



Bases de données

Conception relationnelle

Introduction

MLR_02
v101b
2025-04-07

Christina.Khnaisser@USherbrooke.ca
Luc.Lavoie@USherbrooke.ca

© 2018-2021, **Matus** (<http://info.usherbrooke.ca/llavoie>)
CC BY-NC-SA 4.0 (<https://creativecommons.org/licenses/by-nc-sa/4.0/>)

1

Plan

- *Introduction*
- *Exercice*

2025-04-07
MLR_02 : Conception relationnelle - introduction (v101b) © 2018-2025, Matus - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

2

2

Introduction

- Démarche
- Base d'analyse (modèle conceptuel)
- Vocabulaire
- Correspondance

2025-04-07

MIR_02 : Conception relationnelle - introduction (v101b) © 2018-2025, Mifs - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

3

3

Démarche 3 phases

- Phase 1 – Modélisation essentielle
- Phase 2 – Validité, cohérence, efficacité
- Phase 3 – Complétude, évolutivité et efficience

2025-04-07

MIR_02 : Conception relationnelle - introduction (v101b) © 2018-2025, Mifs - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

4

4

Démarche
Phase 1 – Modélisation essentielle

1. Analyse
2. Synthèse de la liste des classes, des attributs, des types et des traitements
3. Détermination des dépendances fonctionnelles
4. Synthèse de relations

2025-04-07

MIR_02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. P. S. - CC BY-NC-SA 4.0
 Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

5

5

Démarche
Phase 2 – Validité, cohérence, efficacité

1. Normalisation
 - a. 1FN : typage et clé référentielle
 - b. FNBC : intégration des dépendances fonctionnelles
 - c. FNPI : intégration des dépendances de projection-jointure
2. Vérification des clés (candidates et référentielles)
3. Traitement des données potentiellement absentes

2025-04-07

MIR_02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. P. S. - CC BY-NC-SA 4.0
 Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

6

6

Démarche**Phase 3 – Complétude, évolutivité et efficience**

1. Inclusion des règles
 - a. de domaine (toujours)
 - b. de métier (sauf exception)
 - c. d'affaires (discutable)
 - d. ...
2. Prise en compte de l'évolutivité
3. Optimisation ?!?!?

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MfP5 - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

7

7

Démarche

1. Analyse
2. Synthèse de la liste des classes, des attributs, des types et des traitements
3. Détermination des dépendances fonctionnelles
4. Synthèse de relations
5. Normalisation
6. Vérification des clés (candidates et référentielles)
7. Traitement des données potentiellement absentes
8. Inclusion des règles
9. Prise en compte de l'évolutivité
10. Optimisation

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MfP5 - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

8

8

Base d'analyse *Classes et entités*

- Une classe représente un ensemble d'entités (physiques ou abstraites) partageant des caractéristiques communes.
- Relativement à un problème donné, une classe répond généralement aux critères de pertinence suivants :
 - Elle est digne d'intérêt pour les parties prenantes au problème.
 - Elle participe à la définition d'au moins un des processus métier mis en cause par le problème.
 - Elle est référencée par au moins un traitement, fonction ou rapport mis en cause par le problème.
 - Plusieurs instances de la classe sont mises en cause dans une instance typique du problème.

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. P. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

9

9

Base d'analyse *Attributs et clés*

- Un attribut correspond à une caractéristique d'une entité représentée par une valeur (contrainte par un type).
- Deux entités d'une même classe ont les mêmes attributs.
- Une clé est un sous-ensemble des attributs d'une classe tel que toute entité se distingue des autres entités de la classe sur la base des valeurs des attributs de la clé.
- Toute classe possède au moins une clé.

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. P. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

10

10

Base d'analyse Associations

- Une association est une classe représentable par la mise en relation de classes (autrement dit, les attributs de la classe définie désignent des entités appartenant à d'autres classes).
- Lorsqu'une classe est hybride (attributs désignant des entités et attributs n'en désignant pas), elle est généralement considérée comme une association.

