la idea que intenta probar que cualquier **persona A** puede estar conectado a cualquier otra **persona B** del planeta a través de una **cadena de conocidos** que no tiene más de cinco intermediarios (conectando a ambas personas con sólo seis enlaces).

Persona A > conocido 1 > conocido 2 > … > conocido N > persona B

Si esto ocurre en general, debe ocurrir también entre actores a través de una cadena de compañeros de reparto a lo largo de las películas en las que han intervenido. Este enfoque nos habla de que no solo son importantes los datos (titulo de una película y los miembros del reparto y del equipo) sino **las relaciones entre ellos, las cuales se convierten en un dato más**.

Esta idea puede convertirse a su vez en varios productos: juegos de mesa, juego para móvil mono o multijugador, un formato de juego/concurso para televisión, …

Los juegos pueden tener varios niveles de dificultad:

* Básico: solo con actores principales y directores.
* Medio: actores principales, directores y actores secundarios (o de reparto).
* Avanzado: con todo el elenco (actores principales y secundarios, director, productores, director de fotografía, vestuario, director VFX …..)

Para este enfoque necesitaremos una base de datos de películas donde la **tabla de hechos** debe contener al menos: título de la película y elenco completo (nombre de la persona y rol en la peli, por ejemplo:

* Título: “Larry Crowne, nunca es tarde“
* Dirección: [Tom Hanks](https://www.filmaffinity.com/es/search.php?stype=director&sn&stext=Tom%20Hanks)
* Guion: Tom Hanks, Nia Vardalos
* Música: James Newton Howard
* Fotografía: Philippe Rousselot
* Reparto: [Tom Hanks](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Tom%20Hanks), [Julia Roberts](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Julia%20Roberts), [Bryan Cranston](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Bryan%20Cranston), [Nia Vardalos](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Nia%20Vardalos" \o "Nia Vardalos), [Pam Grier](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Pam%20Grier" \o "Pam Grier), [Jon Seda](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jon%20Seda), [Cedric the Entertainer](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Cedric%20the%20Entertainer" \o "Cedric the Entertainer), [Tom Budge](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Tom%20Budge), [Holmes Osborne](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Holmes%20Osborne), [Rami Malek](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Rami%20Malek), [Bob Stephenson](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Bob%20Stephenson), [Rita Wilson](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Rita%20Wilson), [Wilmer Valderrama](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Wilmer%20Valderrama), [Taraji P. Henson](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Taraji%20P.%20Henson" \o "Taraji P. Henson), [Julia Cho](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Julia%20Cho), [Gugu Mbatha-Raw](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gugu%20Mbatha-Raw" \o "Gugu Mbatha-Raw), [Randall Park](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Randall%20Park), [Alex Quijano](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Alex%20Quijano), [Roxana Ortega](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Roxana%20Ortega), [Grace Gummer](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Grace%20Gummer), [Dale Dye](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Dale%20Dye), [Claudia Stedelin](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Claudia%20Stedelin), [Barry Sobel](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Barry%20Sobel), [Rob Riggle](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Rob%20Riggle), [Sy Richardson](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Sy%20Richardson" \o "Sy Richardson)

Los juegos podrían referirse a cine estadounidense, cine norteamericano (USA y Canadá), cine de américa del sur, cine europeo, cine oriental…. En este caso la **tabla de hechos** debería, además de lo anterior, contener el **país**.

Los juegos podrían ampliarse incluyendo también documentales y series de televisión …..

A partir de esta base de datos mínima, se pueden ampliar los datos recogidos:

* Año de estreno.
* Género.
* Localizaciones usadas para el rodaje.
* Premios obtenidos.
* Título de las canciones que componen la banda sonora.
* Intérprete de las canciones de la banda sonora.
* Audio de la banda sonora.
* Fotos de las personas del elenco.
* Fotos del cartel de la película.
* Productora.
* Distribuidora en cada país.
* Datos de taquilla (recaudación).
* Datos de crítica.
* Sinópsis.
* Trailer (el archivo de vídeo).
* Conversaciones en Twitter sobre la peli.

