Bases de données

Données manquantes – *absentes, inconnues, nulles, facultatives, etc.*

TMR_07
v242a

2025-02-18

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

Plan

- Préambule
- *De quelques logiques non classiques*
- Approches par modification de la théorie relationnelle
- Approches par modélisation
- Quelle approche choisir ?
- *De la théorie relationnelle aux modèles relationnels*
- Vocabulaire
- Références

Préambule

- Pourquoi une donnée serait-elle manquante?
  - Les réponses de SPARC!
- Un modèle simple
  - proposé par Codd
- Solutions AVEC annulabilité
- Solutions SANS annulabilité
- État de l'art

Préambule
Données manquantes… selon SPARC (1/4)

1. L'information est applicable, mais la valeur n'est pas encore connue (*date de décès d'une personne vivante*).
2. L'information est inapplicable (*nombre de sommets d'un cercle*).
3. L'information existe, mais il n'est pas permis (*légalement*) de l'enregistrer (*religion d'un employé*).
4. L'information existe, mais on n'a pas les moyens de trouver la valeur (*évaluation d'un employé alors qu'il travaillait pour une organisation concurrente*).
5. L'information existe, mais elle n'est pas encore enregistrée (*en raison du manque de l'employé préposé à la saisie*).

Préambule
Données manquantes… selon SPARC (2/4)

6. L'information est enregistrée, mais pas encore disponible (*texte écrit, saisi, stocké, mais pas encore publié*).
7. L'information a été enregistrée puis supprimée (*un utilisateur ne veut plus que le nom de son conjoint soit conservé*)
8. L'information est disponible, mais en changement et donc potentiellement invalide (*solde d'un compte bancaire sur lequel une opération est en cours*).
9. L'information est disponible, mais on ne sait pas si elle est fiable (*la note d'examen non encore approuvée par le doyen*).
10. L'information est disponible, mais invalide (*si une erreur s'est produite lors du calcul de la valeur*)

Préambule
Données manquantes… selon SPARC (3/4)

11. La classe d'information est sécurisée (*les informations personnelles des professeurs ne sont pas accessibles aux étudiants*).
12. L'objet représentant l'information est sécurisé (*un utilisateur bloque l'accès à ses infos personnelles sur un réseau social*).
13. Une information est sécurisée durant un certain laps de temps (*le budget préalablement à sa communication au parlement*).
14. L'information est calculée à partir d'au moins une information manquante ou incertaine (*l'âge en fonction d'une date de naissance par ailleurs manquante*).

L'approche par « recensement » de SPARC est

- Inappropriée en regard de la définition d'un modèle relationnel
  - Les raisons de manque varient selon le contexte, la nature du problème voire la finalité de la requête. Les raisons appartiennent au domaine du problème et ne doivent pas être imposées par le modèle relationnel. Ce dernier doit cependant permettre de les définir et de les traiter.
  - Cette approche pourrait être utilisée lors de l'élaboration d'un modèle de données découlant d'un problème particulier.
- Souvent trop complexe
  - Tant pour la saisie que pour les requêtes, le nombre de cas à considérer est trop grand.

- **N**
  - L'information n'est **pas applicable**.
  - Dans ce cas, l'utilisation de l'annulabilité est à remettre en question; une bonne modélisation permet généralement d'éviter d'y avoir recours.
- **I**
  - L'information est **inconnue**.
  - Dans ce cas, l'annulabilité pourrait être légitime; la question est de savoir comment la représenter pour que cela pose le moins de problèmes possible.

Préambule
Le choix de la communauté

- L'approche de Codd s'est rapidement imposée.
- Nous retenons toutefois des approches précédentes qu'elles sont porteuses d'une connaissance utile dans de très nombreux contexte, mais qui est perdue si on utilise (exclusivement) l'approche de Codd.

- Que faire lorsqu'une donnée est manquante?

- Trois solutions classiques
  - corriger cette lacune à la source (dans la réalité, avant la collecte);
  - modifier le modèle pour en tenir compte;
  - introduire la notion d'*annulabilité* dans la théorie relationnelle.

- Avantage
  - Réduire la complexité des modèles de données
    (mais pas forcément celle des assertions sur ces modèles)
- Conséquences
  - Remplacement au sein de la théorie relationnelle de la logique classique par une logique non classique
    - Impact sur l'égalité, essentielle à la définition des opérateurs d'affectation, de restriction, de jointure, d'union...
    - Impact sur l'inférence logique (et en particulier la déduction) qui est nécessaire à la démonstration de l'exactitude des requêtes.
  - Modification du modèle relationnel pour y introduire la dénotation du manque, grâce à l'un des deux artifices suivants :
    - un marqueur NUL (une propriété des attributs) ou
    - une valeur NULLE (ajoutée à tous les domaines).

