



École nationale supérieure d'informatique et de mathématiques appliquées

Passage au Relationnel

Notation UML

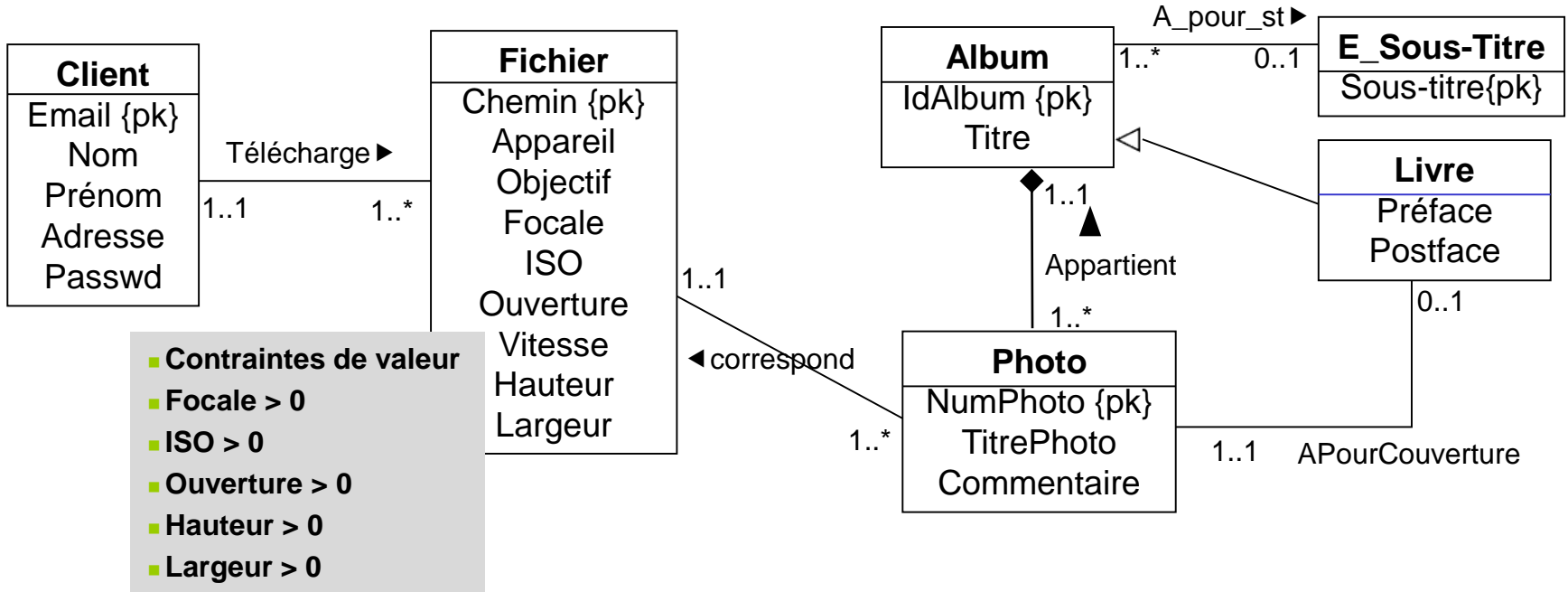
Equipe pédagogique BD



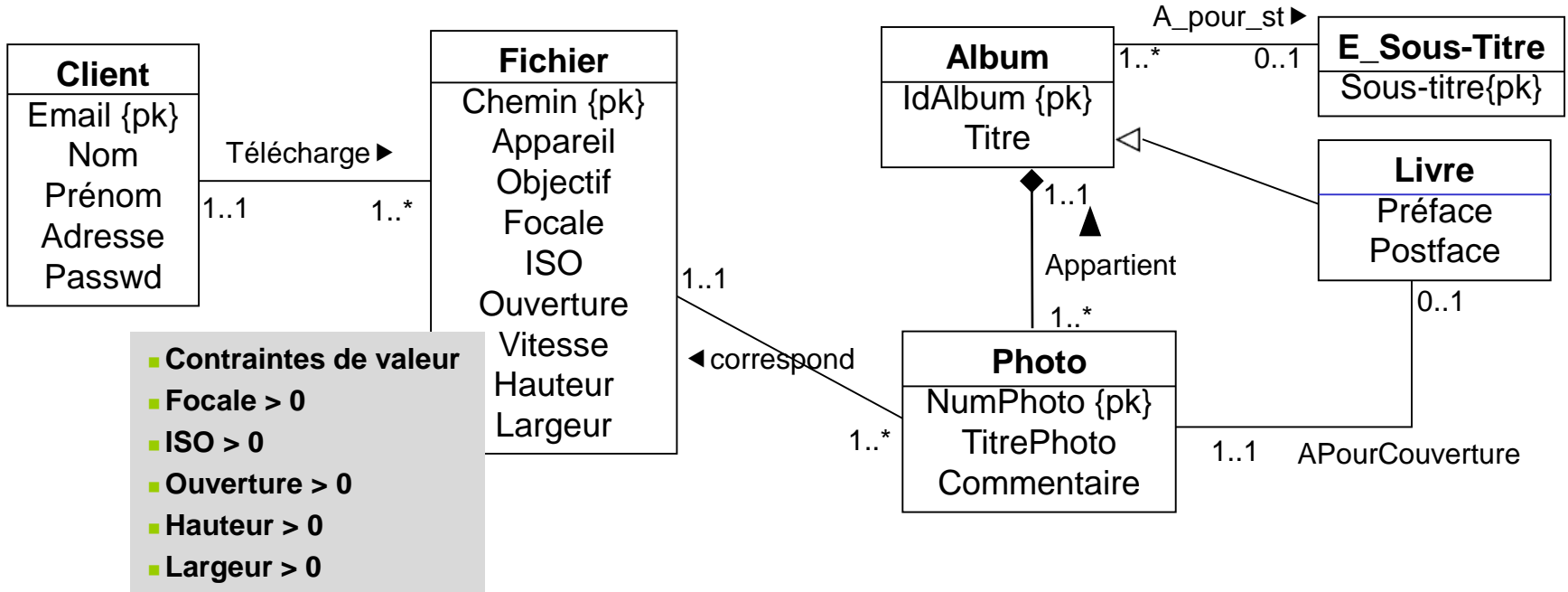
Principe

- Algorithme glouton
 - Tout concept n'est traduit qu'une et une seule fois
- Ordre de traduction
 1. Types d'Entités simples