Y con esta ampliación de datos se podrían ofrecer servicios más avanzados de análisis basados en NLP e IA, a modo de ejemplo:

* ¿Se puede predecir la recaudación de una peli en función de su sinopsis, critica, texto del trailer, colores usados en la fotografía del trailer, banda sonora, productora, colores usados en el cartel?
* ¿Se puede predecir la crítica de una peli en función de alguno de estos factores?
* ¿Hay correlación entre la crítica y la recaudación?
* ¿Tienen algo en común las 500 pelis más taquilleras de la historia?

Diseño del DAaaS

### Definición la estrategia del DAaaS

Me centro en el siguiente objetivo: Dados dos actores de cine, dos directores o un actor y un director, quiero disponer de un servicio web para España que proporcione el numero de grados de separación entre ellos, y que además proporcione la cadena de compañeros de reparto (actores y/o directores) que les une y las películas en las que han coincidido.

Por ejemplo, elegimos Melina Mercouri y Tom Hanks y el servicio nos devuelve:

Entre Melina Mercouri y Tom Hanks hay 4 intermediarios, o 5 enlaces que son:

* Melina Mercouri coindice con Jules Dassin en “Topkapi”
* Dassin coincide con Tatum O'Neal en “Círculo de dos”
* O´Neal coincide con Jennifer Aniston en [“Lío en Broadway](https://www.filmaffinity.com/es/film467579.html)“
* Jennifer Aniston coincide con Julia Roberts en “Feliz día de la madre”
* Julia Roberts coincide con Tom Hanks en “Larry Crowne, nunca es tarde“

### Arquitectura DAaaS

Los elementos que voy a necesitar son:

* **Dataset de partida:**

La página web más completa sobre datos de cine es IMDb (Internet Movie Database) (www.imdb.com). Esta página proporciona un dataset con un total de 7 archivos. Este data set lo actualiza IMDb cada día.

Cada conjunto de datos está contenido en un archivo con formato de valores separados por tabuladores (TSV) comprimido con gzip en el conjunto de caracteres UTF-8. La primera línea de cada archivo contiene encabezados que describen lo que hay en cada columna. Un '\N' indica que un campo en particular falta o que es nulo para ese título/nombre.

Los archivos son:

1. **title.akas.tsv.gz**: básicamente contiene datos de los títulos (con indicador único alfanumérico), si se trata o no del título original, el idioma de ese título y región del mundo en que aparece con ese título.
2. **title.basics.tsv.gz**: contiene para cada título info de formato (cine, corto, serie de TV…), año de estreno, duración y género.
3. **title.crew.tsv.gz**: para cada título nos dice el director y el escritor.
4. **title.principals.tsv.gz**: datos del reparto y del equipo (con indicador único alfanumérico) por título, junto con info de su rol en ese título y en el caso de actores el nombre del personaje en ese título
5. **title.ratings.tsv.gz**: contiene la calificación de IMDb y la información de votos para los títulos
6. **name.basics.tsv.gz**: para cada persona contiene el año de nacimiento y fallecimiento, las 3 profesiones principales de la persona y títulos por los que la persona es conocida
7. **title.episode.tsv.gz**: datos sobre episodios de series de TV

De ellos para el objetivo que me he planteado solo necesito los cuatro primeros datasets. El uso requiere reconocer la fuente de los datos incluyendo la siguiente declaración:

Información cortesía de IMDb (https://www.imdb.com).  
Usado con permiso.

* **Un descompresor de archivos.**
* **Un bucket en google storage para guardar el dataset.**
* **Un Hadoop con Talend que arranque automáticamente y que transforme el dataset al formato csv y lo coloque en el bucket.**
* **Un Hadoop con una base de datos orientada a grafos (Neo4j).**

Neo4j es una base de datos orientada a grafos escrita en Java que sigue el modelo property graph o grafo de propiedades en los que tenemos los datos en formato clave/valor, almacenando datos en los nodos y relaciones direccionales. En este caso los nodos son los actores y directores y las aristas o relaciones son las pelis en las que han intervenido.