Vocabulaire (1/2)

- Les classes (qui sont des ensembles d'entités) sont souvent désignées du même nom que les entités qui les composent. Il est donc fréquent d'utiliser le mot «entité» pour désigner également la classe elle-même.
- Bien que les associations soient essentiellement des classes, de nombreux auteurs les considèrent comme une catégorie distincte, de nature différente des classes.
- Plusieurs synonymes sont utilisés pour marquer cette différence
 - pour la classe : concept, universel, type, catégorie
 - pour l'entité : particulier, noeud, individu
 - pour l'association : règle, axiome, arc

Correspondance

Modèle conceptuel	Modèle relationnel	SQL
classe	relation	TABLE
entité	tuple	ROW
attribut	attribut	COLUMN
type	type	TYPE ou DOMAIN
association	relation	TABLE ou VIEW

2025-04-07

MIR-07 : Conception relationnelle - introduction (v101b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

13

13

Vocabulaire (2/2)

- Il est souvent difficile de distinguer
 - un type (tel que défini par la logique),
 - d'une classe (tel qu'utilisée par la modélisation).
- En fait,
 - une classe comportant un seul attribut peut être considérée comme un type *scalaire*.
 - une classe comportant plusieurs attributs peut être considérée comme un type *non scalaire*.
- Ce lien structurel très fort sera exploité lorsque nous voudrons convertir un modèle conceptuel en modèle logique.

2025-04-07

MIR-02 : Conception relationnelle - introduction (v101b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

14

14

Exemple

Évaluation et gestion d'activités pédagogiques

1. Analyse
2. Synthèse
3. Solution
par détermination des dépendances fonctionnelles
4. Élaboration du schéma relationnel
5. Normalisation
6. Vérification des clés
7. Traitement des données potentiellement absentes
8. Inclusion des règles
9. Prise en compte de l'évolutivité
10. Optimisation

2025-04-07

MIR_02 : Conception relationnelle - introduction (v101b) © 2018-2025, Mifs - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

15

15

Énoncé initial

«On désire développer un système pour gérer les inscriptions aux cours dans une université. Les cours offerts sont décrits dans l'annuaire de l'université. On désire affecter les cours selon la disponibilité des professeurs, leur compétence et l'offre des cours par trimestre. Un étudiant s'inscrit à un groupe d'un cours pour un trimestre donné s'il en a complété tous les préalables. On désire également consigner la note obtenue par chacun des étudiants dans chacun des cours.»

2025-04-07

MIR_02 : Conception relationnelle - introduction (v101b) © 2018-2025, Mifs - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

16

16

<p>1. Analyse</p> <p>Identifier classes associations et traitements</p> <p><i>classe (prédicat)</i></p> <p><i>association (prédicat)</i></p> <p><i>traitement</i></p> <p><i>non retenu</i></p>	<p>On désire développer un système pour <i>gérer</i> les <i>inscriptions</i> aux <i>cours</i> dans une <i>université</i>. Les <i>cours</i> offerts sont décrits dans l'<i>annuaire</i> de l'université. On désire <i>affecter</i> les <i>cours</i> selon la <i>disponibilité</i> des <i>professeurs</i>, leur <i>compétence</i> et l'<i>offre des cours</i> par <i>trimestre</i>. Un <i>étudiant</i> s'inscrit à un <i>groupe</i> d'un <i>cours</i> pour un <i>trimestre</i> donné s'il en a complété tous les <i>préalables</i>. On désire également <i>consigner</i> la <i>note</i> obtenue par chacun des <i>étudiants</i> dans chacun des <i>cours</i>.</p>	2025-04-07 M.I.R. 02 : Conception relationnelle - introduction (v101b) © 2018-2025, M.Fr.S. - CC BY-NC-SA 4.0 Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec
--	---	--

17

<p>2. Synthèse...</p> <p><i>Examen des classes candidates (1/5)</i></p> <p>○ <i>gérer</i></p> <ul style="list-style-type: none"> • traitement (en fait, divers traitements, fonctions, rapports) ; • retenu, il faut toutefois en dériver des traitements précis. <p>○ <i>inscription</i></p> <ul style="list-style-type: none"> • classe ; • MAIS AUSSI l'action de le produire, donc un traitement • donc retenu comme classe ET comme traitement (inscrire). <p>○ <i>cours</i></p> <ul style="list-style-type: none"> • classe ; • retenu. 	2025-04-07 M.I.R. 02 : Conception relationnelle - introduction (v101b) © 2018-2025, M.Fr.S. - CC BY-NC-SA 4.0 Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec
---	--

18

2. Synthèse...

Examen des classes candidates (2/5)

○ université

- classe ;
- MAIS le système s'applique toujours à la *même* université, donc non retenu ;
- si on gérait les cours pour un réseau d'universités, ou pour des programmes multi-universitaires, la classe université serait alors pertinente).