- Avantage
  - Conserver la théorie relationnelle telle quelle.
- Conséquences
  - Séparer les propositions complètes des incomplètes (par modélisation).
  - Conserver les causes de manque séparément (par modélisation).

- Dans les années 1970, la communauté de pratique a choisi la solution avec marqueur d'annulabilité.
- Depuis, de nombreux chercheurs n'ont eu de cesse de souligner les incohérences qui en découlent et ont mis au point diverses autres propositions.
- L'émergence des bases de données temporalisées rend impraticables les solutions AVEC annulabilité.
- On constatera, à la fin du présent module, qu'il possible de recourir aux solutions SANS annulabilité, tout en continuant d'utiliser SQL.

PRÉAMBULE
Et SQL ?

- En principe, le langage SQL a recours à la solution avec marqueur d'annulabilité.
- Par contre, de nombreux dialectes et le standard ISO lui-même définissent certains comportements en utilisant e concept de valeurs nulles (en particulier pour le traitement des expressions booléennes et les égalités implicites requises par les opérateurs relationnels).
- On constatera, à la fin du présent module, qu'il est néanmoins possible de recourir aux solutions SANS annulabilité, tout en continuant d'utiliser SQL.

De quelques logiques non classiques

- Quatre valeurs logiques (4VL, Belnap)
- Trois valeurs (3VL, Priest, Belnap, Kleene)
- Et SQL ?

Logique non classique
4V

- B : sur-déterminé; N : sous-détrminé

$f_{\neg}$	
T	F
B	B
N	N
F	T

$f_{\wedge}$	T	B	N	F
T	T	B	N	F
B	B	B	F	F
N	N	F	N	F
F	F	F	F	F

$f_{\vee}$	T	B	N	F
T	T	T	T	T
B	T	B	T	B
N	T	T	N	N
F	T	B	N	F

- La recommandation de Codd... mais elle n'a pas été suivie par le comité de standardisation du langage SQL.

Logique non classique
3V

P3 (Priest : – I est surdétrminée – T et I sont vraies – avec tautologie) – choix possible

¬	
T	F
I	I
F	T

∧	T	I	F
T	T	I	F
I	I	I	F
F	F	F	F

∨	T	I	F
T	T	T	T
I	T	I	I
F	T	I	F

$\rightarrow_K$	T	I	F
T	T	I	F
I	T	I	I
F	T	T	T

$\leftrightarrow_K$	T	I	F
T	T	I	F
I	I	I	I
F	F	I	T

B3 (Belnap, variante faible : I est réductrice – T seule valeur vraie – avec tautologie) – choix possible

¬	
T	F
I	I
F	T

$\wedge_+$	T	I	F
T	T	I	F
I	I	I	I
F	F	I	F

$\vee_+$	T	I	F
T	T	I	T
I	I	I	I
F	T	I	F

$\rightarrow_+$	T	I	F
T	T	I	F
I	I	I	I
F	T	I	T

$\leftrightarrow_K$	T	I	F
T	T	I	F
I	I	I	I
F	F	I	T

K3 (forte : I est sous-déterminée - T seule valeur vraie – pas de tautologie) – choix impraticable

K3 n'est donc pas représentée ici

Logique non classique SQL

- CHECK satisfait
  ssi T ou U
  comme P3

- WHERE satisfait
  ssi T
  un mélange de P3 et B3!

- Chercher la cohérence !

OR	true	false	unknown
true	true	true	true
false	true	false	unknown
unknown	true	unknown	unknown

AND	true	false	unknown
true	true	false	unknown
false	false	false	false
unknown	unknown	false	unknown

P	NOT P
true	fase
false	true
unknown	unknown

IS	true	false	unknown
true	true	false	false
false	false	true	false
unknown	false	false	true

Approches par modélisation

- Décompositions
  - PJ (projection-jointure)
  - RU (restriction-union)
- Modélisation du manque
  - selon McGovern (PJ)
  - selon Darwen (RU)
- Modélisation du manque et de sa cause
  - selon Darwen (PJ+RU)
- Voir
  - http://www.dcs.warwick.ac.uk/~hugh/TTM/Missing-info-without-nulls.pdf

Approches par modélisation
Décomposition de projection-jointure (PJ)

- Au tableau ou dans les exemples de cours!

Approches par modélisation
Décomposition de restriction-union (RU)

- Au tableau ou dans les exemples de cours!

Approches par modélisation
Modélisation du manque selon McGovern (PJ)

- Au tableau ou dans les exemples de cours!

