

Ἀκαδήμεια Μῆτις

Département d'informatique
Faculté des sciences
Université de Sherbrooke, Québec

BD021x Traduction-EA-UML_cas-Sondage

Émetteur : Luc Lavoie

Date de prestation initiale : 2020-11-26

Date de diffusion de la version 0.1.0 : 2021-11-14

Objectif : Illustrer la technique de traduction systématique d'un schéma conceptuel de type EA en un schéma logique de type relationnel en 5FN

Plan

Sommaire	2
Avertissement	2
Historique des révisions	2
1 Introduction	3
1.1 Objet et portée du document	3
1.2 Évolution du document	3
1.3 Travail en cours ou projeté	3
1.4 Contenu des sections	3
1.5 Notation	3
2 Le cas d'étude Sondage	3
3 Démarche	4
3.1 Entités fortes (attributs stockés et uniques ; clés)	4
3.2 Associations simples	4
3.3 Associations déterminantes et entités faibles	4
3.4 Dérivations et unions	4
3.5 Attributs multiples	5
3.6 Attributs calculés	5
3.7 DF et FNBC et DJ et 5FN	5
3.8 Invariants	6
4 Conclusion	6

Sommaire

Le présent document s'inscrit dans le cadre du projet Bases de données. Il a pour principaux objectifs de :

- ◇ familiariser le lecteur à la traduction de schéma conceptuel EA en un schéma logique relationnel;
- ◇ permettre au lecteur de constater que le schéma ainsi produit est en FNBC ;
- ◇ permettre au lecteur de constater qu'il est rarement nécessaire de faire une étude poussée pour conclure que ce même schéma est également en 5FN.

Groupe Ακαδημία Μητίς
Faculté des sciences
Université de Sherbrooke
Sherbrooke, QUÉBEC J1K 2R1

© 2018-2024, Akademia Metis (<http://info.usherbrooke.ca/llavoie>)
CC BY-4.0 (<https://creativecommons.org/licenses/by/4.0/>)

Avertissement

Le présent module est en cours d'élaboration. Merci de signaler les erreurs et améliorations par courriel à luc.lavoie@usherbrooke.ca.

Historique des révisions

version	date	auteur	description
0.1.0a	2020-11-28	LL	Mise en forme
0.0.1a	2020-11-26	LL	Récupération de notes de prestations antérieures.

3 Démarche

La démarche comprend les étapes suivantes :

- (1) Entités fortes (attributs stockés et uniques ; clés)
- (2) Associations simples
- (3) Entités faibles et associations déterminantes (attributs stockés et uniques ; clés)
- (4) Dérivations et unions
- (5) Attributs multiples
- (6) Attributs calculés
- (7) DF et FNBC ; DJ et 5FN
- (8) Invariants

3.1 Entités fortes (attributs stockés et uniques ; clés)

Questionnaire {idQ, titre, debutSaisie, finSaisie} clé {idQ}

*** attribut multiple : auteur

Étudiant {matricule, nom, prénom, ddn, programme, courriel} clé {matricule}

*** attribut calculé : âge

Question {noQ, mode, libellé} clé {noQ}

3.2 Associations simples

Contenu {idQ, noQ} clé {idQ, noQ}

3.3 Associations déterminantes et entités faibles

Sondage

Formulaire {idQ, matricule, dateR} clé {idQ, matricule}

Résultat

Réponse {idQ, matricule, noQ, momentSaisie} clé {idQ, matricule, noQ}

Choix

ChoixQCM {libellé, noC, noQ} clé {noQ, noC} ref {noQ} -> Question

3.4 Dérivations et unions

3.4.1 Question – (o) – QCM, QO

Les entités QCM et QO ne comportent pas d'attributs, elles ne seront donc pas intégrées au schéma relationnel. Deux contraintes sont à considérer :

Participation (2..*)

Pas de contrainte à cause de l'insertion initiale, mais une fonction de validation devra être fournie dans l'IMM.

Intégrité du type de question

Un tuple ChoixQCM ne peut être associé à une QO. Afin de faciliter la vérification du type de la question (QCM, QCMO, QO), l'attribut de contrôle **type** ajouté aux attributs de Question. L'invariant est donc le suivant (SQL) :

```
create assertion ChoixQCM_non_QO
check (
  not exists (
    select * from Question natural join ChoixQCM where type = 'QO'
  )
)
```

3.4.2 Question – (d) – RCM, RO

Les entités RCM et RO comportent des attributs, elles doivent donc être intégrées au schéma relationnel.

RCM {idQ, matricule, noQ, noC} clé {idQ, matricule, noQ}

RO {idQ, matricule, noQ, texte} clé {idQ, matricule, noQ}

La validation de leur exclusion mutuelle ne nécessite pas l'ajout d'un attribut de contrôle à la relation Réponse. L'invariant est donc le suivant (SQL) :

```
create assertion RCM_RP_disjoint
check (NOT EXISTS (select * from RCM natural join RO))
```

3.5 Attributs multiples

On fera ici l'hypothèse que tous les auteurs porteront un nom (ou seront désignés par un CIP) distincts.

Questionnaire_Auteurs {idQ, auteur} clé {idQ, auteur}

3.6 Attributs calculés

```
create view Etudiant_V as
select
  matricule, nom, prénom, ddn, programme, courriel,
  (now()-ddn) as age
from Etudiant
```

3.7 DF et FNBC et DJ et 5FN

Questionnaire {idQ, titre, debutSaisie, finSaisie} clé {idQ}

DFNC : aucune => FNBC ; au moins 1 attribut non clé => 5FN.

Questionnaire_Auteurs {idQ, auteur} clé {idQ, auteur}

DFNC : aucune => FNBC ; aucune DJ non triviale => 5FN.

Question {noQ, mode, libellé, type} clé {noQ}

DFNC : aucune => FNBC ; au moins 1 attribut non clé => 5FN.

ChoixQCM {noQ, noC, libellé} clé {noQ, noC} ref {noQ} -> Question

DFNC : aucune, FNBC ; au moins 1 attribut non clé => 5FN.

Contenu {idQ, noQ} clé {idQ, noQ}

DFNC : aucune, FNBC ; au moins 1 attribut non clé => 5FN.

Étudiant {matricule, nom, prénom, ddn, programme, courriel} clé {matricule}

DFNC : aucune => FNBC ; au moins 1 attribut non clé => 5FN.

Formulaire {idQ, matricule, dateR} clé {idQ, matricule}

DFNC : aucune, FNBC ; au moins 1 attribut non clé => 5FN.

Réponse {idQ, noQ, matricule, momentSaisie} clé {idQ, noQ, matricule}

DFNC : aucune, FNBC ; au moins 1 attribut non clé => 5FN.

RCM {idQ, noQ, matricule, noC} clé {idQ, noQ, matricule}

DFNC : aucune => FNBC ; au moins 1 attribut non clé => 5FN.

RO {idQ, noQ, matricule, texte} clé {idQ, noQ, matricule}

DFNC : aucune => FNBC ; au moins 1 attribut non clé => 5FN.

3.8 Invariants

Pour les invariants, en l'absence de CREATE ASSERTION, il faudra définir des TRIGGER et des TRIGGER FUNCTION.

4 Conclusion

*** À développer suite à la mise en oeuvre de la conception.