 2. Sous-types d'entités
 3. Types d'Entités faibles
 4. Types d'Associations
 - a) Cardinalité 1..1
 - b) Cardinalité 0..1
 - c) Cardinalité ?..*
 5. Sous-types d'associations

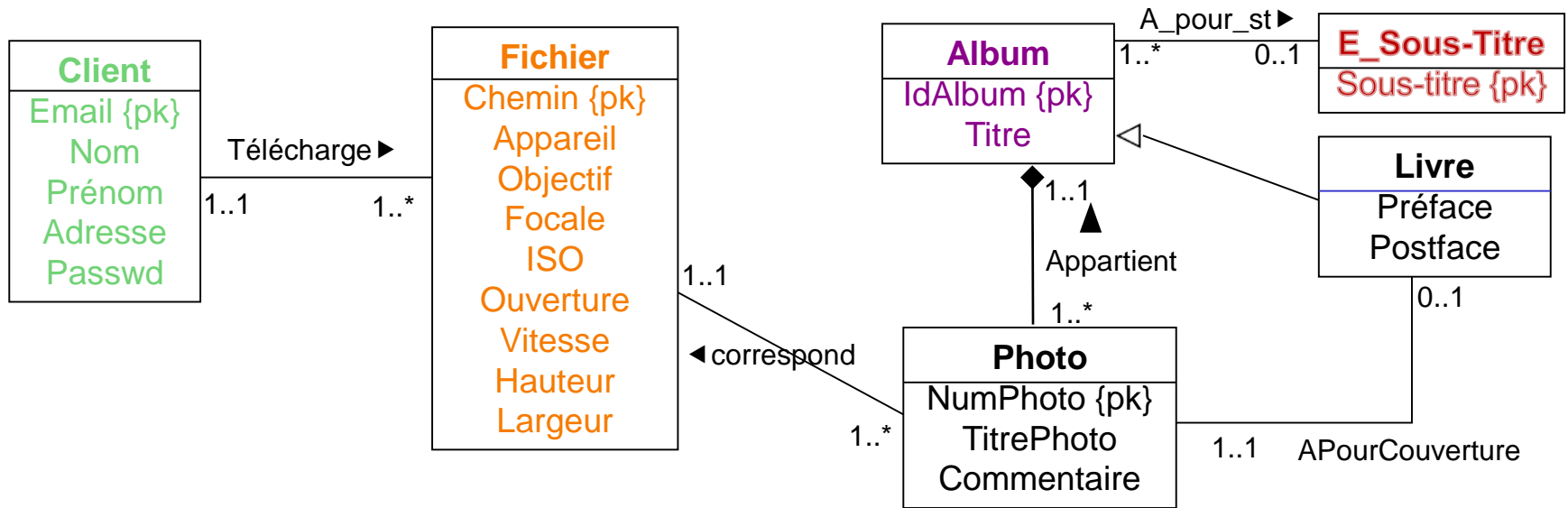
Fin conception Photonum



1. Types d'entités simples



1. Types d'Entités simples



Client(Email, Nom, Prénom, Adresse, Passwd)