Approches par modélisation
Modélisation du manque selon Darwen (RU)

- Au tableau ou dans les exemples de cours!

Approches par modélisation
Modélisation du manque et de sa cause selon Darwen (PJ+RU)

- Au tableau ou dans les exemples de cours!

Décomposition projet-jointure
Décomposition par restriction-union

- Au tableau !
- Voir exemples Naissance et Décès

Deux opérateurs relationnels prévus pour ça!

Semi-jointure (matching)

$R \ltimes S = (R \bowtie S) \pi R$

A	B
a1	b1
a2	b1
a3	b3
a4	b4

$\ltimes$

B	C
b1	c1
b2	c2
b3	c3
b3	c4

=

A	B
a1	b1
a2	b1
a3	b3

Semi-différence (not matching, anti-join…)

$R \ltimes S = R - (R \ltimes S)$

A	B
a1	b1
a2	b1
a3	b3
a4	b4

$\overline{\ltimes}$

B	C
b1	c1
b2	c2
b3	c3
b3	c4

=

A	B
a4	b4

- Les opérateurs MATCHING et NOT MATCHING ne sont malheuresement pas disponibles en SQL
- On se débrouillera donc avec la jointure externe (voir SQL_08).

- Les opérateurs MATCHING et NOT MATCHING ne sont malheuresement pas disponibles en SQL
- On se débrouillera donc avec la jointure externe

Quelle approche choisir ?

- Solution AVEC annulabilité 4V
  - La perte de causalité
  - La complexité logique
- Solution AVEC annulabilité 3V
  - La perte de sens et de causalité
  - La complexité logique
- Solution SQL
  - Tous les défauts de la 3V et en plus
    - L'incohérence logique (P3 vs B3)
    - Les dangers du *nul*
- Solution par modélisation
  - Le sens et la causalité
  - La flexibilité (du modèle)
  - La lourdeur des décompositions... en SQL

Quelle approche choisir ?
Solution AVEC annulabilité 4V

- Une solution scientifiquement acceptable mais qui nécessite un apport en modélisation sous peine d'encourir
  - une perte de sens (NA)
  - une perte de causalité (IN)
- La complexité logique induite par la logique 4V demeure un enjeu important

Quelle approche choisir ?
Solution AVEC annulabilité 3V

- Une solution scientifiquement acceptable mais qui nécessite un apport en modélisation sous peine d'encourir
  - une confusion entre IN et NA
  - une perte de causalité (IN)
- La complexité logique induite par la logique 3V demeure un enjeu important

Quelle approche choisir ?
Solution SQL

- Une solution peu acceptable scientifiquement en raison
  - de l'incohérence du modèle logique
- Une sollution qui nécessite de toutes façons un apport en modélisation sous peine d'encourir
  - une confusion entre IN et NA
  - une perte de causalité (IN)
- La complexité logique induite par la logique SQL demeure un enjeu d'autant plus important qu'elle comporte deux contextes interprétatifs (CHECK et WHERE).

Quelle approche choisir ?
Solution SQL : les dangers du null!

Qui sommes-nous pour prétendre maitriser une telle solution, alors que les experts d'Oracle n'y arrivent manifestement pas !

☺

De: oracle-acct_ww@oracle.com
Objet: Nom d'utilisateur de votre compte Oracle
Date: 2 octobre 2014 19:44
À: luc.lavoie@usherbrooke.ca

ORACLE

Cher/Chère NULL !,

Vous avez demandé à recevoir par email le nom d'utilisateur de votre compte Oracle.

Votre nom d'utilisateur est : **luc.lavoie@usherbrooke.ca**

Merci !

L'équipe de gestion des comptes Oracle

Mettez votre compte à jour :

- Abonnez-vous aux communications dédiées aux thèmes qui vous intéressent.
- Devenez membre des communautés Oracle.
- Pour modifier votre adresse email, votre mot de passe ou toute autre information de votre compte, cliquez sur le lien Compte en haut des pages Oracle.com.

Obtenir de l'aide

- Des questions ? Aide (page Account Help)
- Se connecter
  - Envoyer une demande d'aide
  - profilehelp_ww@oracle.com

Hardware and Software
ORACLE
Engineered to Work Together

Aide (page Account Help) | Ne pas envoyer d'email | Mentions légales | Conditions d'utilisation | Confidentialité

Quelle approche choisir ?
En pratique, que faire ?

