

ECF : Bases de Données

1. Conception d'une base de données

1.1. Etude de cas BTS René Cassin

1.1.1. Dictionnaire des données

Donnée	Mnémotechnique	Domaine	Contraintes
Identifiant d'Enseignement	enseignement_id	INTEGER	O/U
Date de début d'Enseignement	enseignement_date_debut	DATE	O
Date de fin d'Enseignement	enseignement_date_fin	DATE	O
Identifiant de Matière	matiere_id	INTEGER	O/U
Libellé de Matière	matiere_libelle	VARCHAR(50)	O/U
Identifiant d'Option	option_id	INTEGER	O/U
Libellé d'Option	option_libelle	VARCHAR(0)	O/U
Coefficient de Matière	coefficient	INTEGER	O
Numéro du Professeur	professeur_no	INTEGER	O/U
Nom du Professeur	professeur_nom	VARCHAR(50)	O
Prénom du Professeur	professeur_prenom	VARCHAR(50)	O
Numéro de rue du Professeur	professeur_no_rue	INTEGER	
Rue du Professeur	professeur_rue	VARCHAR(50)	O
Code postal du Professeur	professeur_cp	VARCHAR(10)	O
Ville du Professeur	professeur_ville	VARCHAR(50)	O
Téléphone du Professeur	professeur_tel	VARCHAR(50)	O
Date de naissance du Professeur	professeur_date_naissance	DATE	O
Niveau de qualification du Professeur	professeur_niveau	VARCHAR(50)	
E-mail du Professeur	professeur_email	VARCHAR(50)	O LIKE '%@%'
Années d'ancienneté du Professeur	professeur_anciennete_annees	INTEGER	O
Numéro du Cours	cours_no	INTEGER	O/U
Heure prévue de début de Cours	cours_heure_debut_prevue	TIME	O
Heure prévue de fin de Cours	cours_heure_fin_prevue	TIME	O
Nombre d'élèves prévus du Cours	cours_nombre_eleves_prevus	INTEGER	O
Numéro de Salle de Cours	salle_cours_no	INTEGER	O/U
Libellé de Salle de Cours	salle_cours_libelle	VARCHAR(50)	O
Indication géographique de Salle de Cours	salle_cours_indication	VARCHAR(50)	
Capacité d'accueil de Salle de Cours	salle_cours_capacite	INTEGER	O
Numéro d'Elève	eleve_no	INTEGER	O/U
Nom d'Elève	eleve_nom	VARCHAR(50)	O
Prénom d'Elève	eleve_prenom	VARCHAR(50)	O
Numéro de rue d'Elève	eleve_no_rue	INTEGER	
Rue d'Elève	eleve_rue	VARCHAR(50)	O
Code postal d'Elève	eleve_cp	VARCHAR(10)	O
Ville d'Elève	eleve_ville	VARCHAR(50)	O
Téléphone d'Elève	eleve_tel	VARCHAR(50)	O
E-mail d'Elève	eleve_email	VARCHAR(50)	O

			LIKE '%@%'
Date de naissance d'Elève	eleve_date_naissance	DATE	O
Précédent diplôme d'Elève	eleve_precedent_diplome	VARCHAR(50)	
Année scolaire d'Elève	eleve_annee_scolaire	INTEGER	O
Numéro de Contrôle	contrôle_no	INTEGER	O/U
Note du Contrôle	contrôle_note	DECIMAL(4,2)	O
Date de passage du Contrôle	contrôle_date_passage	DATE	O

1.1.2. Règles de gestion

1 élève ne choisit qu'1 et 1 seule option.

1 option peut être choisie par 0 ou plusieurs élèves.

1 enseignement correspond à 1 et 1 seule matière.

1 matière est correspondante à 1 ou plusieurs enseignements.

1 professeur n'enseigne qu'1 et 1 seule matière.

1 matière peut être enseignée par 1 ou plusieurs professeurs.

1 enseignement n'est caractérisé que par 1 et 1 seule option.

1 option peut caractériser 1 ou plusieurs enseignements.