Fichier(Chemin, Appareil, Objectif, Focale, ISO, Ouverture, Vitesse, Hauteur, Largeur)

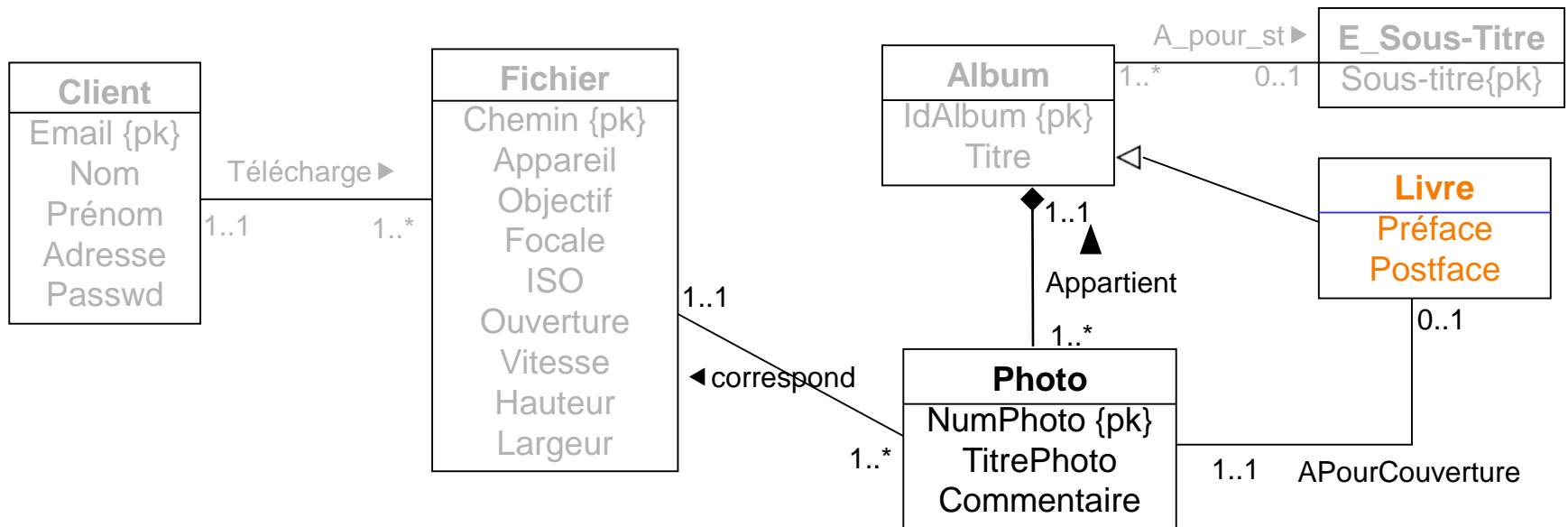
Album(IdAlbum, Titre)

E_Sous-Titre(Sous-Titre)

- Focale > 0
- ISO > 0
- Ouverture > 0
- Hauteur > 0
- Largeur > 0

2. Sous-types d'entité...

- **Semi-automatisable**
- **Plusieurs traductions possibles**
 - ◆ Contraintes à vérifier
 - ◆ Requêtes fréquentes



... 2. Sous-types d'entité...

- Cas 1 : Duplication

- **Album**(IdAlbum, Titre)

- **Livre**(IdAlbum, Titre, Préface, Postface)

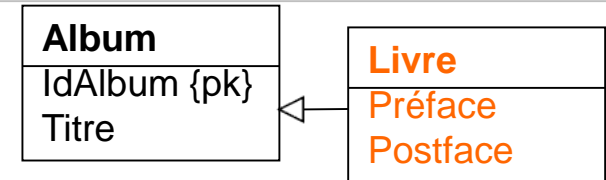
- IdAlbum de livre référence Album

- **Avantages**

- Accès aux n-uplets de Livre simple et immédiat
 - Facilité d'expression de contraintes (ex : associations, autres sous-types)

- **Inconvénients**

- Double insertion
 - Attention aux mises à jour
 - Plus d'espace occupé



... 2. Sous-types d'entité...