Por ejemplo para la película “Topkapi”, dirigida por Jules Dassin y con un reparto formado por: Melina Mercouri, Peter Ustinov, Maximilian Schell, Robert Morley, Jess Hahn, Gilles Segal y Akim Tamiroff sería algo del tipo:

{

(Melina Mercouri, Jules Dassin, Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

(Jules Dassin, [Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), Topkapi),

(Jules Dassin, [Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), Topkapi),

(Jules Dassin, [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

(Jules Dassin, [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

(Jules Dassin, [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

(Jules Dassin, [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

([[Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell)v](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([[Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley)](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi)

}

Neo4j se integra perfectamente con múltiples lenguajes como Java, PHP, Ruby, .Net, Python, Node, Scala, etc. La base de datos está embebida en un servidor Jetty. Está especialmente indicada para modelar redes sociales y sistemas de recomendación.

* **Un proceso basado en Cypher (Lenguaje de consultas de Neo4j) que cargue los datos CSV alojados en el bucket en Neo4j.**
* **Un hosting web para alojar una página web dinámica que recoja los nombres de los dos actores con los que queremos jugar, proporcionados por un usuario externo.**
* **Un proceso basado en Cypher que calcule el camino más corto (camino crítico) entre los dos actores y proporcione los nodos (actores y directores) y aristas (películas) de ese camino crítico. Este algoritmo ya está contenido en neo4j.**
* **Un proceso que transforme el resultado anterior a formato html.**
* **Un proceso que envíe ese html a la página web.**

### DAaaS Operating Model Design and Rollout

La operativa será la siguiente:

1. Cada viernes a última hora de la mañana una persona se descarga en local los archivos de la data set de IMDb que nos interesan (title.akas.tsv.gz, title.basics.tsv.gz, title.crew.tsv.gz y title.principals.tsv.gz) y los descomprime obteniendo archivos TSV. Aunque la plataforma IMDb actualiza el data set cada día, como estoy haciendo el diseño para ser usado en España y en España se estrenan las películas los viernes, hacemos la extracción de datos solo ese día a la semana.
2. Esa persona sube los archivos TSV a un bucket en GCloud.
3. Mediante Cloud Scheduler, el viernes por la noche se dispara automáticamente un proceso en Talend que:
   1. Lee el TSV del bucket y lo transforma en CSV
   2. A partir del csv genera una tabla csv tipo [nodo, arista] como explico más abajo.
   3. Coloca dicha tabla en el bucket que actuará de Data Lake (de modo que podamos disponer de todos los datos para futuros posibles usos por el servicio web).
   4. Borra el data set que está en formato TSV
4. Mediante Cloud Scheduler el sábado por la mañana lanzo una función o un script alojado en Hadoop, que cargue los datos CSV en Neo4j (Neo no requiere que le proporcione una estructura de base de datos previa, sino que a medida que cargas datos se crea la estructura automáticamente).
5. Cuando alguien a través de la pagina web introduzca los datos de dos actores/directores mediante Cloud Function se ejecuta en Neo4j alojado en Hadoop el algoritmo de cálculo del camino más corto entre ambos actores/directores. Y Hadoop devuelve el resultado en un formato html a la web app.

### Desarrollo de la plataforma DAaaS.

**Sobre el proceso de transformación de los datos bajados de IMDb:**

De todo el dataset que proporciona esta plataforma solo necesito 4 archivos:

* title.akas.tsv: básicamente contiene datos de los títulos (con indicador único alfanumérico), si se trata o no del título original, el idioma de ese título y región del mundo en que aparece con ese título.
* title.basics.tsv: contiene para cada título info de formato (cine, corto, serie de TV…), año de estreno, duración y género.
* title.crew.tsv: para cada título nos dice el director y el escritor.
* title.principals.tsv: datos del reparto y del equipo (con indicador único alfanumérico) por título, junto con info de su rol en ese título y en el caso de actores el nombre del personaje en ese título