- Utiliser le langage SQL, mais de façon disciplinée et seulement après avoir fait un maximum au niveau de la modélisation.
  - Décomposer en fonction des besoins.
  - Systématiser NOT NULL pour tous les attributs.
  - Pallier l'indisponibilité des opérateurs
    MATCHING ($\ltimes$) et NOT MATCHING ($\rtimes$)
    en utilisant les jointures externes en prenant soin toutefois de ne pas faire percoler les NULL au-delà du SELECT (donc en utilisant la fonction COALESCE dès que possible).

Que choisir ?
Les constats

1. Donnée manquante ::=
   donnée (attribut) non applicable
   | donnée (valeur) inconnue
2. La différence entre « non applicable » et « inconnue » est irréconciliable.
3. L'interprétation d'une « valeur inconnue » est dépendante de plusieurs facteurs dont
   - la cause du manque et
   - le prédicat associé au résultat.

Que choisir ?
Les règles

1. La non-applicabilité doit se refléter dans le modèle la valeur non applicable doit disparaitre du modèle.
2. Les prédicats complets (sans valeur inconnue) doivent être séparés des prédicats incomplets par décomposition PJ.
3. Aucune valeur ne peut être substituée par le modèle lui-même (pas de valeur « par défaut »).
4. Il est nécessaire que la requête détermine explicitement l'interprétation devant être donnée au manque.
5. Si cette interprétation n'est pas unique, la cause du manque doir être conservée par décomposition RU.

De la théorie aux modèles

- Pourquoi?
- Modèle de Codd I
- Modèle de Codd II
- Modèle de Date
- Modèle d'Ullman
- Modèles SQL
- Au final...

De la théorie aux modèles
Pourquoi n'y a-t-il pas un seul modèle?

- Parce qu'il n'y a pas consensus sur la bonne façon de traiter les données manquantes.
- Parce que certains sont prêts à sacrifier l'intégrité de leurs données et des résultats de leurs requêtes au profit des gains de performance (généralement éphémères et illusoires).
- Pour permettre d'intégrer de nouveaux résultats théoriques facilitant la modélisation et l'exploitation de données.

Il faut cependant être très prudent avant d'introduire un nouveau modèle, car tout mauvais modèle dès lors qu'il est utilisé acquiert une latence ÉNORME.

De la théorie aux modèles
Modèle de Codd I

- Transposition directe de la théorie avec les exceptions suivantes
  - marqueur «nul»
  - logique trivaluée
  - pas d'attributs de type relation

- Transposition directe de la théorie avec les choix suivants
  - extension de tous les types avec «non applicable» et «nul»
    - corolairement, une logique quadrivaluée (Belnap)
  - pas d'attributs de type relation

- Transposition directe de la théorie, conséquemment
  - pas de marqueur nul ni de valeur nulle
  - logique bivaluée
  - intégration des relations dans le système de typage (donc ajout des opérateurs *tclose*, *wrap* et *unwrap*)

De la théorie aux modèles
Modèle d'Ullman

- Transposition de la théorie relationnelle à l'aide de collections
  - marqueur «nul»
  - logique trivaluée
  - possibilité de doublons dans les relations
  - non équivalence entre relation et prédicat

- Transposition de la théorie relationnelle à l'aide de collections et de listes
  - marqueur «nul»
  - logique trivaluée
  - possibilité de doublons dans les relations
  - les attributs d'un tuple sont ordonnés et peuvent être anonymes, voire synonymes (!!!)
  - possibilité d'attributs non typées
  - non équivalence entre relation et prédicat, tuple et propostion.

Au final

- Nous maintenons la position adoptée en TRM_01 :
  - Pour l'exposé des principes relationnels, nous utiliserons toujours le modèle de Date.
  - Pour la programmation SQL, nous présenterons des techniques permettant d'être aussi proche que possible du modèle de Date, en indiquant les écarts possibles en fonction du modèle SQL ISO 9075:2016.

Les colles du prof

- Reclasser les 14 cas recensés par SPARC selon les catégories N, I et X.
- Faire le lien entre les catégories N, I, X et les trois solutions permettant de traiter les valeurs manquantes (corriger la source, modifier le modèle, introduire le concept d'annulabilité).
- Conclure en statuant sur la nécessité (ou non) du concept d'annulabilité.
- Quels sont les modèles relationnels utilisés en cours?