1 enseignement est composé d'1 ou plusieurs cours.

1 cours ne compose qu'1 et 1 seul enseignement.

1 cours est localisé dans 1 et 1 seule salle de cours.

1 salle de cours localise 0 ou plusieurs cours.

1 élève peut avoir 0 ou plusieurs notes.

1 note ne peut avoir été eue que par 1 et 1 seul élève.

1 note n'évalue qu'1 et 1 enseignement.

1 enseignement peut être évaluée par 0 ou plusieurs notes.

1.1.3. Dépendances fonctionnelles

1.1.3.1. *DF Simples*

enseignement_id → enseignement_date_debut, enseignement_date_fin

enseignement_id → matiere_id, option_id

matiere_id → matiere_libelle

option_id → option_libelle

professeur_no → professeur_nom, professeur_prenom, professeur_no_rue, professeur_rue, professeur_cp, professeur_ville, professeur_tel, professeur_date_naissance, professeur_niveau, professeur_email, professeur_anciennete_annees

professeur_no → matiere_id

cours_no → cours_heure_debut_prevue, cours_heure_fin_prevue, cours_nombre_eleves_prevus

cours_no → enseignement_id, salle_cours_no

salle_cours_no → salle_cours_libelle, salle_cours_indication, salle_cours_capacite

eleve_no → eleve_nom, eleve_prenom, eleve_no_rue, eleve_rue, eleve_cp, eleve_ville, eleve_tel, eleve_email, eleve_date_naissance, eleve_precedent_diplome, eleve_annee_scolaire

eleve_no → option_id

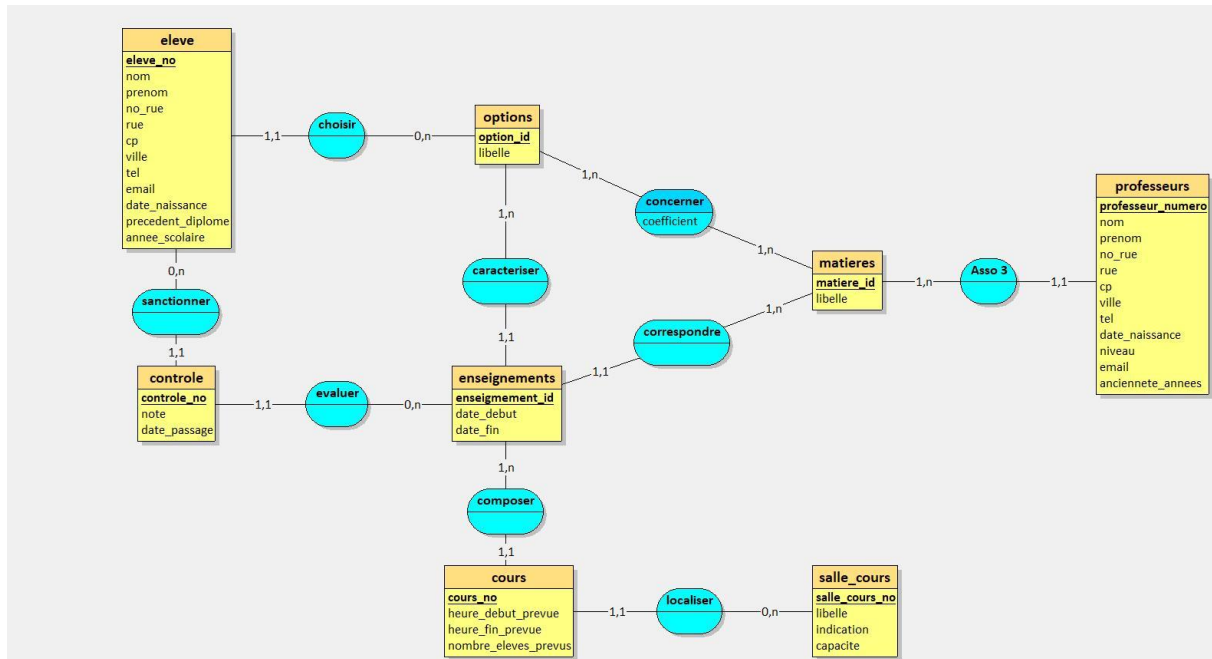
controle_no → controle_note, controle_date_passage

