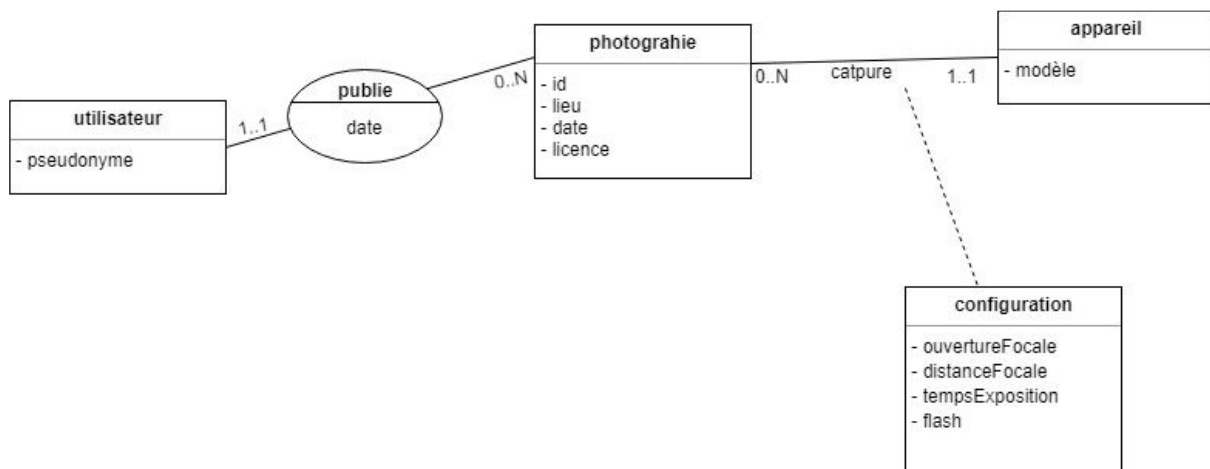


HMIN122M Entrpôts de données et Big data

TP 1

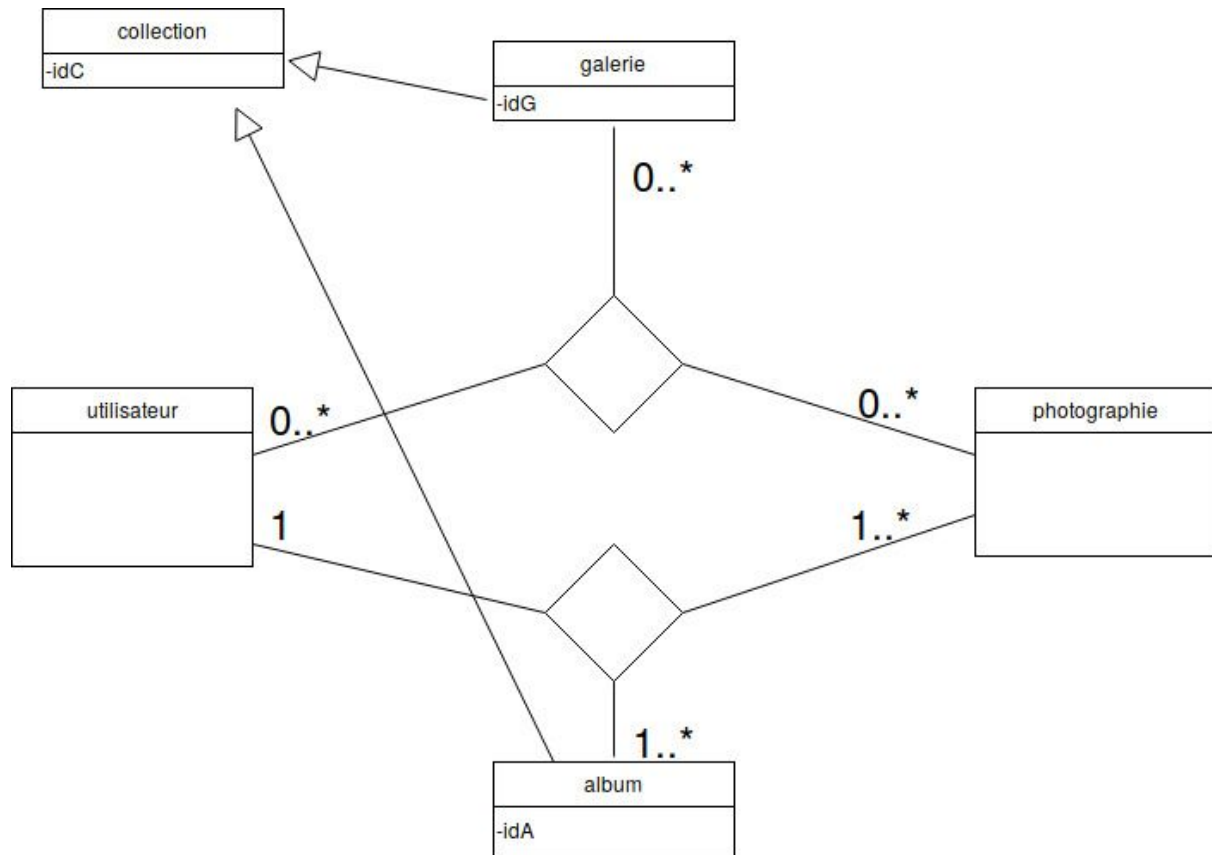
1) Modélisation conceptuelle : la gestion des photos dans la plateforme flickr :