- Cas 2 : Référence

- **Album(IdAlbum, Titre)**

- **Livre(IdAlbum, Préface, Postface)**

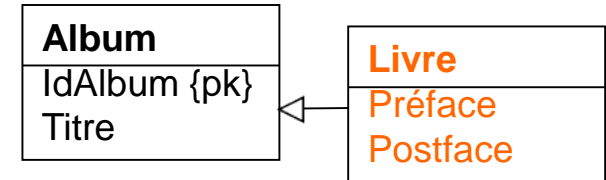
- IdAlbum de Livre référence Album

- **Avantages**

- Facilité d'expression des contraintes (ex : associations, sous-types)
 - Moins d'espace occupé
 - Pas de problème de mise à jour

- **Inconvénients**

- Double insertion
 - Accès aux n-uplets de Livre plus complexes (jointure)



... 2. Sous-types d'entité

- Cas 3 : Unification

- **Album**(IdAlbum, Titre, **Préface**, **Postface**, **Type**)

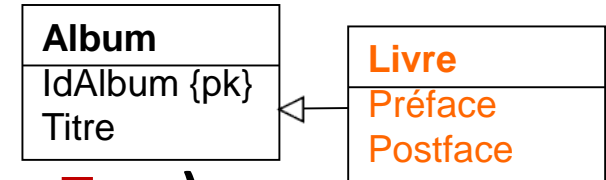
- Si **Type** = 'Album' alors Préface et Postface doivent être nuls

- **Avantages**

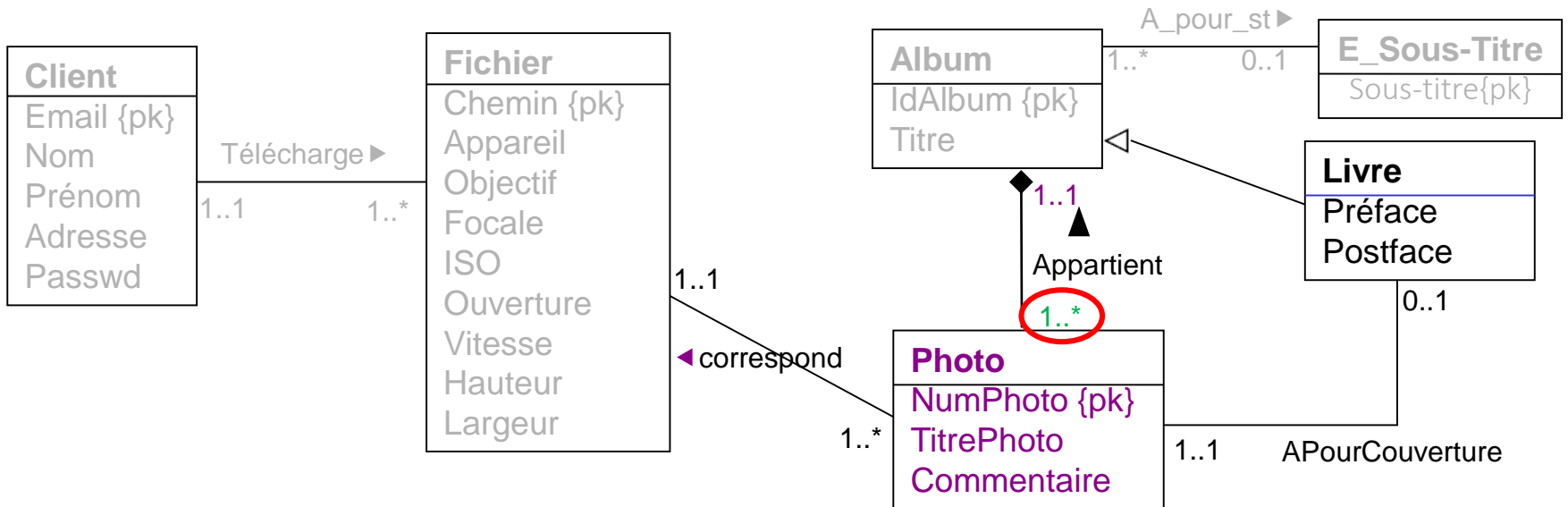
- Insertion unique
 - Pas de problème de mise à jour
 - Accès aux n-uplets de Livre immédiat

- **Inconvénients**

- Expression de contraintes de référence vers Livre gérée par l'application (e: associations, sous-types)