Références (1)

- Une théorie [mathématique] est un ensemble d'affirmations dont certaines sont des axiomes et les autres des théorèmes démontrables à partir de ces axiomes au moyen de règles d'inférence [exprimée à l'aide de la] logique.
  - http://fr.wikipedia.org/wiki/Théorie
- Un modèle est une représentation conforme à une théorie.
  - http://fr.wikipedia.org/wiki/Modèle
- Un langage (formel) est un formalisme permettant de décrire des propositions sémantiquement interprétables en termes d'un modèle.
  - http://fr.wikipedia.org/wiki/Langage_formel
- Une théorie peut être à l'origine de plusieurs modèles, un modèle de plusieurs langages, un langage de plusieurs dialectes.
- Pour en savoir plus sur le calcul des prédicats :
  - http://fr.wikipedia.org/wiki/Calcul_des_prédicats

Références (2)

- Théorie relationnelle
  - E.F. Codd. 1990.
    The Relational Model for Database Management: Version 2.
    Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc.
  - C.J. Date, H. Darwen. 2007.
    Databases, types and the relational model: the third manifesto.
    Reading, Mass.: Addison-Wesley.
  - F. de Sainte Marie. 2013.
    Bases de données relationnelles et normalisation : de la première à la sixième forme normale.
    ftp://ftp-developpez.com/fsmrel/basesrelationnelles/normalisation/normalisation.pdf
  - H. Darwen. 2006.
    How To Handle Missing Information Without Using NULL.
    http://www.dcs.warwick.ac.uk/~hugh/TTM/Missing-info-without-nulls.pdf
- Manuels classiques
  - [C. J. Date 2004], chapitre 3.
  - [Elmasri and Navathe 2004], chapitre 4.
  - [Elmasri and Navathe 2011], chapitre 3.
  - [Elmasri and Navathe 2016], chapitre 8.
  - [Ullman and Widom 2008], chapitre 3.

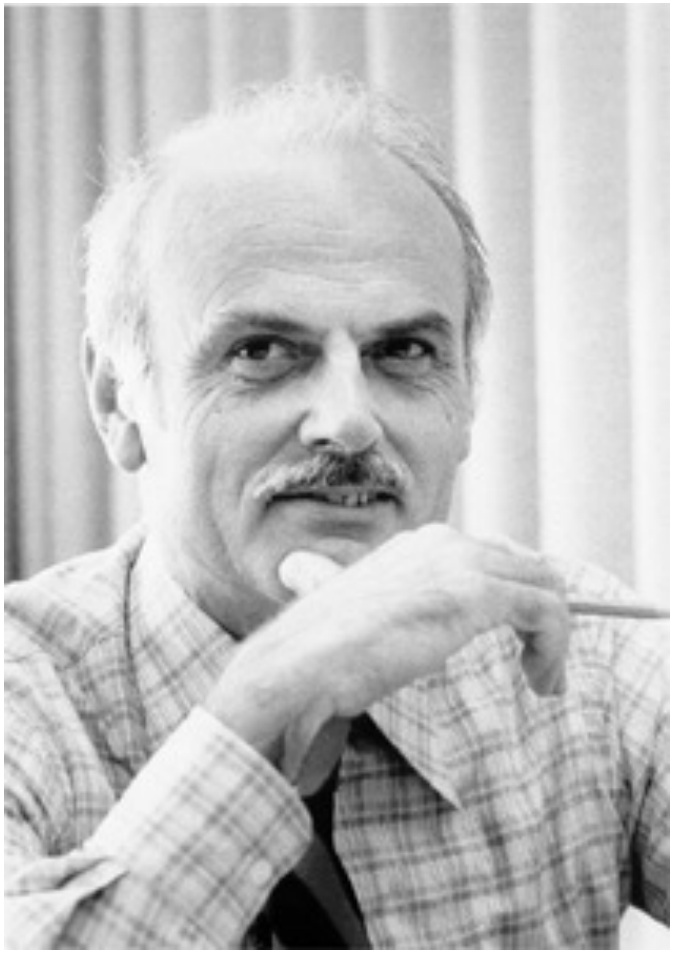
Références (3)

- Théorie des Types
  - Luca Cardelli, Peter Wegner (1985)
    On understanding types, data abstraction, and polymorphism.
    ACM Computing Surveys 17, 4; pp. 471–523.
    DOI:http://dx.doi.org/10.1145/6041.6042
  - Chris J. Date (2016)
    Type inheritance and relational theory: subtypes, supertypes, and substitutability.
    O'Reilly Media, Sebastopol, CA.
    ISBN:978-1-4919-5999-2

Autres sources

- Une synthèse des conséquences du NULL en SQL
  - https://en.wikipedia.org/wiki/Null_(SQL)
- Codd et Date débattent du sujet
  - http://web.archive.org/web/20100531071357/http://www.dbdebunk.com/page/page/1706814.htm

Edgar Frank Codd



https://en.wikipedia.org/wiki/Edgar_F._Codd

Christopher J. Date

Photo of Chris Date by Douglas Robertson, Edinburgh
https://en.wikipedia.org/wiki/Christopher_J._Date