

# INFO-H-303 : Bases de données

## Projet : *Internet Movie Database (IMDB)*

Professeur : Esteban Zimányi  
Assistant : Michaël C. Waumans  
<http://cs.ulb.ac.be/public/teaching/infoh303>

Année académique 2016-2017

---

### Étude de cas

On vous demande de créer une application de consultation d'informations cinématographiques en utilisant les données fournies par *Internet Movie Database*<sup>1</sup>.

L'application stocke la description d'un ensemble de films et de séries. Pour chaque film, le système retiendra le titre, les acteurs ainsi que leur rôle, le ou les directeurs, le ou les auteurs, le ou les genres, la langue originale, le pays, l'année de création et leur note. Pour chaque série, on retiendra en plus son éventuelle année de fin ainsi que la liste des épisodes avec pour chaque épisode, son titre, son numéro de saison, son numéro d'épisode, ses acteurs et leur rôle. Pour chaque acteur, on enregistrera son nom, prénom et genre.

Le système acceptera deux types d'utilisateurs : les utilisateurs anonymes et les administrateurs. Les utilisateurs anonymes auront uniquement le droit de consultation. Les administrateurs auront en plus les droits d'insertion, de suppression et de modification. Les administrateurs auront un identifiant (leur adresse email) ainsi qu'un mot de passe haché (à l'aide par exemple de SHA-256) et pourront s'authentifier sur le système.

Tous les utilisateurs devront pouvoir consulter facilement toutes les informations présentes dans la base de données. Notamment, ils devront pouvoir :

- Consulter toutes les informations d'un film, d'une série ou d'un épisode d'une série.
- Rechercher des films ou des séries sur base de leur nom, de leur année de création, des personnes y ayant participé, le genre...
- Consulter toutes les informations d'une personne ainsi que tous les films, séries ou épisodes auxquels elle a participé ainsi que son rôle.

### Format des données

Les fichiers que vous devez importer dans votre base de données relationnelle se trouvent à l'adresse <ftp://ftp.fu-berlin.de/pub/misc/movies/database/>. Ces fichiers sont auto-documentés. A vous de choisir quels fichiers sont à importer et quels outils vous utilisez pour extraire les données et les importer dans votre base de données.

Pour des raisons de performance, veillez à n'importer que les données concernant des films, séries, épisodes ayant eu lieu entre 2000 et 2010 inclus.

On vous demande de créer un script d'insertion de ces données dans votre base de données.

---

1. <http://www.imdb.com/>

# Déroulement du projet

## Première partie

Pour cette partie, on vous demande de modéliser le problème à l'aide du formalisme entité-association et de préciser les contraintes d'intégrité nécessaires. Ces contraintes doivent être exprimées en français et utiliser les mêmes noms d'entités, d'associations ou d'attributs que dans votre modèle conceptuel. Vous pouvez également exprimer et justifier des hypothèses sur votre modèle. Ces hypothèses peuvent résulter par exemple d'ambiguïtés dans l'énoncé du problème.

Déduisez ensuite de ce modèle conceptuel le modèle relationnel correspondant ainsi que ses contraintes. Vos choix de modélisation doivent être justifiés.

Ces modèles doivent être suffisamment riches pour pouvoir servir de support à l'application décrite ci-dessous ainsi que pour pouvoir répondre aux requêtes de la deuxième partie.

## Deuxième partie

### Création

On vous demande tout d'abord de déduire de votre modèle relationnel un script SQL DDL de création de la base de données et de ses différentes tables ainsi que de créer cette base de données.

### Initialisation

On vous demande d'écrire un script permettant d'importer dans votre base de données les fichiers nécessaires. Ces données devront être présentes dans votre base de données lors de la défense.

### Application

On vous demande ensuite de développer une interface graphique (web ou client lourd) pour votre base de données permettant au minimum de réaliser les opérations ci-dessous :

- consulter les films, séries et épisodes
- consulter les personnes (acteurs, auteurs, producteurs) et leur rôle dans les films, séries et épisodes
- rechercher des films sur base de leur nom
- rechercher des films sur base de critères avancés comme l'année, un acteur, un producteur, le genre, ...
- afficher les résultats des requêtes demandées (voir plus loin)
- connecter un administrateur. Les administrateurs pourront :
  - enregistrer un administrateur (à l'aide d'un autre compte administrateur)
  - ajouter, supprimer et modifier les données (acteurs, producteurs, films, séries, épisodes ...)