El proceso de transformación debería hacer lo siguiente:

* De title.akas.tsv obtengo una tabla de títulos para españa
* Filtro esa tabla y me quedo con aquellos registros que sean de cine (eliminando series de TV, cortometrajes, etc..)
* Mediante un join con title.crew.tsv añado a la tabla el dato del director.
* Mediante otro join con title.principals.tsv añado a la tabla los actores

De tal forma que dispusiéramos de un csv del tipo:

[ titulo, director, actor1, actor2, … , actorN]

Y finalmente ese archivo csv debe ser convertido a un formato más adaptado a la carga en Neo4j como, por ejemplo:

[

[nodos, aristas]

[director1, peli1]

[actor1, peli1]

[actor2, peli1]

[actor3, peli1]

[director1, peli2]

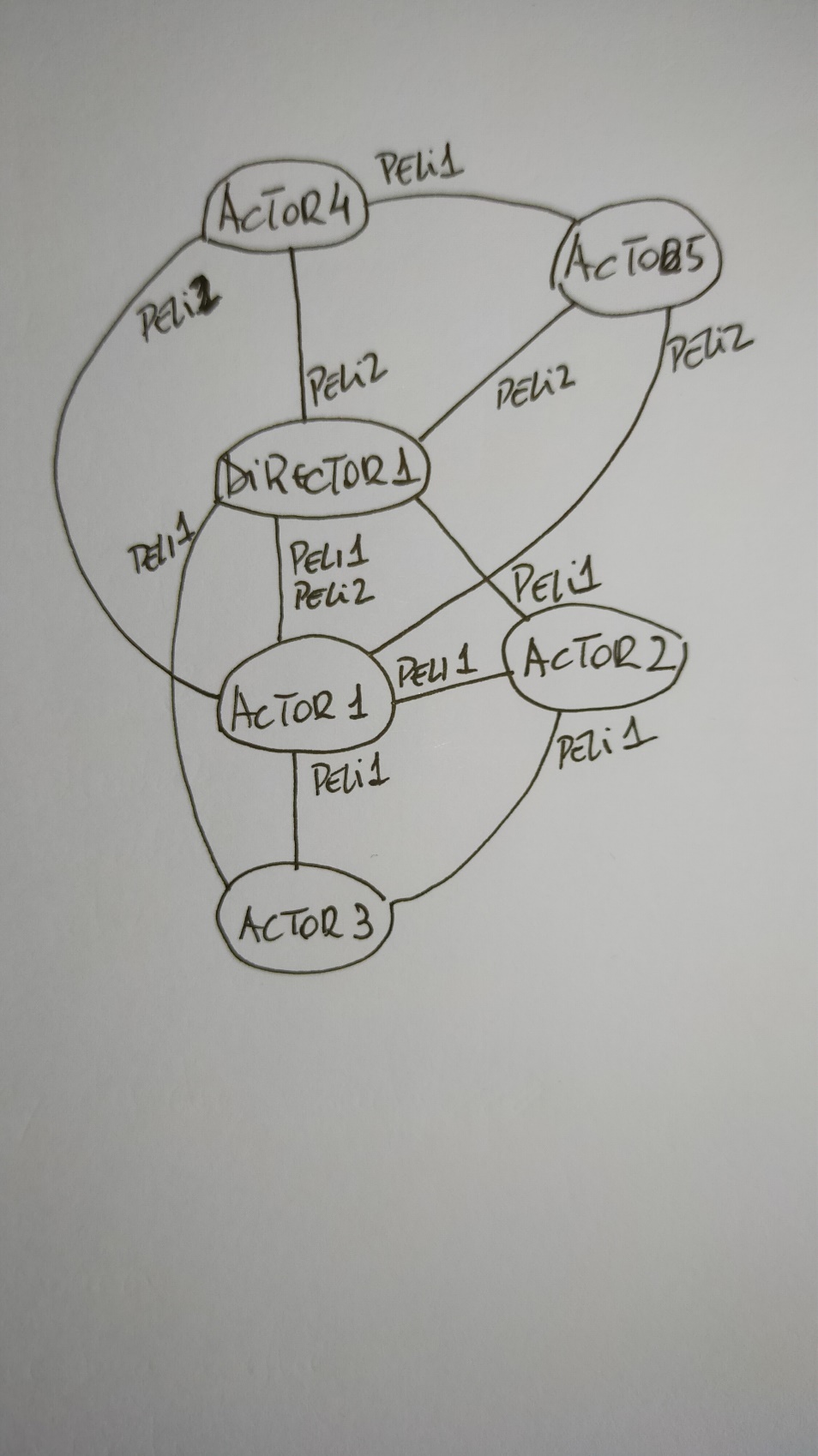
[actor4, peli2]