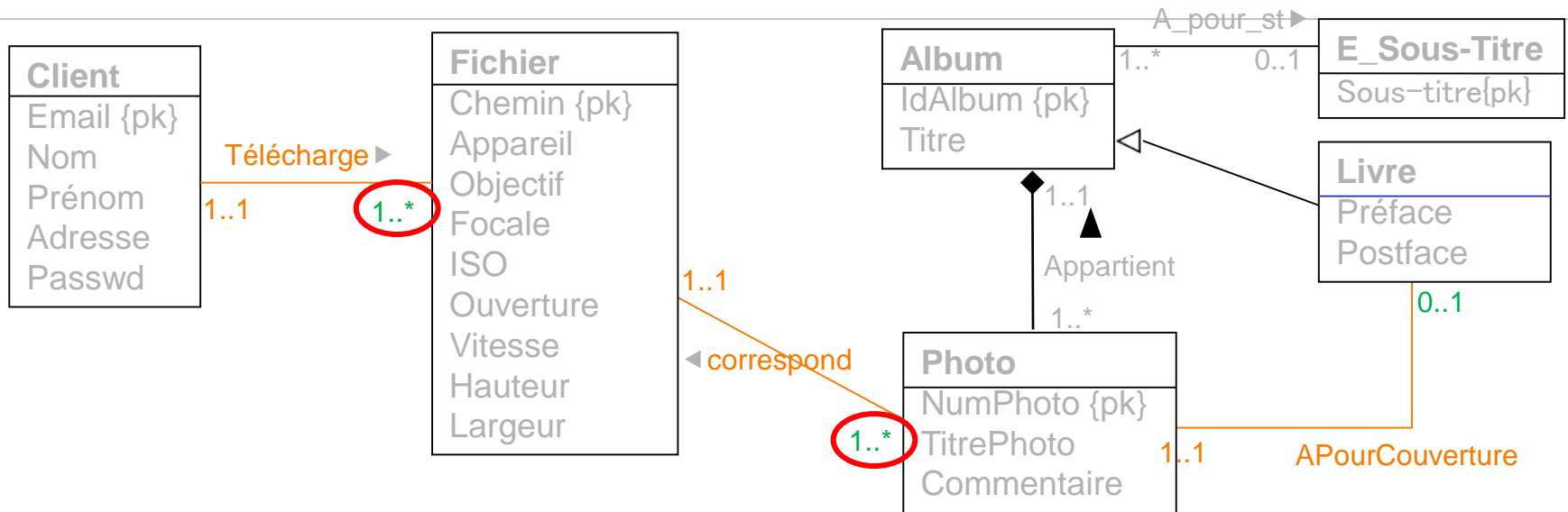
3. Types d'entité faible



Photo(IdAlbum, NumPhoto, TitrePhoto, Commentaire)

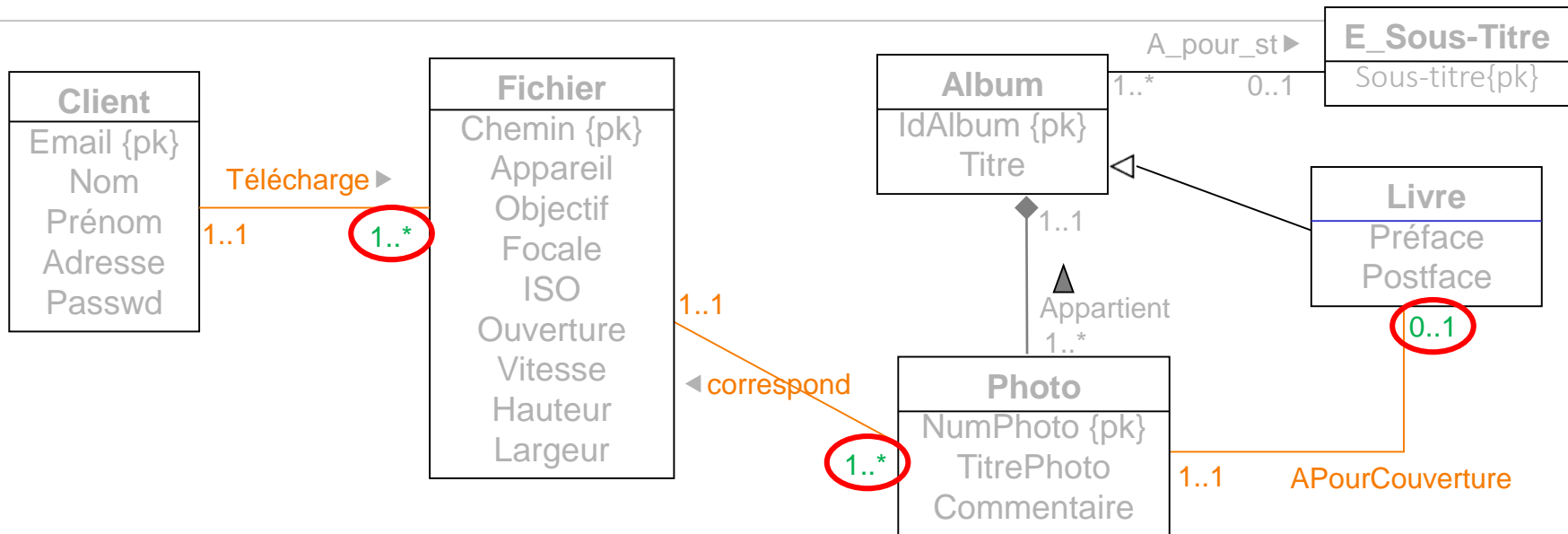
- ◆ IdAlbum référence Album(IdAlbum dans Album)
- ◆ Vérifier que tout album ait au moins une Photo