Cahier des charges #1 : les photos



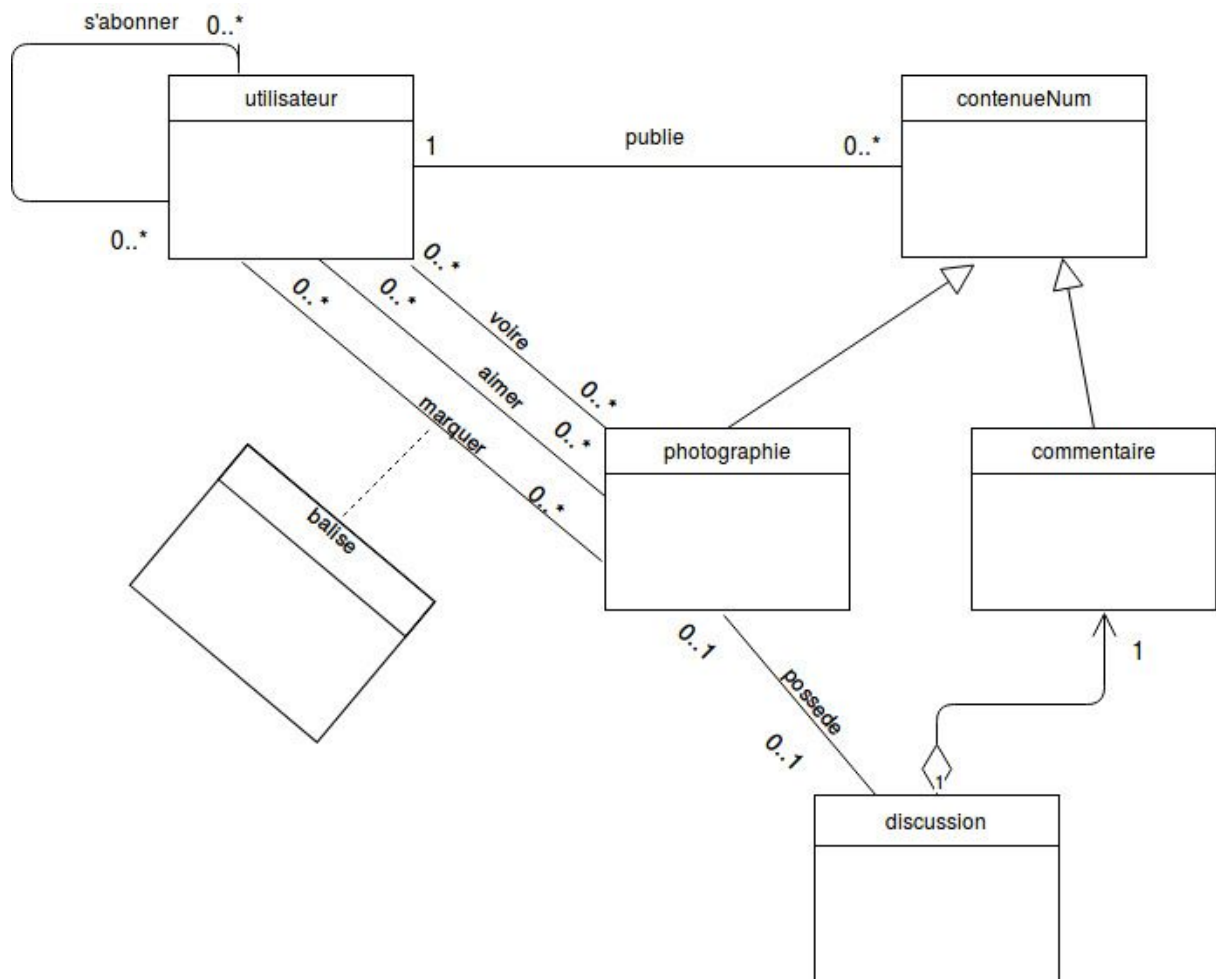
1. Création d'une relation de publication entre 2 classes : l'utilisateur et la photographie avec une date de publication
2. Ajout d'une relation entre une photographie et l'appareil qui l'a capturé avec une classe d'association pour stocker la configuration de l'appareil
3. Le lieu et la date de réalisation de l'image sont enregistrés dans la classe photographie qui lui sera beaucoup demandé lors de recherche sur twitter
4. L'id de la classe peut être un code pour repérer une photographie dans le système de stockage distribuée des photos de l'application
5. Propriété de la licence de la photo ajouté aux attributs de la classe photographie