○ annuaire

- classe
- MAIS c'est la même que cours, donc non retenu.

2025-04-07

MIR_02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. F. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

19

19

2. Synthèse...

Examen des classes candidates (3/5)

○ professeur

- classe ;
- retenu.

○ disponibilité

- relation entre professeur et trimestre.

○ compétence

- relation entre professeur et cours.

○ affecter

- traitement

2025-04-07

MIR_02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. F. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

20

20

2. Synthèse...

Examen des classes candidates (4/5)

○ offre_de_cours

- relation entre cours et trimestre.

○ trimestre

- classe ou type ? (\Rightarrow type)
- retenu.

○ étudiant

- classe ;
- retenu.

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. P. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

21

21

2. Synthèse...

Examen des classes candidates (5/5)

○ groupe (ou groupe-cours)

- classe ou type ? (\Rightarrow classe)
- retenu.

○ préalables

- relation entre cours... et cours.

○ note

- classe ou type ? (\Rightarrow classe ET type)
- retenu.

○ consigner

- traitement

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, M. P. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

22

22

2. Synthèse...

Identifier quelques requêtes intéressantes (1/2)

Découvrir ainsi d'autres relations...

- Quels cours pourraient être offerts à un trimestre donné ?
- Quels sont les cours effectivement offerts ?
- À quels cours un étudiant est-il inscrit ?
- Combien de groupes d'un même cours y a-t-il à un trimestre donné ?
- Par quels professeurs un groupe donné est-il offert ?
- ...

2. Synthèse...

Identifier quelques requêtes intéressantes (2/2)

... ou vérifier que celles déjà présentes sont suffisantes !

- Quels *cours* pourraient être *offerts* à un *trimestre* donné ?
- Quels sont les *cours* effectivement *offerts* ?
- À quels *cours* un *étudiant* est-il *inscrit* ?
- Combien de *groupes* d'un même *cours* y a-t-il à un *trimestre* donné ?
- Par quels *professeurs* un *groupe* donné est-il *offert* ?

Conclusion

- Il manque une façon de représenter l'inscription elle-même, indépendamment de l'évaluation.
- Quelle est la différence entre offert, offert et offert ?

3. Solution par détermination des dépendances fonctionnelles

Étapes

1. Dresser l'inventaire des dépendances fonctionnelles (DF).
2. Montrer que ces DF sont induites par les clés des relations auxquelles elles sont applicables.

2025-04-07

MIR-02 : Conception relationnelle - introduction (v101b) © 2018-2025, M. F. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

25

25

3. Solution possible (sans la division de la relation Inscription)

- Cours {sigle, titre}
 - sigle -> titre
 - L'*activité* identifiée par le sigle «sigle», décrite par le titre «titre», **est offerte par l'UdeS.**
- Étudiant {matriculeE, nom, adresse}
 - matriculeE -> nom
 - matriculeE -> adresse
 - L'*étudiant* identifié par le matricule «matriculeE», décrit par le nom «nom» et l'adresse est «adresse» **est admis à l'UdeS (et peut donc s'inscrire à des cours).**
- Professeur {matriculeP, nom, adresse}
 - matriculeP -> nom
 - matriculeP -> adresse
 - Le *professeur* identifié par le matricule «matriculeP», décrit par le nom «nom» et l'adresse est «adresse» **est enseignant à l'UdeS (et peut donc y offrir des cours).**

2025-04-07

MIR-02 : Conception relationnelle - introduction (v101b) © 2018-2025, M. F. S. - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

26

26

3. Solution possible (suite)

Inscription pourrait être un association si nous avions modéliser le trimestre comme une classe.

Il faudra élaborer les contraintes entre Inscription, Groupe, Résultat et Offre_de_cours.

