

# Définir la structure de données

Rappel : dans un système de type base de données, l'information réside autant dans les "objets" (ou "entités") qui jouent un rôle distinct au sein du système d'information que dans la mise en relation de certains objets entre eux (liens).

L'analyse préalable à l'élaboration d'une base ne va pas de soi. Quand on ne dispose pas d'un outil de modélisation spécialisé, il faut, en premier lieu, mettre à plat sur le papier l'organisation des données, sans tenir compte à cette étape de la façon dont elle se traduira concrètement dans le logiciel que l'on utilise : ce "schéma conceptuel" doit donner une représentation *purement logique* des "objets" qui entrent en jeu dans le système d'information et des liens qu'ils entretiennent. Sa traduction en structure de données réelle (tables et liens) n'interviendra que dans un deuxième temps.

Pour vous aider dans cette phase d'analyse, voici un rappel (très condensé et sans trop de jargon spécialisé...) de quelques principes à toujours garder en tête.

## Règles d'or (une table = un type d'objet, une rubrique = une propriété de l'objet)

- Chaque table doit héberger les informations correspondant à un seul type d'"objet" ou "entités";
- Chaque rubrique d'une table doit correspondre à une *propriété unique de cet objet* (on ne peut saisir qu'une seule donnée dans la rubrique et l'information correspondante n'est disponible que si l'objet existe).
- La même donnée ne doit jamais être redondante (*i.e.* saisie à deux endroits);

Le non-respect de ces principes entraîne toujours des effets pervers : risque d'incohérence des données, de perte d'informations, travail de saisie inutile...

Il faut donc arriver à cerner le contour de vos objets (au sens d'entités distinctes) : en résumé, cela consiste à regrouper les données qui caractérisent un même objet, en se demandant *de quoi dépend chaque donnée prise en compte*. La difficulté vient de ce que les objets nécessaires à un système informatisé ne recouvrent pas ce que nous percevons comme tel dans la "vraie vie" : des données que perçues spontanément comme faisant partie des propriétés naturelles d'un objet, devront, dans une base de données, être considérées comme des objets distincts et être placées dans une table à part :

### Exemple 1 : dictionnaire du cinéma

La notion de générique recouvre en fait des films et des individus mis en relation (la personne A participe au film B en tant que...)

### Exemple 2 : scolarité à l'Université

La notion d'inscription pédagogique recouvre en fait des étudiants (Objet A) et des EP (Objet B) qui sont mis en relation (A est inscrit à B).

## Comment voir s'il faut une ou plusieurs tables ? Les principaux cas

Il faut considérer que l'on a deux objets autonomes, et donc les mettre dans des tables distinctes, dès que l'une ou l'autre de ces conditions est remplie :

- *L'information A peut être en relation avec plusieurs B ou vice versa* : dans ce cas en effet, réunir A et B dans une même table engendrerait la redondance de l'information

A pour chaque B la concernant, avec à la clé un encombrement inutile et d'inévitables erreurs de saisie...

La question à se poser :

**La donnée A peut-elle correspondre à plusieurs données B ou inversement ?**

Une personne intervient dans plusieurs films / Un étudiant est inscrit dans plusieurs EP / un film a plusieurs personnes à son générique EP accueille plusieurs étudiants

- *On veut pouvoir avoir des infos sur A, même s'il ne correspond à aucun B* : dans ce cas A doit être représenté dans une table à part, sinon il y a risque de perte d'info en cas de suppression (ou de non saisie) de B.

La question à se poser :

**Est-ce que A doit pouvoir exister indépendamment de B (A est un objet autonome) ou bien A n'est-il pertinent que si B existe (A n'est qu'une propriété de B) ?**

Il ne peut y avoir d'information "Date de sortie du film" sans film : "Date de sortie" est donc une propriété intrinsèque du film.

