

Bases de Données Avancées

<>

ToDo

Il faut ajouter les cardinalités

Questions pour LiFang

Comment insérer une zone de texte sur une ligne ou une figure ? Double clic

Où est-ce qu'on écrit la liste des valeurs possibles pour un champs ?

Exemple : Skills[skill_type] = job, instrument, language, style

1. Est-ce qu'on peut faire des contraintes sur les valeurs ? Que ce soit supérieure à 0 ou un choix dans une liste de string ?

I. TABLES

2. A partir de quel moment y a-t-il une table supplémentaire lorsque 2 tables sont liées ? Si les 2 tables ont des cardinalités supérieures à 1 ? Ou bien est-ce suffisant si c'est le cas d'une seule d'entre elles ?

3. Comment représenter sur le schéma E/R les tables de relation (i.e. une table représentant la relation entre 2 autres tables qui ont des cardinalités supérieures à 1 sur leur lien). Ajoute-t-on un rectangle comme pour les autres tables ? Un losange ? Si on ne met rien, comment sait-on le nom de la table ?

Différence entre entité faible et simple relation

Relation (lien simple, entre 2 table) : la clé primaire est les clés primaires des autres tables

Relation faible (lien double) : la clé primaire est les clés primaires des autres tables + un autre champs (pas unique sinon ce n'est plus une entité faible)

Les cardinalités sur les autres tables sont forcément 1-1

5. Comment appelle-t-on une relation dans laquelle la clé primaire de la nouvelle table est :

- les clés primaires des autres tables de la relation

- plus un autre champs UNIQUE

Ce n'est pas une relation faible car le champs en plus est unique. Ce n'est pas une simple relation car on a ajouté un champs.

Relation de spécialisation (héritage dans schéma E/R) : une table découlant d'une autre (par exemple artiste découlant de Contact) et personne d'autre que le parent n'est lié avec l'enfant.

En termes de tables (schéma relationnel), il faut éliminer les spécialisations. Il y a plusieurs méthodes, cf. transparents de cours de L3

Schémas entités/reliations

schémas relationnel : liste des tables, avec les attributs, les clés primaires, les clés étrangères

Il faut présenter le schéma E/R dans le rapport mais on peut ajouter le schéma relationnel.

Il ne faut pas avoir d'info qui se répètent. Ce n'est pas grave s'il faut faire 4 jointures pour avoir une info.

Par contre on peut mettre des index pour que les jointures aillent plus vite. Si la table est petite, l'index ne sert à rien. Mais pour une table de 1000 lignes et des requêtes qu'on fait souvent, ça peut être utile.

A voir en fonction des requêtes qu'on va faire.

Trigger : seulement si on peut pas faire autrement

Avenant, reviews, awards, festivals doivent être des tables à part

Que sont FK et PK ?

Est-ce problématique d'avoir une table où des infos se répètent ?

Une solution simple est de ne pas répéter les info et de faire une table pour faire la jointure

Par exemple des tables : skills, artists, et une table de jointure entre les 2

Faut il justifier que notre schéma est FNBC ou 3NF ou 4NF ? Pas nécessaire. Peut être utile pour nous de faire la liste des contrainte fonctionnelles, auquel cas c'est à ajouter en annotation (mais normalement ça ne devrait pas être nécessaire).

Qu'est ce qu'une contrainte ? Une expression mathématique sur les champs ? Ou bien un phrase en français ?

Une contrainte est sur une ou plusieurs tables.

Peut être une phrase en français.

Pour une table, on utilise un CHECK ou un TRIGGER

Pour plusieurs tables, on utilise un TRIGGER (à vérifier).

Pourquoi mettre dans la modélisation des commentaires avec « la table ... contient les champs ... » ? Car cela triple le travail (on met les noms des champs dans la modélisation, les commentaire, et le document tableDetails.txt)

Demands :

- Est-ce qu'il ne vaudrait pas mieux mettre CreationID que CreationName ? Je n'arrive pas à me décider.
- Est ce qu'il faut indiquer le champs CreationID dans la modélisation ou bien est-ce sous entendu par le lien entre la table Demand et la table Creations ?