controle_no → eleve_no, enseignement_id

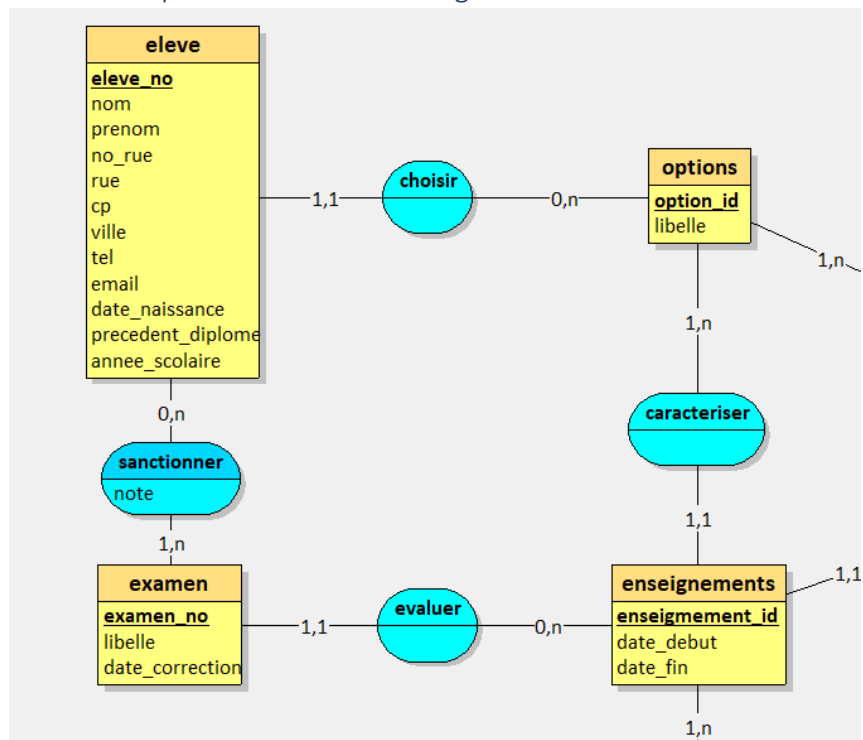
1.1.3.2. *DF Composées*

matiere_id, option_id → coefficient

1.1.4. Modèle Conceptuel des données



1.1.5. Modèle Conceptuel de Données corrigé



1.2. Règles de transformation

Le diagramme se transforme en modèle logique suivant :

ARTICLE (référence, libellé)

NOMENCLATURE (#référence, #composant)

L'importance des rôles intervient dans une association cyclique asymétrique. Qu'un article A compose un article B ne veut pas dire que A est composé de B. Il y a donc un sens d'interprétation et de relation entre les occurrences.

Dans le MLD ci-dessus, au sein de la table « NOMENCLATURE », « référence » désigne le composé, et « composant » le composant.

2. Modèle Physique et langage de définition

2.1. Exercice 1

```
ALTER TABLE realisations
  ADD PRIMARY KEY (id_rea)
;

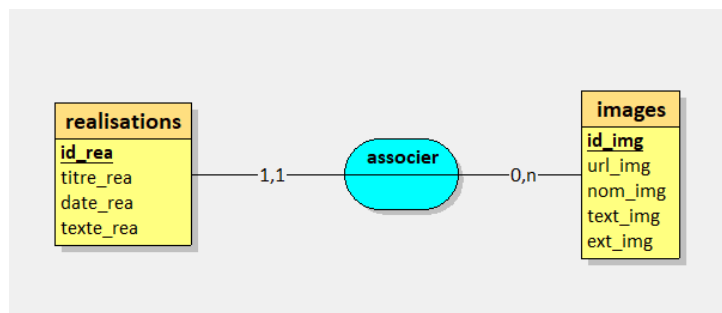
ALTER TABLE realisations
  AUTO_INCREMENT = 10
;
```