En revanche, un film peut n'avoir aucun acteur (dessins animés, documentaires...), il faut quand même pouvoir mémoriser et retrouver les informations qui le concernent. Les données concernant le film lui-même (titre, genre, etc.) constituent donc un objet indépendant de celles propres aux acteurs.

→ on a besoin de 2 objets : films / acteurs

Il ne peut y avoir d'information "libellé de l'EP" sans EP, la donnée n'est disponible que si l'EP existe : "libellé EP" est donc une simple propriété de l'EP.

En revanche, un EP peut ne pas être rattaché à un diplôme à un moment donné (s'il est pas ouvert une année, enseignements optionnels), mais on a tout de même besoin des informations le concernant (libellé, volume horaire, règlement de scolarité...)

→ on a besoin de 2 objets : EPs / diplômes

## Comment voir à quoi rattacher une propriété ?

Si la plupart des données sont assimilables à des propriétés d'un objet donné (comme le prénom ou l'âge d'une personne), ce n'est pas toujours le cas : certaines informations ne sont pas caractéristiques d'un objet mais de l'association de deux objets distincts :

- *Cas particulier : la donnée est une propriété non pérenne, c'est une information qui caractérise non pas un objet mais le relation entre deux objets.*

La question à se poser :

**Suffit-il que l'objet A existe pour que la propriété P existe ? Ou bien P n'a-t-elle de pertinence que si l'objet A est en relation avec un objet B ?**

La donnée "rôle interprété" (P) n'est propre ni à un film (le même rôle se retrouve dans plusieurs en cas de "remake"), ni à un acteur (il interprète divers rôles au fil du temps) :

Cette information n'est en fait pertinente que par l'association d'un acteur et d'un film (L'acteur A joue le rôle P dans le film B).

L'information "Validé/Échec" (P) n'est propre ni à l'EP (valeur différente selon les étudiants), ni à l'étudiant (valeur différente selon les EP auxquels il est inscrit).

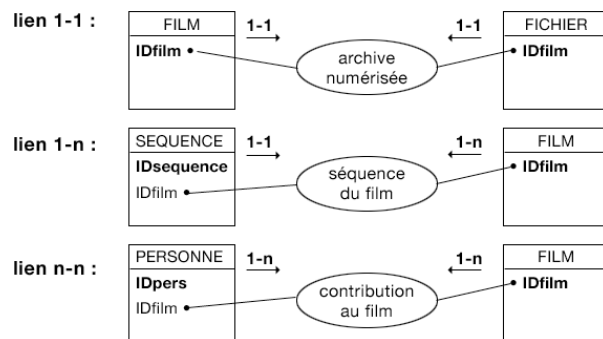
Cette information n'a de sens que par l'association d'un étudiant et d'un EP. (L'étudiant A obtient le résultat P à l'EP C).

## Les liens

### Analyse en amont : caractériser précisément chaque lien

Le travail d'analyse qui aboutit au schéma conceptuel doit aussi permettre de caractériser précisément les liens. Pour chacun d'eux, il faut systématiquement préciser ces points :

- les rubriques (identifiants) de l'une et l'autre table sur lesquelles se fonde le lien,
- la *signification* précise du lien (que veut dire le fait de relier 2 enregistrements de l'une et l'autre table par ce lien ?),
- la *cardinalité maximale* du lien, (*i.e.* combien d'enregistrements peut-il y avoir à chaque bout de la ficelle ?) Rappel : ce que l'on nomme, par facilité langagière, liens 1-1, 1-n ou n-n, recouvre les cardinalités du lien observé dans les deux sens. Exemples :



- bien que ces exemples n'en rendent pas compte, on doit aussi décider de la *cardinalité minimale* du lien<sup>1</sup> ; pour un lien A—B, cela revient à poser la question : chaque objet A doit-il obligatoirement être relié à un objet B ou non ?