4.a) Associations avec cardinalité 1..1



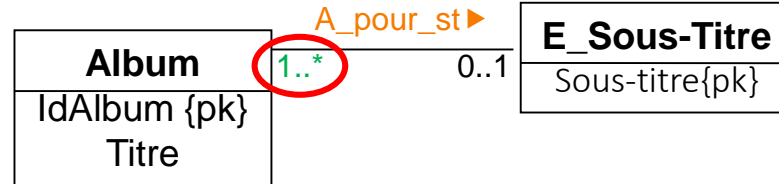
- Fichier(Chemin, Appareil, Objectif, Focale, ISO, Ouverture, Vitesse, Hauteur, Largeur, **Email**)
 - Email non nul et référence Client
 - Vérifier qu'un client ait téléchargé au moins un fichier
- Photo(IdAlbum, NumPhoto, TitrePhoto, Commentaire, **Chemin**)
 - Chemin non nul et référence Fichier
 - Vérifier qu'un fichier soit exploité au moins une fois par une photo
 - Corollaire : Tout fichier non exploité doit être supprimé

4.a) Associations avec cardinalité 1..1



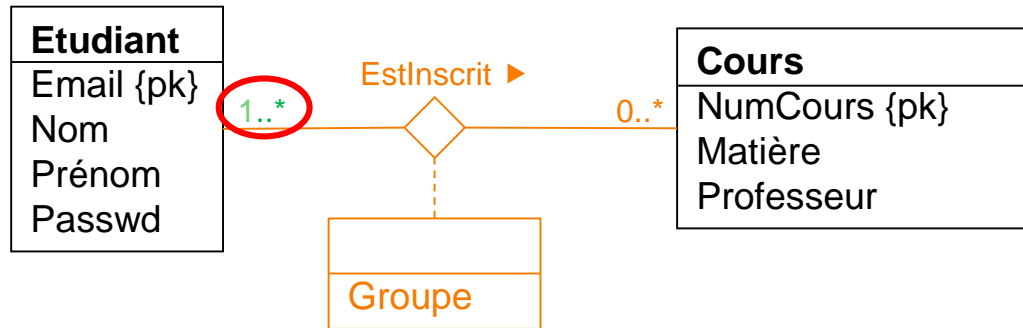
- Photo(IdAlbum, NumPhoto, TitrePhoto, Commentaire, **Chemin**)
 - Chemin non nul et référence Fichier
 - Vérifier qu'un fichier soit exploité au moins une fois par une photo
 - Corollaire : Tout fichier non exploité doit être supprimé
- Livre(IdAlbum, Préface, Postface, **NumPhoto**)
 - ❖ NumPhoto est non nul et référence une Photo du livre (IdAlbum, NumPhoto)
 - ❖ pas de vérification, les photos n'appartenant qu'à un livre

