Contrôle terminal de BDBIO

UCBL - Bases de données pour la bioinformatique - 2017 / 2018

Documents papier autorisés. Anonymisez votre copie. Durée : 1h30.

Villa *Abondance*, le couple Glouton organise une cérémonie en l'honneur de leur prix prestigieux concernant la médecine personnalisée via les signatures génétiques. Mais au cours de la soirée, quelqu'un subtilise et dévore le gâteau, ce qui gâche complètement la cérémonie ! Une enquête est menée, et vous êtes chargé.e de la résoudre : parmi les six invité.e.s, qui a dévoré le gâteau, dans quelle pièce, à quelle heure, et qui sont ses complices ?

Vos assistants ont déjà collecté des informations sur la soirée. Évidemment, vos assistants auraient pu s'accorder pour vous présenter les informations avec le même modèle, mais ça aurait été trop facile.

- Le premier assistant a produit **deux tables relationnelles**. La première table contient des informations sur les six personnes présentes, tandis que la seconde liste leurs déclarations, c'est à dire le lieu où chaque invité.e a déclaré se trouver pour chaque heure de la soirée. Par exemple, la personne identifiée par 3 était dans la cuisine à 19h (jusque 20h) puis dans la véranda à 20h (jusque 21h), etc. Alors oui, c'est plutôt suspect de rester seul.e une heure entière dans la salle de bains...
- La deuxième source est un **document XML** qui fournit la localisation de chaque personne d'après son téléphone (de gros moyens ont été déployés pour un malheureux gâteau...). C'est une information fiable (contrairement aux déclarations des personnes). Le lieu 1 représente la véranda, le lieu 2 pour le salon, le lieu 3 correspond à la salle de bains et le lieu 4 est la cuisine. L'assistant qui a compilé cette source, visiblement feignant mais anglophile, a utilisé l'initiale du nom de la personne et l'heure en anglais...
- La troisième source est un **graphe RDF**, qui présente les témoignages ainsi que le repas de chaque invité.e. Le préfixe x correspond à l'espace de nom pour l'enquête. Si X est témoin pour Y, c'est que X certifie que les déclarations de Y sont vraies. Vous doutez quant à l'utilité des informations sur les repas pour résoudre l'enquête... si elles se révèlent inutiles, vous blâmerez votre assistant.

Votre intuition vous trompe rarement : en recoupant ces sources, vous devriez résoudre l'enquête. Le ou la coupable était forcément seul.e dans une pièce pour dévorer le gâteau, et sa déclaration est probablement fausse (par rapport à la géolocalisation de son téléphone). Son ou ses complices ont distrait les autres invité.e.s. Leur identification sera simple, puisqu'ils ont témoigné en faveur du coupable afin de couvrir le méfait.

Personnes (idP, titre, nom, metier)
Declarations (#idP, piece19h, piece20h, piece21h)

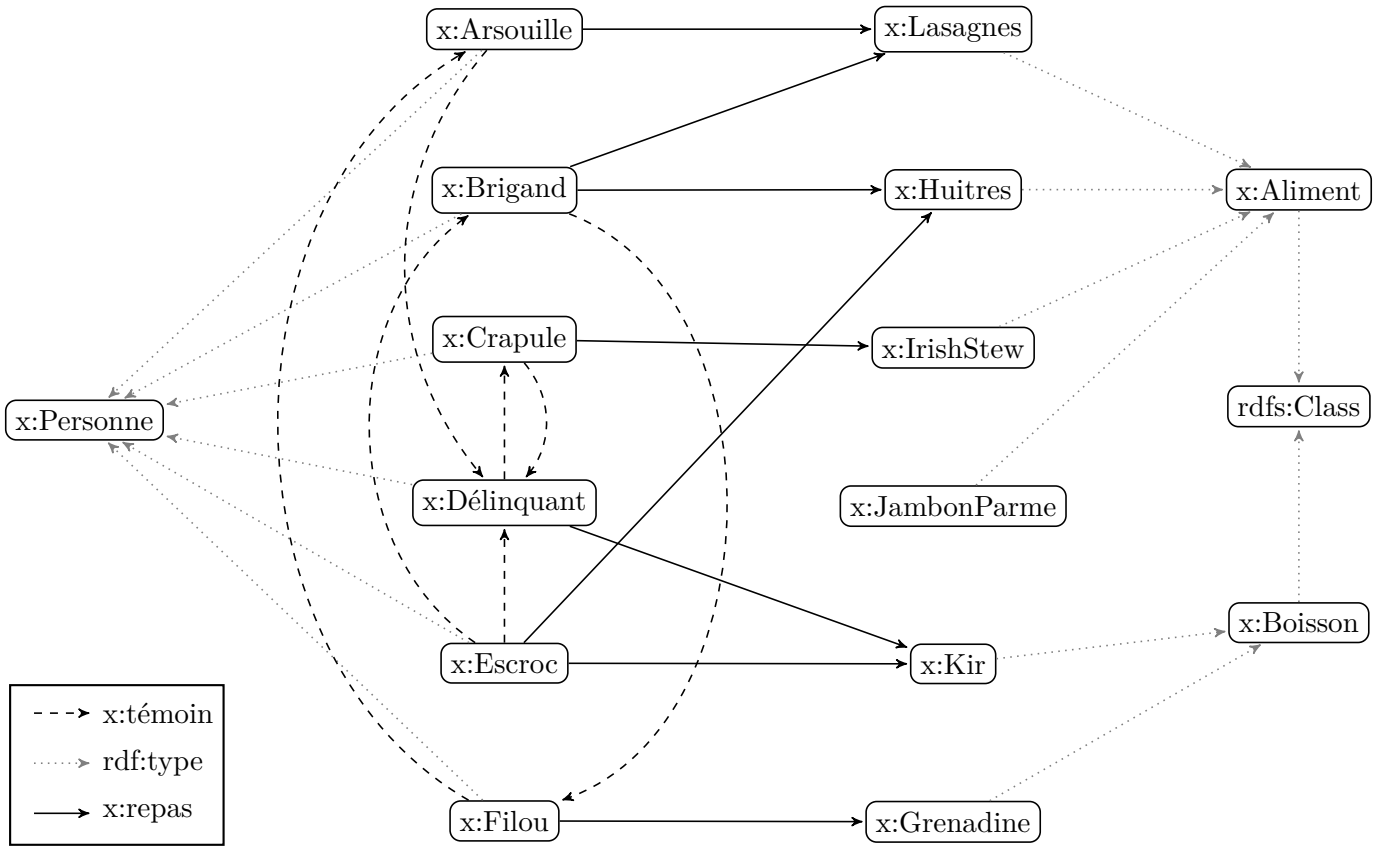
idP	titre	nom	metier
1	Mme	Arsouille	alchimiste
2	Mr	Brigand	banquier
3	Mme	Crapule	comptable
4	Mr	Délinquant	devin
5	Mme	Escroc	experte
6	Mr	Filou	fiscaliste

Relation Personnes

idP	piece19h	piece20h	piece21h
1	véranda	véranda	véranda
2	véranda	cuisine	salon
3	cuisine	véranda	sdb
4	salon	salon	véranda
5	salon	salon	cuisine
6	sdb	véranda	véranda

Relation Declarations

```xml
<geolocalisations>
 <lieu id="1">
  <personnes heure="7pm">A, B</personnes>
  <personnes heure="8pm">A, F</personnes>
  <personnes heure="9pm">A, D, F</personnes>
 </lieu>
 <lieu id="2">
  <personnes heure="7pm">D, E</personnes>
  <personnes heure="8pm">C, D, E</personnes>
  <personnes heure="9pm">B</personnes>
 </lieu>
 <lieu id="3">
  <personnes heure="7pm">F</personnes>
  <personnes heure="8pm">B</personnes>
  <personnes heure="9pm">C</personnes>
 </lieu>
 <lieu id="4">
  <personnes heure="7pm">C</personnes>
  <personnes heure="8pm"></personnes>
  <personnes heure="9pm">E</personnes>
 </lieu>
</geolocalisations>
```



Exercice 1 Compréhension des sources (6 points)

Répondez aux questions en cochant les réponses qui vous semblent correctes.

1. La relation Personnes est-elle bien modélisée ?
   ☐ Non, elle n'est pas en première forme normale.
   ☐ Non, elle n'est pas en deuxième forme normale.
   ☐ Non, elle n'est pas en troisième forme normale.
   ☐ Évidemment ! Vous avez confiance dans le travail de votre assistant.

2. Quel problème de modélisation détectez-vous pour la relation Declarations ? Proposez une solution à ce problème.

3. Si l'on voulait stocker, au niveau de la source relationnelle, les liens entre les personnes (e.g., "mariage", "cousin.e", "collègue de travail"), comment ferait-on ?
   ☐ Ajout d'attribut(s) dans la relation Personnes.
   ☐ Création d'une nouvelle table, avec un `idP` comme clé primaire.
   ☐ Création d'une nouvelle table, avec deux `idP` comme clé primaire.
   ☐ C'est réalisable en relationnel, mais aucune des solutions précédentes ne convient.
   ☐ Ce n'est pas réalisable en relationnel, il faut un autre modèle de données.

4. Si l'on construisait une DTD pour le document XML, quelles contraintes seraient correctes ?
   ☐ `<!ELEMENT geolocalisations (lieu)>`
   ☐ `<!ELEMENT lieu EMPTY>`
   ☐ `<!ELEMENT personnes (#CDATA)>`
   ☐ `<!ATTLIST lieu id #REQUIRED>`

5. Si l'un de vos assistants propose de transformer le document XML pour le stocker dans un SGBD acceptant des documents JSON, vous répondez...
   ☐ Un document JSON pourrait correspondre à un lieu. L'une de ses propriétés serait une liste des personnes ayant fréquenté ce lieu et une autre propriété serait une liste pour les heures.
   ☐ Un document JSON pourrait correspondre à une personne. L'une de ses propriétés serait un tableau de sous-documents, qui contiendrait chacun un lieu et une heure.
   ☐ Certains documents JSON pourraient correspondre à une localisation (heure et lieu) et d'autres à une personne. Dans les documents de localisation, une propriété serait une liste avec les noms de personnes.
   ☐ Les documents JSON pourraient correspondre soit à une heure, soit à une personne, soit à un lieu. Les relations se feront donc par référence, ce qui améliore les performances.
   ☐ Il y a déjà trois modèles différents, ça commence à bien faire !

6. Quelles contraintes sont valides sur les données RDF ?
   ☐ `e:repas rdfs:range f:Aliment`
   ☐ `e:repas rdfs:isDefinedBy rdf:Property`
   ☐ `e:temoin rdfs:domain rdfs:Resource`
   ☐ `e:temoin rdfs:range e:Personne`

Exercice 2 Interrogation des sources (4 points)

Avant de vous lancer dans l'intégration des sources, vous décidez de vous "échauffer" en écrivant quelques requêtes sur chacune des sources.

1. En SQL : le nom des personnes qui n'étaient pas dans le salon à 19h ou dont le métier contient un *e*.

2. En SQL : lister le nombre de personnes dans chaque pièce, pour l'horaire 21h. Le résultat sera ordonné par nombre décroissant.

3. En XPath ou XQuery : les informations (lieu, heure, et nombre de personnes) où se trouvait la personne D, mais uniquement quand elle n'était pas avec la personne E. La méthode *contains(x, y)* cherche si la chaine x contient la chaine y. La méthode `tokenize(x, y)` coupe la chaine x selon le délimiteur y, et retourne une liste de tokens.

4. En SPARQL : les personnes qui ont mangé (un aliment) ou qui ont témoignés pour *Crapule*

Exercice 3 Intégration de données (10 points)

Pour faciliter la découverte des malfaiteurs, vous décidez d'intégrer les sources de données dans un seul SGBD (architecture centralisée type entrepôt).

1. Proposez un schéma relationnel qui servira de schéma global pour l'entrepôt. Il doit inclure tous les concepts présents dans les trois sources.

2. On considère que vous avez intégré les données de la source 1 (relationnelle) dans votre entrepôt. La source 2 (XML) contient le même type d'informations que la source 1 (lieux et heures pour chaque personne). Expliquer comment vous feriez pour intégrer cette seconde source).

3. Écrire les mappings entre la source 3 (RDF) et votre schéma global sous forme d'un programme informatique (exécuté au niveau d'un chargeur ETL). On considère que les sources 1 et 2 ont déjà été intégrées dans l'entrepôt. Vous utiliserez les langages de requêtes appropriés pour extraire les données de la source et pour peupler la BD globale. Le reste du programme sera codé en pseudo-langage (syntaxe libre, mais suffisamment explicite pour que le programme soit implémentable). Pour le niveau de détail, utilisez des appels de fonctions pour simplifier le code (e.g., si vous devez trier un tableau, écrivez `tabSorted = sort(tab)` accompagné d'un commentaire mais n'écrivez pas un algorithme complet de tri de tableau !). Utilisez des commentaires, par exemple pour expliquer comment vous résolvez les conflits.

Exercice 4 Bonus - votre solution à l'enquête (1 point)

1. Alors, qui est le ou la coupable ? Cocher son idP.
   ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6

2. Dans quelle pièce a été commis le méfait ?
   ☐ Véranda ☐ Salon ☐ Cuisine ☐ Salle de bains

3. À quelle heure a été commis le méfait ?
   ☐ 19h ☐ 20h ☐ 21h

4. Enfin, qui sont le ou les complices ? Cocher le(s) idP.
   ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6