2.2. Exercice 2

```
ALTER TABLE images
  ADD PRIMARY KEY (id_img)
;

ALTER TABLE images
  AUTO_INCREMENT = 1
;
```

2.3. Exercice 3



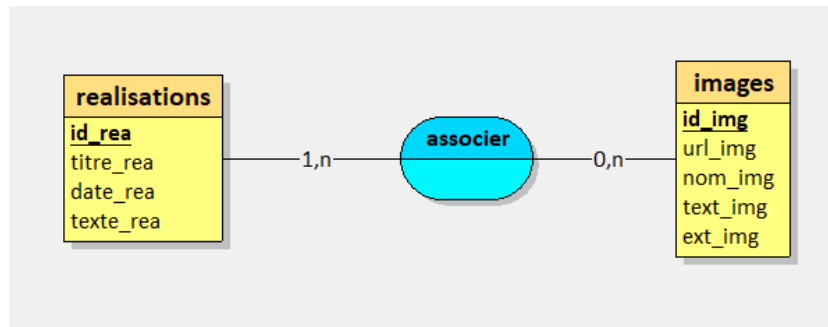
realisations (id_rea, titre_rea, date_rea, texte_rea, #id_img)

images (id_img, url_img, ,nom_img, text_img, ext_img)

Supposons que les commandes 2.1. et 2.2. ont été exécutées. Les suivantes seront :

```
ALTER TABLE realisations
  ADD COLUMN id_img INT(10) UNSIGNED NOT NULL,
  ADD CONSTRAINT fk_rea_img FOREIGN KEY (id_img)
  REFERENCES images(id_img)
;
```

2.4. Exercice 4



realisations (id_rea, titre_rea, date_rea, texte_rea)

images (id_img, url_img, nom_img, text_img, ext_img)

associations (#id_rea, #id_img)

2.5. Exercice 5

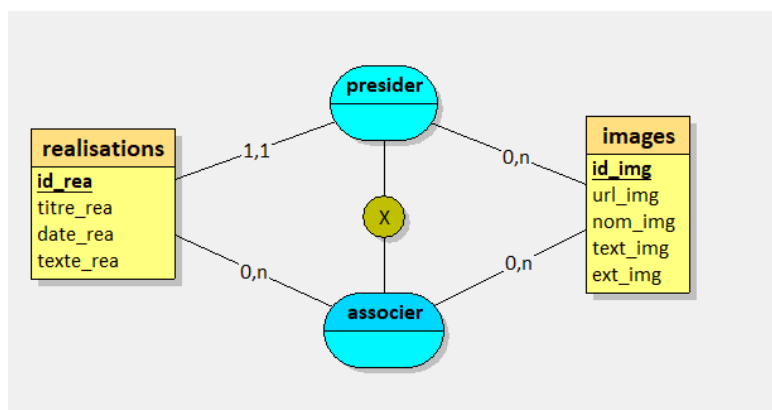
```
ALTER TABLE realisations
  DROP FOREIGN KEY fk_rea_img,
  DROP COLUMN id_img
;

CREATE TABLE associations (
  id_rea INT(10) UNSIGNED NOT NULL,
  id_img INT(10) UNSIGNED NOT NULL,

  PRIMARY KEY (id_rea, id_img)
);

ALTER TABLE associations
  ADD CONSTRAINT fk_asso_rea FOREIGN KEY (id_rea)
  REFERENCES realisations(id_rea),
  ADD CONSTRAINT fk_asso_img FOREIGN KEY (id_img)
  REFERENCES images(id_img)
;
```

2.6. Exercice 6



La contrainte inter-associations n'a de sens que s'il est primordial qu'une image ne puisse pas à la fois être principale et secondaire d'une même réalisation. Dans le cas contraire, on l'abat.

realisations (id_rea, titre_rea, date_rea, texte_rea, #id_img)

images (id_img, url_img, nom_img, text_img, ext_img)

associations(#id_rea, #id_img)