4.b) Association avec cardinalité 0..1



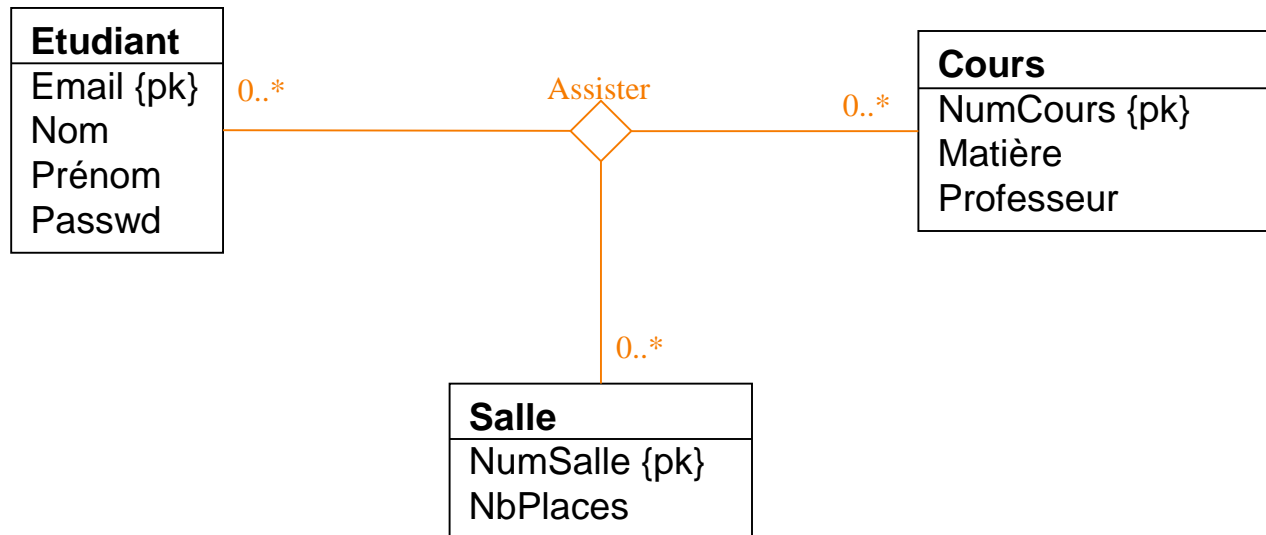
- **Album(IdAlbum, Titre)**
- **E_Sous_Titre(Sous-Titre)**
- **A_pour_st(IdAlbum, Sous-Titre)**
 - ◆ IdAlbum référence Album(IdAlbum)
 - ◆ Sous-Titre référence E_Sous-Titre(Sous-Titre)
 - ◆ Vérifier qu'un sous-titre soit toujours utilisé

4.c) Association avec cardinalité ?..*



- Etudiant(Email, Nom, Prénom, Passwd)
- Cours(NumCours, Matière, Professeur)
- EstInscrit(Email, NumCours, Groupe)
 - Email référence Etudiant
 - NumCours référence Cours
 - Vérifier qu'un cours a au moins un étudiant

Associations ternaires (ou plus)



- Etudiant(Email, Nom, Prénom, Passwd)
- Cours(NumCours, Matière, Professeur)
- Salle(NumSalle, NbPlaces)
- Assister(Email, NumCours, NumSalle)

5. Sous-types d'associations

- Pas de règle générale
- Traduction au cas par cas en fonction de
 - Cardinalités
 - Propriétés supplémentaires éventuelles
 - Entités associées
 - Contraintes
 - ...

Conclusion

- Le passage au relationnel
 - Automatisable
 - Types d'entité simples
 - Types d'entité faibles
 - Types d'association
 - Manuel
 - Sous-types d'entités ou d'associations
 - Contraintes
 - Induites par la traduction
 - Cardinalités non traduites
- Ne pas oublier les contraintes non représentées !***

Conception Photonum

DF	Cont. Valeur	Cont. Multiplicité
	<ul style="list-style-type: none"> ■ Focale > 0 ■ ISO > 0 ■ Ouverture > 0 ■ Hauteur > 0 ■ Largeur > 0 	