Cahier des charges #2 : publications, albums, et galeries



1. On a créé 2 associations ternaires afin que les photos se rangent dans les galeries et les albums de façons indépendantes.
2. La cardinalité 1 du côté de l'entité **utilisateur** dans l'association ternaire entre les entités **utilisateur**, **album** et **photographie** permet l'appartenance de toutes les photos d'un album au même propriétaire.
3. Grâce aux cardinalités dans l'association ternaire entre entités **utilisateur**, **galerie** et **photographie**, on octroie la possibilité d'avoir dans une même galerie des photos d'utilisateurs différents.
4. En décidant d'utiliser l'héritage avec l'entité **collection**, nous avons opté pour les classes filles (**album**, **galerie**) de rajouter des clés étrangères référençant la classe mère (**collection**).

Cahier des charges #3: les interactions entre utilisateurs



1. On a créé les relations 's'abonner', 'aimer' et 'publie' afin que les utilisateurs de la plateforme Flickr puisse interagir de différentes façons
2. La relation 'marquer' permet à un utilisateur de marquer ses photos et leur associer des balises qui seront des attributs de la classe d'association 'balise'
3. La relation 'voir' permettra de stocker le nombre de vu par photographie
4. On a créé une agrégation entre la classe commentaire et discussion
5. La cardinalité sur l'agrégation de 1:1 nous permet d'avoir qu'un commentaire par discussion
6. L'effacement d'une photo supprimera la classe 'contenuNum' et ainsi les commentaires qui y seront liés et donc la discussion
7. Les classes 'photographie' et 'commentaire' sont des sous-classes de 'contenuNum'

Schéma relationnel :

photographie	(<u>idP</u> , lieu, licence, #idCNum)
publie	(<u>idP</u> , <u>idU</u> , date)
utilisateur	(<u>idU</u> , pseudonyme, #idCNum ,#idA ,#idP)
configuration	(<u>idApp</u> , <u>idP</u> , ouvretureF, tempsE, flash, distanceF)
appareilPhoto	(<u>idApp</u> , marque, #idPh)
appartenirGalerie	(# <u>idU</u> , # <u>idG</u> , # <u>idP</u>)
appartenirAlbum	(<u>idU</u> , # <u>idP</u> , # <u>idA</u>)
collection	(<u>idC</u>)
galerie	(<u>idG</u> , #idCollec)
album	(<u>idA</u> , #idCollec)
contenuNum	(<u>idCNum</u>)
commentaire	(<u>idCom</u> , #idDisc, #idCNum)
discussion	(<u>idDisc</u> , #idP)
balisePhoto	(# <u>idU</u> , # <u>idP</u>)
aimerPhoto	(# <u>idU</u> , # <u>idP</u>)
marquerPhoto	(# <u>idU</u> , # <u>idP</u>)
voirePhoto	(# <u>idU</u> , # <u>idP</u>)
possèdeDiscussion	(# <u>idP</u> , # <u>idDisc</u>)
abonnement	(# <u>idU</u> _1, # <u>idU</u> _2)