De Proposition à ContratArtProd, est-ce que ça ne devrait pas être 0-1 (au lieu de 0-n) ? Car une proposition ne peut donner qu'un contrat au plus.

ContratArtProd :

- Est ce qu'on ne devrait pas ajouter des champs pour l'id de l'artiste, l'id du producteur, l'id de la creation et l'id de la demande (ou de la proposition) ?

Comptabilite :

- Est-ce qu'il ne faut pas faire apparaître dans la modélisation le lien entre la table Comptabilite et les tables ContractArtAgency et ContractArtProd ?
- Est ce qu'il ne serait pas préférable de remplacer ContratAAID par PercentageCommissionAgency ? Car c'est la seule info du contrat dont on ait besoin, il n'y a pas vraiment besoin du reste.

1. CONTACTS

On fusionne les tables *artists* et *contacts*. On ajoute un champs *societe* indiquant à quelle société la personne appartient. Ce champs doit valoir « agency » dans le cas des artistes (représentés par l'agence).

awards, *review*, *festivals* sont des zones de texte avec toutes les info dedans (dates, nom, etc.). Cela permet de ne pas multiplier le nombre de tables. Seul le champs *festivals* ne semble pas utile pour les contacts extérieurs à l'agence. *awards* permet d'indiquer si cette personne a déjà reçu des récompenses et *review* de mettre les feedbacks des précédentes coopérations avec elle.

Pour savoir si un contact est une personne représentée par l'agence, on fait une jointure sur `historyRepresent WHERE dateEnd = null ou dateEnd > now()`

2. TRAININGS

Si un artiste a suivi plusieurs formations, elles seront toutes listées dans la table *trainings*.

Les informations se répètent si 2 artistes ont la même formation. Mais il y a peu d'informations donc ce n'est pas très grave.

Cela permet en revanche de ne pas avoir à faire une table supplémentaire pour lier artistes et trainings.

DateEnd peut être NULL. Par exemple quelqu'un qui va avoir un diplôme à la fin de l'année pour un projet l'année suivante.

3. Creations

Réalisations précédentes : films réalisés, pièces de théâtre ou spectacles mis en scène, rôles joués dans les films et au théâtre, les albums (pour les musiciens)

Pour un film dans lequel a joué une actrice de l'agence, il serait intéressant d'indiquer dans les infos du film qui en est la réalisatrice. Mais pas pour un film qui a été dirigé par une réalisatrice de l'agence. Comment faire ? Simplement ne pas indiquer la réalisatrice ?

Le champ Profits sert à déterminer quel montant la création a généré pour calculer combien va toucher l'artiste en fonction de sa participation.

On doit mettre à jour ce montant régulièrement. Un champ permet d'indiquer quel est la date de dernière mise à jour des profits. Chaque mise à jour ajoute une ligne dans la comptabilité via un trigger.

update creation[profit] =>

TRIGGER BEFORE UPDATE ON CREATION.profit qui ajoute une ligne sur la comptabilité avec

l'amount(new) - amount(old) (ex : 2012)

Date peut être null dans le cas de créations dont on ne sait pas quand elles verront le jour (on met par contre forcément un nom, quitte à le changer par la suite)

4. Review

Une review concerne forcément 1 œuvre mais peut concerner plusieurs artistes.

Donc on lie une review à une œuvre et un artiste pour que ce soit facile de retrouver les reviews concernant un artiste particulier.

On aurait pu considérer qu'une review concerne une œuvre en entier (pas spécifiquement un artiste) et donc enlever le lien entre ReviewRecords et Contact.

5. Awards

Lié à une créations.

De plus, soit un award est lié à une personne (e.g. meilleur acteur), auquel cas, on indique l'id de cette personnes.

Soit l'award n'est pas lié à une personne en particulier (e.g. meilleur album) et dans ce cas on indique NULL dans le champs contact_id.