Les cardinalités minimales font partie des *contraintes d'intégrités* que l'on peut poser pour garantir la cohérence des données. Ainsi, dans le premier exemple, il faudrait probablement imposer que chaque objet FICHER soit rattaché à un objet FILM.

### Traduction des liens du schéma en structure de données réelle

Dans la base de données, les liens du schéma conceptuel seront réalisés de différentes manières selon leur cardinalité maximale. Le principe :

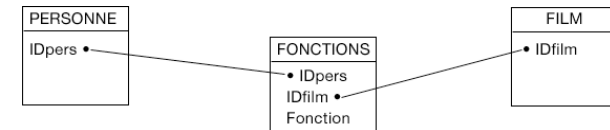
- On peut traduire les liens 1-1 ou 1-n directement sous la forme de lien entre deux tables, il suffit de prévoir dans les deux tables la rubrique de l'identifiant sur lequel repose le lien :

*Pour un lien 1-n où A est lié à plusieurs B, c'est sur l'identifiant de la table située du côté N (B) qu'il faut faire reposer le lien, et à l'autre table (A) qu'il faut ajouter la rubrique destinée à la saisie de l'identifiant "étranger" (celui de B).*

<sup>1</sup> La notation x-x, adoptée ici pour faciliter la lecture, devient alors selon le cas : 1-1 ou 1-0..1, 1-n ou 1-0..n.

Pour le lien SÉQUENCE — (1-n) — FILM de notre exemple, c'est sur l'identifiant du film (côté N) que le lien est fondé, on ajoutera par conséquent à la table SÉQUENCE un champ "IDfilm" pour saisir, dans chaque enregistrement de SÉQUENCE, l'identifiant du film auquel il est rattaché.

- En revanche, *les liens n-n nécessitent l'utilisation d'une table intermédiaire*, table où seront en outre placées les éventuelles propriétés qui caractérisent l'association elle-même (et non l'un ou l'autre des objets liés). Ainsi le lien SÉQUENCE —(n-n)— FILM ("telle personne intervient dans tel film en tant que...") se traduira, dans la base de données, par une table supplémentaire :



Du point de vue des données enregistrées dans les tables, chaque participation à un film fera l'objet d'un enregistrement dans la table FONCTIONS :

Table PERSONNE

IDpers	Nom	Prenom
P004	Godard	Jean-Luc
P008	Visconti	Lucchino
P009	Hitchcock	Emmanuel
P010	Roblès	Emmanuel
P022	Mastroianni	Ruggero
P228	Karina	Anna
P820	Belmondo	Jean-Paul
P444	Blier	Bernard

Table FONCTIONS

IDpers	fonction	IDfilm
P004	Réalisation	F002
P004	Scénario	F002
P228	Acteur rôle A	F002
P820	Acteur rôle B	F002
P008	Réalisation	F014
P008	Scénario	F014
P010	Scénario	F014
P022	Scénario	F014
P228	Acteur rôle X	F014
P444	Acteur rôle Y	F014

Table FILM

IDfilm	Titre
F002	Pierrot le fou
F014	L'Étranger

Cette solution est la seule qui permette de répondre à tous les cas de figure, parce que *la relation se fait par simple ajout d'enregistrements dans la table FONCTIONS*. Cette organisation donne toute liberté :

- de jouer sur un éventail de fonctions non déterminé à l'avance,
- de permettre qu'une personne assume 0, 1 ou plusieurs fonctions différentes dans un ou plusieurs films.

### Implémentation des contraintes issues des cardinalités minimales

La mise en place de contrôles garantissant le respect des contraintes liées aux cardinalités minimales pourra se faire de différentes façons, selon les possibilités du logiciel utilisé : au niveau même de la définition des rubriques, par scripts déclenchés à la saisie...

(En FileMaker, les deux possibilités existent : pour mettre la contrainte dans la définition même de la rubrique : voir l'onglet *Contrôle* des options de rubrique, dans la commande *Gérer > Base de données*).