2) Modèle physique des données : implémentation et requêtage sur Oracle

3.a)

- “les photos de Montpellier”

```
SELECT idP FROM photographie  
WHERE lieu = “Montpellier” ;
```

- “les photos sans licence de Paris”

```
SELECT idP FROM photographie  
WHERE licence != NULL  
AND lieu = “Paris”;
```

- “les photos publiées le 08/08/2018”

```
SELECT idP FROM publie  
WHERE date='08/08/2018';
```

3.b)

```
SELECT idP FROM photographie ph, aimerPhoto ap  
WHERE licence = “tous droits réservés”  
AND ap.idP = ph.idP  
GROUP BY idP  
HAVING Max(count(*));
```

3.c)

```
SELECT idP, count(idG) FROM photographie,appartenirGalerie  
WHERE photographie.idP=appartenirGalerie.idP  
GROUP BY idP, count(idG)  
ORDER BY DESC;
```

```

1 CREATE TABLE utilisateur(
2     idU      NUMERIC(5),
3     pseudonyme VARCHAR(50),
4     idA      NUMERIC(5),
5     idCNum   NUMERIC(5),
6     idP      NUMERIC(5),
7     CONSTRAINT PK_utilisateur PRIMARY KEY (idU),
8     CONSTRAINT FK_utilisateur_contenuNum FOREIGN KEY (idCNum) REFERENCES contenuNum(idCNum),
9     CONSTRAINT FK_utilisateur_album FOREIGN KEY (idA) REFERENCES album(idA),
10    CONSTRAINT FK_utilisateur_photographie FOREIGN KEY (idP) REFERENCES photographie(idP)
11 );
12
13
14 CREATE TABLE photographie(
15     idP      NUMERIC(5),
16     lieu     VARCHAR(20),
17     licence  VARCHAR(50),
18     idDisc   NUMERIC(5),
19     idCNum   NUMERIC(5),
20     CONSTRAINT PK_photographie PRIMARY KEY (idP),
21     CONSTRAINT FK_photographie_discussion FOREIGN KEY (idDisc) REFERENCES discussion(idDisc),
22     CONSTRAINT FK_photographie_contenuNum FOREIGN KEY (idCNum) REFERENCES contenuNum(idCNum)
23 );
24
25 CREATE TABLE appareilPhoto(
26     idApp    NUMERIC(5),
27     marque   VARCHAR(10),
28     idP      NUMERIC(5),
29     CONSTRAINT PK_appareilPhoto PRIMARY KEY (idApp),
30     CONSTRAINT FK_appareilPhoto_photographie FOREIGN KEY (idP) REFERENCES photographie(idP)
31 );
32
33 CREATE TABLE publie(
34     idU      NUMERIC(5),
35     idP      NUMERIC(5),
36     DATE T   DATE,
37     CONSTRAINT PK_publie PRIMARY KEY (idU, idP)
38 );
39
40 CREATE TABLE configuration(
41     idApp    NUMERIC(5),
42     idP      NUMERIC(5),
43     ouvertureF VARCHAR(10),
44     distanceF  VARCHAR(10),
45     tempsE     VARCHAR(10),
46     flash      VARCHAR(10),
47     CONSTRAINT PK_configuration PRIMARY KEY (idApp, idP)
48 );
49
50 CREATE TABLE appartenirGalerie(
51     idU      NUMERIC(5),
52     idP      NUMERIC(5),
53     idG      NUMERIC(5),
54     CONSTRAINT PK_appartenirGalerie PRIMARY KEY (idU, idG, idP),
55     CONSTRAINT FK_appartGalerie_gal FOREIGN KEY (idG) REFERENCES galerie(idG),
56     CONSTRAINT FK_appartGalerie_util FOREIGN KEY (idU) REFERENCES utilisateur(idU),
57     CONSTRAINT FK_appartGalerie_photo FOREIGN KEY (idP) REFERENCES photographie(idP)
58 );
59
60 CREATE TABLE apparteniralbum(
61     idU      NUMERIC(5),
62     idP      NUMERIC(5),
63     IDA      NUMERIC(5),
64     CONSTRAINT PK_apparteniralbum PRIMARY KEY (idU, IDA, idP),
65     CONSTRAINT FK_apparteniralbum_photo FOREIGN KEY (idP) REFERENCES photographie(idP),
66     CONSTRAINT FK_apparteniralbum_album FOREIGN KEY (idA) REFERENCES album(idA)
67 );
68
69 CREATE TABLE collection(
70     idC      NUMERIC(5),
71     CONSTRAINT PK_collection PRIMARY KEY (idC)
72 );
73
74 CREATE TABLE galerie(
75     idG      NUMERIC(5),
76     idC      NUMERIC(5),
77     CONSTRAINT PK_galerie PRIMARY KEY (idG),
78     CONSTRAINT FK_galerie_collection FOREIGN KEY (idC) REFERENCES collection(idC)
79 );
80
81 CREATE TABLE album(
82     idA      NUMERIC(5),
83     idC      NUMERIC(5),
84     CONSTRAINT PK_album PRIMARY KEY (idA),
85     CONSTRAINT FK_album_collection FOREIGN KEY (idC) REFERENCES collection(idC)
86 );
87
88 CREATE TABLE contenuNum(
89     idCNum   NUMERIC(5),
90     CONSTRAINT PK_contenuNum PRIMARY KEY (idCNum)
91 );
92
93 CREATE TABLE commentaire(
94     idCom    NUMERIC(5),
95     idDisc   NUMERIC(5),
96     idCNum   NUMERIC(5),
97     CONSTRAINT PK_commentaire PRIMARY KEY (idCom),
98     CONSTRAINT FK_commentaire_discussion FOREIGN KEY (idDisc) REFERENCES discussion (idDisc),
99     CONSTRAINT FK_commentaire_contenuNum FOREIGN KEY (idCNum) REFERENCES contenuNum (idCNum)
100 );