Pour trouver tous les awards d'un contact, on fait une union de 2 jointures (une entre Contact/Award, une entre Contact/involvement/Creations/awards)

6. Festivals

Plusieurs artistes peuvent avoir participer au même festival

7. HISTORY

La table *history* permet d'indiquer l'historique des agents ayant représenté un artiste. Cela aurait pu également se faire par le biais d'une table *contractAA* entre un agent et un artiste mais les contrats passés sont entre les artistes et l'agence elle-même (et non un agent particulier).

8. ContractAA

Si DateEnd = Null, c'est que le contrat est toujours en cours

9. ContractAP

Il n'y a pas de lien direct entre l'agent et le contrat passé entre un artiste et un producteur. En effet, les agents n'ont pas d'intéressement, c'est l'agence qui est payée directement. Si on a besoin de connaître l'agent, on peut faire une jointure avec la table *history* ce qui est plus coûteux mais devrait rarement arriver.

Remuneration est un champs different de Demand[Budget] car le montant peut être négocié.

De même, la participation n'apparaît pas dans la demande car cela dépend beaucoup de l'artiste choisi et est donc négocié à la signature du contrat.

On a ajouté id (contrairement à la table contractAA) car on en a besoin pour les avenants.

5. Tous les nombres sont des doubles. Simplifie les calculs (pourcentage par exemple). Mais cela prend plus de place en mémoire.

6. « L'agent met en relation les producteurs et les réalisateurs ou metteurs en scène. Les musiciens peuvent être mis en relation avec les producteurs ou avec les studios. L'agence peut aussi mettre en relation les artistes (par exemple comédiens et comédiennes) avec les directeurs de casting, avec les réalisateurs ou metteurs en scène (ces derniers représentent peut-être par une autre agence). »

Est ce que ce sont là des contraintes (i.e. il faut qu'on ajoute des contraintes à notre schéma) ou bien juste des exemples pour notre information

Que met on dans le champ Avenant ?

Du texte décrivant les modifications ? Dans ce cas là, une proposition est liée à 0-1 contrat (au lieu de 0-n)

Pour retrouver toutes les info (nom de l'artiste, nom du producteur, nom de la piece, etc.), il faut faire au moins 4 jointures. Est ce qu'il ne serait pas préférable de découpler ContractAP et Proposition et d'indiquer dans ContractAP les infos nécessaires (artistID, producerID, CreationID...) ?

10. Demands

Faut-il lier chaque entrée de la table *demand* à une entrée de la table *creation* ?

Cela permet de lier plusieurs demandes liées à une même creation entre elles

Mais peut nécessiter de créer une "création" bien qu'on n'ai pas encore le titre et les autres infos.

11. Proposition

On met un lien avec la table *agence* afin de pouvoir vérifier les « performances » des agents.

12. Comptabilite

La table *comptabilite* a un champs statut. Cela permet d'indiquer les versements prévus lors de la signature du contrat (statut « pending »), puis ce qui a réellement été payé (status « done »), quitte à apporter des modifications aux montants et dates.

Il serait possible d'arguer que les dates prévus pour les paiements devraient être indiqués dans une table à part, tandis que la table *comptabilite* ne devrait contenir que les paiements effectivement effectués. Nous préférons cependant éviter cette solution afin de limiter le nombre de tables.

Nous avons opté pour la création d'une seule table plutôt que d'avoir une table pour la comptabilité des artistes et une pour celle de l'agence. A partir du montant versé, on déduit la part de l'agence ainsi que celle de l'artiste en fonction du pourcentage prévu dans le contrat correspondant.

C'est une simplification de la modélisation par rapport à la réalité où les versements sur les comptes bancaires de l'agence et de l'artiste ne sont probablement pas faits au même moment.

Le montant versé à l'artiste et celui versé à l'agence est calculé grâce à un trigger qui récupère le pourcentage de commission de l'agence dans le contrat entre artiste et agence.

Le

a) **Constantes de classe**

b)

c)

II. PORTES

1. Classe Gate

2. Entrées / Sorties

3. Portes logiques

a) Constantes de classe

b)

c)