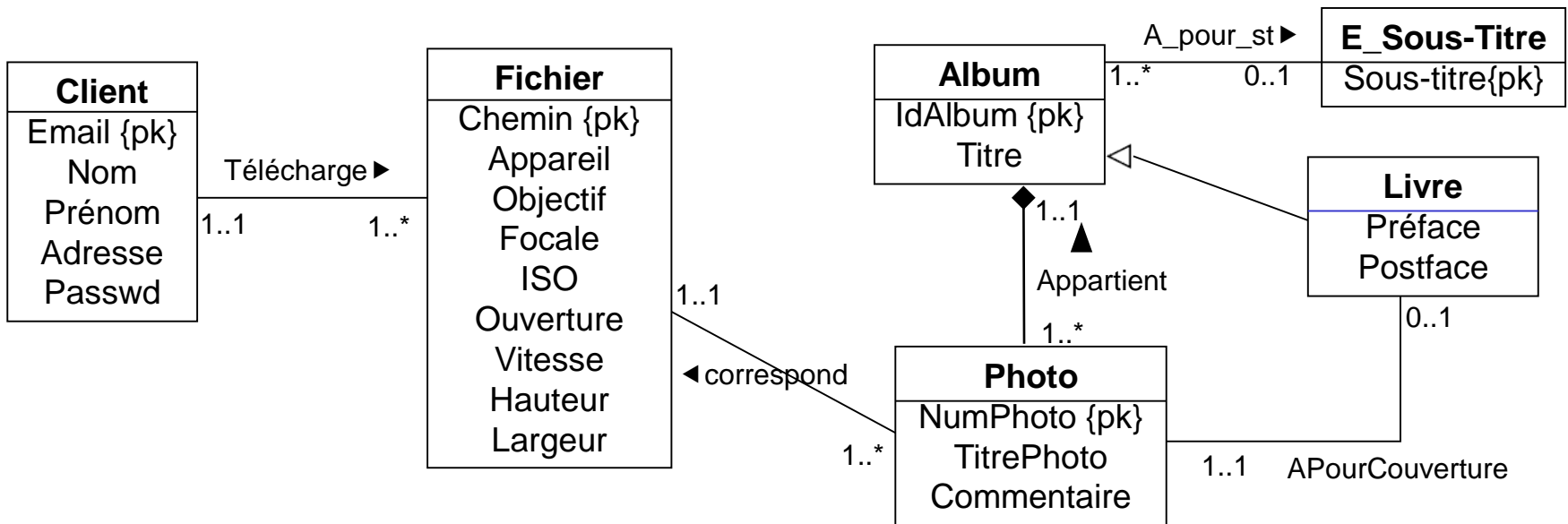


Schéma Relationnel - PhotoNum

- Client(Email, nom, prénom, adresse, passwd)

- Focale > 0
- ISO > 0
- Ouverture > 0
- Hauteur > 0
- Largeur > 0

Fichier(Chemin, appareil, objectif, focale, ISO, ouverture, vitesse, hauteur, largeur, email)

- ◆ email référence Client

- Album(idAlbum, titre)

- E_Sous_Titre(Sous-Titre)

- A_pour_st(IdAlbum, Sous-Titre)

- ◆ IdAlbum référence Album(IdAlbum)
- ◆ Sous-Titre référence E_Sous-Titre(Sous-Titre)

- Photo(idAlbum, NumPhoto, titrePhoto, commentaire, chemin)

- ◆ Chemin référence Fichier

- Livre(idAlbum, préface, postface, couverture) par exemple choix 2

- ◆ idAlbum référence Album
- ◆ (idAlbum, couverture) référence Photo(idAlbum, NumPhoto)

■ Contraintes non représentées

- ◆ Tout album doit avoir au moins une photo
- ◆ Tout client a téléchargé au moins un fichier
- ◆ Tout fichier doit être exploité au moins une fois par une photo
 - ★ Penser au corollaire : supprimer les fichiers sans photo, etc.

En résumé

- Schémas
 - Schéma Entités/Associations
 - Etat cohérent de la BD
 - Inclut toutes les contraintes
 - Schéma logique (relationnel)
 - Structures de données
 - Contraintes implantées
- Fonctionnalité
 - Transactions
 - Séquences de requêtes LMD
 - Assurent la cohérence conformément au schéma E/A (et contraintes associées)

Analyse de fct : Exemple PhotoNum

- Fonctionnalité

- Création d'un nouvel album ou livre par un client existant
- Vérification des contraintes (pas forcément à la fin)
 - Tout album doit avoir au moins une photo
 - ~~Tout client a téléchargé au moins un fichier~~
 - Tout fichier doit être exploité au moins une fois par une photo
 - Penser au corollaire : supprimer les fichiers sans photo, etc.

Est-ce cohérent ?

- Etapes

- Création de l'album (relation Album)
 - Titre, sous-titre
- Ajout de photos (relation Photo)
 - Titre, commentaire, chemin
 - Peut entrainer le téléchargement de fichiers (relation Fichier)
- Création du livre (relation livre)
 - Préface, postface, couverture

NON !

OUI
si contraintes
vérifiées

OUI
si contraintes
vérifiées

Exemple PhotoNum

- Création d' un nouvel album ou livre

1. **Begin;**

2. **Insert into** Album **values** (A, ?, ?);

3. **Insert into** Photo **values** (A, x, ?, ?, C);

- Peut entrainer : **Insert into** Fichier **values** (C, ...);
- 3 peut être répété plusieurs fois

// il peut y avoir des fichiers non utilisés

1. **Delete from** Fichier **where** Chemin **not in**
(**select** Chemin **from** Photo);

// il peut y avoir des Albums sans photos

1. **Delete from** Album **where** idAlbum **not in**
(**select** idAlbum **from** Photo);

2. **[Commit;]** -- Pour un album

3. **Insert into** Livre **values** (A, ?, ?, x);

4. **Commit;** -- Pour un livre