```

```

102 CREATE TABLE discussion(
103     idDisc NUMERIC(5),
104     idP NUMERIC(5),
105     CONSTRAINT PK_discussion PRIMARY KEY (idDisc)
106 );
107
108 CREATE TABLE balisePhoto(
109     idU NUMERIC(5),
110     idP NUMERIC(5),
111     contenu VARCHAR(10),
112     CONSTRAINT PK_balise PRIMARY KEY (idU, idP),
113     CONSTRAINT FK_balise_utilisateur FOREIGN KEY (idU) REFERENCES utilisateur (idU),
114     CONSTRAINT FK_balise_photo FOREIGN KEY (idP) REFERENCES photographie (idP)
115 );
116
117 CREATE TABLE aimerPhoto(
118     idU NUMERIC(5),
119     idP NUMERIC(5),
120     CONSTRAINT PK_aimer PRIMARY KEY (idU, idP),
121     CONSTRAINT FK_aimer_utilisateur FOREIGN KEY (idU) REFERENCES utilisateur (idU),
122     CONSTRAINT FK_aimer_photo FOREIGN KEY (idP) REFERENCES photographie (idP)
123 );
124
125 CREATE TABLE marquerPhoto(
126     idU NUMERIC(5),
127     idP NUMERIC(5),
128     CONSTRAINT PK_marquer PRIMARY KEY (idU, idP),
129     CONSTRAINT FK_marquer_utilisateur FOREIGN KEY (idU) REFERENCES utilisateur (idU),
130     CONSTRAINT FK_marquer_photo FOREIGN KEY (idP) REFERENCES photographie (idP)
131 );
132
133 CREATE TABLE possedeDiscussion(
134     idP NUMERIC(5),
135     idDisc NUMERIC(5),
136     CONSTRAINT PK_possedeDiscussion PRIMARY KEY (idDisc, idP)
137 );
138
139 CREATE TABLE abonnement(
140     idU_1 NUMERIC(5),
141     idU_2 NUMERIC(5),
142     CONSTRAINT PK_abonnement PRIMARY KEY (idU_1, idU_2)
143     CONSTRAINT FK_abonnement_utilisateur1 FOREIGN KEY (idU_1) REFERENCES utilisateur (idU);
144     CONSTRAINT FK_abonnement_utilisateur2 FOREIGN KEY (idU_2) REFERENCES utilisateur (idU);
145 );

```