- Inscription {sigle, trimestre, matriculeE}
 - sigle, trimestre, matriculeE -> noGroupe
 - L'*étudiant* dont le matricule est «matriculeE» **est inscrit au groupe de l'activité «activité» au trimestre «trimestre» à l'UdeS.**
- Groupe {sigle, trimestre, noGroupe, matriculeP}
 - sigle, trimestre, noGroupe -> matriculeP
 - Le *professeur* dont le matricule est «matriculeP» **enseigne au groupe «noGroupe» de l'activité «activité» au trimestre «trimestre» à l'UdeS.**
- Résultat {sigle, trimestre, matriculeE, note}
 - sigle, trimestre, matriculeE -> note
 - La *note* «note» **a été obtenue par l'étudiant** identifié par le matricule est «matriculeE» inscrit à l'activité «sigle» au trimestre «trimestre».

2025-04-07

MIR-02 : Conception relationnelle - introduction (v101b) © 2018-2025, Mir-02 - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

27

27

3. Solution possible (fin)

La façon dont les groupes sont modélisés introduit des contraintes vraisemblablement trop rigides.

Quelles sont ces contraintes ?

Comment les assouplir tout en préservant l'intégrité du modèle ?

- Préalable {sigle, siglePréalable}
 - relation totale
 - L'*activité* dont le sigle est «siglePréalable» **est préalable à l'activité «sigle».**
- Compétence {sigle, matriculeP}
 - relation totale
 - Le *professeur* identifié par le matricule «matriculeP» **est habilité à enseigner l'activité «sigle».**
- Disponibilité {trimestre, matriculeP}
 - relation totale
 - Le *professeur* identifié par le matricule «matriculeP» **est disponible au trimestre «trimestre».**
- Offre_de_cours {sigle, trimestre}
 - relation totale
 - L'*activité* dont le sigle est «sigle» **est offerte par l'UdeS au trimestre «trimestre».**

2025-04-07

MIR-02 : Conception relationnelle - introduction (v101b) © 2018-2025, Mir-02 - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

28

28

4. Élaboration du schéma relationnel

- Il est fortement recommandé de définir les prédicats des relations avant même de normaliser.
- Certaines erreurs seront ainsi directement détectées.
- Il sera plus facile de choisir les décompositions requises si nécessaire.

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

29

29

5. Normalisation (sans la division de la relation Inscription)

a) 1FN

- Y a-t-il des attributs non atomiques ?

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

30

30

5. Normalisation

b) FNBC

- Par construction, ce modèle logique relationnel est en FNBC.
- Qu'en est-il si la relation Inscription est divisée en deux ?

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, Mifs - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

31

31

5. Normalisation

c) FNPJ

- Le nouveau modèle est-il en FNPJ ?
- Pourquoi ?

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, Mifs - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

32

32

6. Vérification des clés (candidates et référentielles)

- Y a-t-il d'autres clés candidates ?
- Y a-t-il d'autres clés référentielles ?
- Y a-t-il d'autres contraintes ?

2025-04-07

MIR_07 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

33

33

7. Traitement des données potentiellement absentes

- Voir module TMR_07-Donnees-manquantes_PRE

2025-04-07

MIR_07 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

34

34

8. Inclusion des règles

- Revenir vers les parties prenantes avec le modèle et s'enquérir des règles applicables
 - a. de domaine (toujours)
 - b. de métier (sauf exception)
 - c. dites «d'affaires» (discutable)
 - d. de présentation (normalement pas)
 - e. contextuelles (non)

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MfP5 - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

35

35

9. Prise en compte de l'évolutivité

- Ceci est la partie «artistique» de l'exercice.
- Celle où il faut être en mesure d'anticiper les besoins futurs.
- Exemples : trimestre, admission...
- 😊

2025-04-07

MIR-02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MfP5 - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

36

36

10. Optimisation

- Voir les cours IGE487, IGE677, IFT677 et IFT723 ☺

2025-04-07

MIR_02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

37

37



2025-04-07

MIR_02 : Conception relationnelle - introduction (v10.1b) © 2018-2025, MFRS - CC BY-NC-SA 4.0
Département d'informatique, Faculté des sciences, Université de Sherbrooke, Québec

38

38