Votre application devra veiller à ce que la base de données reste cohérente en utilisant autant que possible le moteur de base de données (contraintes de clé étrangère, etc.).

Vous pouvez bien sûr ajouter des fonctionnalités à votre application comme par exemple

- analyser le petit monde<sup>2 3</sup> des acteurs.
- un module de jeu de style Quizz (une question et quatre réponses possible)
- voter sur un film, une série ou un épisode
- trier les films selon des critères (notes, date, ...)
- présenter les films sur une carte (par exemple, à l'aide de Google Maps)
- prendre en compte plus de données d'IMDB
- etc.

Ces apports personnels seront valorisés à hauteur de 4 points sur 20.

---

2. [http://fr.wikipedia.org/wiki/etude\\_du\\_petit\\_monde](http://fr.wikipedia.org/wiki/etude_du_petit_monde)

3. <http://oracleofbacon.org/>

## Requêtes

Nous vous demandons d'écrire en algèbre relationnelle et calcul relationnel tuple les requêtes R1, R2, R3 et R4 ainsi que toutes les requêtes en SQL. Si une requête vous semble imprécise, indiquez dans votre rapport les hypothèses que vous avez faites pour lever ces imprécisions. Attention, pour les requêtes en algèbre relationnelle et en calcul relationnel, vous ne pouvez pas utiliser de fonction d'agrégation.

- R1 : Les acteurs qui ont joué toutes les années entre 2003 et 2007.
- R2 : Les auteurs qui ont écrit au moins deux films pendant la même année.
- R3 : Les acteurs  $Y$  qui sont à une distance 2 d'un acteur  $X$ . Un acteur  $Y$  est à une distance 1 d'un acteur  $X$  si ces deux acteurs ont joué dans le même film.
- R4 : Les épisodes de série où il n'y a aucun acteur masculin.
- R5 : Les acteurs qui ont joué dans le plus de séries.
- R6 : Les séries avec leur nombre total d'épisodes, le nombre d'épisodes moyen par an et le nombre d'acteurs moyen par saison depuis leur année de création et ce pour toutes les séries dont la note est supérieure à la moyenne des notes des séries.

## Rapports

Pour la première partie, on vous demande les documents suivants :

- un diagramme entité-association modélisant le projet ainsi que ses contraintes
- une traduction relationnelle de ce diagramme et ses contraintes
- vos hypothèses et la justification de vos choix de modélisation

Le formalisme utilisé doit être un de ceux vus au cours ou aux TPs.

Pour la deuxième partie, vous rendrez un rapport contenant :

- les documents de la première partie tenant compte des remarques des assistants
- une explication de votre méthode d'extraction des données
- les requêtes demandées
- les explications et justifications de vos choix et hypothèses

## Informations pratiques

- Le projet se fera **obligatoirement** par groupe de deux ou trois personnes.
- La première partie, en version PDF, devra être rendue par email à [mwaumans@ulb.ac.be](mailto:mwaumans@ulb.ac.be) pour le 27 Mars. Nous en discuterons en TP le lendemain, 28 Mars.
- Le rapport de la seconde partie devra être rendu en version PDF. Une archive contenant tous les codes sources devra être envoyée à l'adresse [mwaumans@ulb.ac.be](mailto:mwaumans@ulb.ac.be) pour le 18 Mai à 23 heures 59. La défense du projet aura lieu la semaine suivante suivant un horaire à déterminer. On vous demandera de présenter votre application pendant 10 minutes et de répondre à quelques questions. Le temps d'une défense sera donc de maximum 15 minutes, veillez donc à ce que votre démonstration ne déborde pas des 10 minutes allouées.
- Le projet comptera pour 25% de la note finale du cours.
- Les apports personnels au projet seront valorisés à hauteur de 4 points sur 20. En d'autres mots, un projet parfait sans apport personnel aura une valeur de 16 points sur 20.
- Sauf mention explicite, vous pouvez utiliser les langages et outils de votre choix (MySQL, PostgreSQL, Python, PHP, Java, ...).
- Vous ne pouvez pas employer SQLite!
- Vous pouvez développer sur votre propre machine et présenter vos projets sur un ordinateur portable que vous apporterez lors de la défense.

Bon travail!