[actor5, peli2]

[actor1, peli2]

]

El grafo de esa ultimo csv sería algo así como:



**Sobre Neo4j:**

Neo4j es una base de datos orientada a grafos escrita en Java que sigue el modelo property graph o grafo de propiedades en los que tenemos los datos en formato clave/valor, almacenando datos en los nodos y relaciones direccionales. En este caso los nodos son los actores y directores y las aristas o relaciones son las pelis en las que han intervenido.

Por ejemplo para la película “Topkapi”, dirigida por Jules Dassin y con un reparto formado por: Melina Mercouri, Peter Ustinov, Maximilian Schell, Robert Morley, Jess Hahn, Gilles Segal y Akim Tamiroff sería algo del tipo:

{

(Melina Mercouri, Jules Dassin, Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Melina Mercouri](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Melina%20Mercouri), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

(Jules Dassin, [Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), Topkapi),

(Jules Dassin, [Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), Topkapi),

(Jules Dassin, [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

(Jules Dassin, [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

(Jules Dassin, [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

(Jules Dassin, [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Peter Ustinov](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), Topkapi),

([[Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell)v](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Peter%20Ustinov), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Maximilian Schell](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), [Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), Topkapi),

([Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([[Robert Morley](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Robert%20Morley)](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Maximilian%20Schell), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi),

([Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), [Gilles Segal](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Gilles%20Segal), Topkapi),

([Jess Hahn](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Jess%20Hahn), [Akim Tamiroff](https://www.filmaffinity.com/es/search.php?stype=cast&sn&stext=Akim%20Tamiroff), Topkapi)

}

Neo4j se integra perfectamente con múltiples lenguajes como Java, PHP, Ruby, .Net, Python, Node, Scala, etc. La base de datos está embebida en un servidor Jetty. Está especialmente indicada para modelar redes sociales y sistemas de recomendación.

**Sobre el algoritmo de Dijkstra para encontrar el camino mas corto entre dos nodos de un grafo:**

El algoritmo es bastante simple. Llamaremos nodos a cada uno de los actores/directores y en nuestro caso del cine, la distancia es en todos los casos igual a 1 entre dos personas que intervienen en una peli dada.

Consiste en los siguiente:

1. Marque todos los nodos como no visitados y guárdelos.
2. Establezca la distancia en cero para nuestro nodo inicial y en infinito para otros nodos.
3. Seleccione el nodo no visitado con la distancia más pequeña, ahora es el nodo actual.
4. Encuentre vecinos no visitados para el nodo actual y calcule sus distancias a través del nodo actual. Compare la distancia recién calculada con la asignada y guarde la más pequeña. Por ejemplo, si el nodo A tiene una distancia de 6 y el borde AB tiene una longitud de 2, entonces la distancia de B a través de A será 6 + 2 = 8. Si B se marcó previamente con una distancia mayor que 8, cámbiela hasta 8.
5. Marque el nodo actual como visitado y elimínelo del conjunto no visitado.
6. Deténgase, si se ha visitado el nodo de destino (al planificar una ruta entre dos nodos específicos) o si la distancia más pequeña entre los nodos no visitados es infinita. Si no es así, repita los pasos 3 a 6.

Diagrama:

