

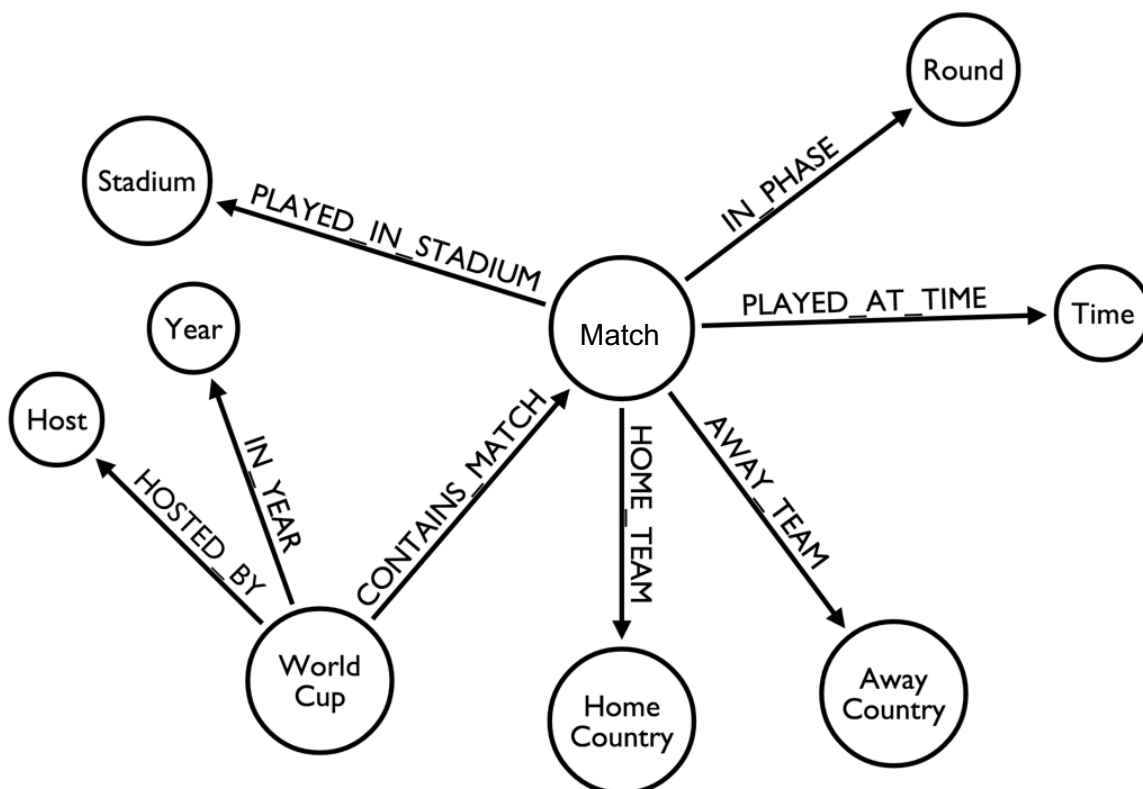
Examen de Modèles et langages pour les méga-données

CONSIGNES :

- Vous utiliserez 2 copies séparées :
 - o une pour les parties 1 et 2
 - o et une pour la partie 3
- Documents papiers autorisés, documents électroniques interdits

1. Partie 1 : BD orientée Graphe (8 points)

On considère la modélisation de graphe Neo4j suivante, modélisant des coupes du monde de football.



Une coupe du monde (World Cup) est organisée par (*hosted_by*) un pays (Host) pendant (*in_year*) une année (Year). Elle est composée (*contains_match*) de matchs (Match). Un match est joué (*played_in_stadium*) dans un stade (Stadium) à une date donnée (*played_at_time* -> Time) durant (*in_phase*) une phase donnée (Round) : phase de poule, 1/4 de finale, 1/2 finale,... Un match oppose un pays hôte (*home_team* -> Home Country) et un pays considéré comme pays jouant à l'extérieur (*away_team* -> Away country).

Les propriétés des nœuds sont les suivantes :

WorldCup, Host, Home Country, Away Country : name

Round : type (poule A, poule B, poule C, poule D, poule E, 1/8 finale, ¼ finale, ½ finale, petite finale, finale)

Time : date

Year : year

Stadium : name, city

Les nœuds match ne possèdent pas d'autre propriété que leur identifiant.

[Q1] (1,5 point). **Insertion.** Insérez les données suivantes :

La coupe du monde 2018 (WorldCup.name : Fifa World Cup 2018) a eu lieu en 2018 (Year.year : 2018) en Russie (Host.name : Russia).

Elle a eu comme match Suède (Home country.name : Sweden) / Angleterre (Away country.name : England) joué au stade Solidarnost Arena (Stadium.name : Solidarnost Arena ; Stadium.city : Samara) le 7 juillet 2018 à 17h (Time.date : 7/07/2018 17 :00 :00).

[Q2] (3 points). **Modélisation.** Modifiez la modélisation pour inclure :

- Les joueurs membres de chaque équipe
- Le score du match et l'équipe gagnante
- Les buts marqués (dans un match et par un joueur). Les buts sont marqués à une certaine minute du match.

[Q3] (3,5 points). **Interrogation.** En considérant la modélisation de la question Q3, donner les requêtes Cypher permettant de donner:

- a) Le nombre de match joué au stade Solidarnost Arena
- b) Les équipes ayant gagné un match dans la poule A de la coupe du monde 2018
- c) Les pays qui ont été en ½ finale en 2014 et en 2018
- d) Les joueurs buteurs de la finale de la coupe du monde 2018.



2. Partie 2 : Mapping Objet-Relationnel (7 points)

L'office du tourisme de la ville de Champagnac-en-Cambrousse souhaite promouvoir le tourisme dans la ville, en mettant en avant sur son site web les différents types d'hébergement possibles, ainsi que les différents « packs de visites » qu'il est possible de réserver en même temps qu'un hébergement.

Les hébergements (*idHébergement*, *adresse*, *description*) peuvent être soit des Gîtes (*capacité*, *nombreEpis*, *nomProprietaire*, *tarif*), soit des hôtels (*nombreEtoiles*), soit des campings (*nombreEtoiles*).

Les clients (*idClient*, *nomClient*, *prenomClient*, *adresseClient*) effectuent des réservations (*idReservation*, *dateReservation*, *dateDebut*, *dateFin*), qui concernent soit un gîte, soit une chambre d'hôtel, soit un emplacement de camping.

Les chambres d'hôtels sont décrites par leur *capacité*, un *numéro*, une *description* et un *tarif*. De façon similaire, les emplacements des campings sont décrits par une *capacité*, un *numéro* et un *tarif*. A des fins de simplification on considère que la tarification des différents logements ne bouge pas pendant l'année (pas de tarification basse/moyenne/haute saison).

Afin d'encourager les visites guidées de la ville et de remplir les musées, l'office du tourisme a mis en place des PackVisite (*idPack*, *description*), composés d'entrées dans différents musées (*idMusee*, *nomMusee*, *horaires*) et/ou de visites guidées (*idVisite*, *description*, *joursvisite*), le tout proposé à un tarif préférentiel (*tarif*).

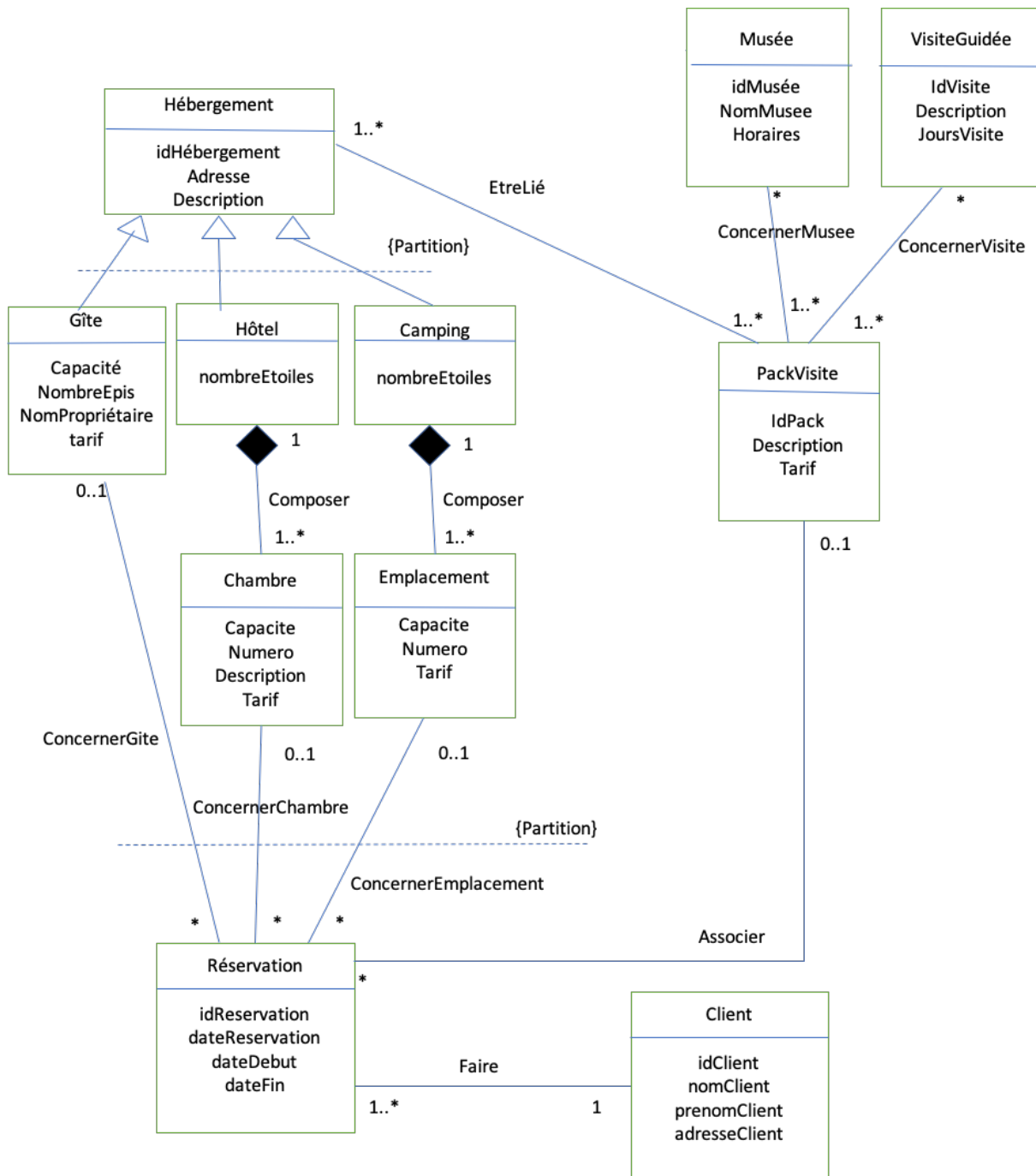
En fonction des hébergements, un choix de packsVisite est proposé aux clients (association *EtreLié*). Les clients peuvent ensuite, s'ils le souhaitent, compléter leur réservation d'hébergement avec un de ces packs (association *Asssocier*).

La modélisation sous forme de diagramme de classes obtenue est présentée en p.4 du sujet.

[Q1] (3 points) Donnez le schéma relationnel complet issu du diagramme de classes, en considérant une **traduction descendante de l'héritage**. Soyez clairs lors de l'expression de vos clés étrangères, en indiquant clairement quelle clé primaire elles référencent.

[Q2] (2 points) Pour chaque table, listez les éventuelles contraintes d'unicité, de non nullité ou CHECK que l'on peut exprimer au moment de sa création (sans nécessairement donner le code SQL associé).

[Q3] (2 points) Quelles sont les contraintes présentes sur le diagramme de classes qui ne sont pas encore traduites ? comment faudrait-il les traduire (sans forcément donner le code, mais soyez précis !)? Commentez notamment le cas de l'héritage.

Diagramme de classe :

3. Partie 3 : Programmation MapReduce (5 points)

Considérons le code Python ci-dessous lié à l'environnement MapReduce vu en TP :

```
#!/usr/bin/env python
import mincemeat

data = ["A|1000", "B|500", "A|500", "C|1500", "B|500"]

datasource = dict(enumerate(data))

def mapfn(k, v):
    line = v.split('|')
    yield line[0], line[1]

def reducefn(k, vs):
    result = 0
    for i in vs:
        result = result + int(i)
    return result/len(vs)

s = mincemeat.Server()
s.datasource = datasource
s.mapfn = mapfn
s.reducefn = reducefn

results = s.run_server(password="changeme")
print(results)
```

[Q1] Indiquez le résultat produit par le programme ci-dessus.

[Q2] Expliquez ce que renvoient les fonctions `mapfn` et `reducefn`.