

Projet de conception BDD

Un lieu culturel

Partie 1 - Conception (MCD)

1. Extraction des propositions pertinentes de l'énoncé

Le centre possède 3 salles

Une salle a un nombre de places

Événement a lieu dans une salle

Événement a lieu dans les différentes salles

Un événement occupe une salle au minimum

Un événement occupe 3 salles au maximum

Spectacle est un type d'événement

Spectacle est une unité de production culturelle

Un spectacle a un nombre de places

Un événement propose un spectacle

Un événement propose plusieurs spectacles

Un artiste produit un événement

Un artiste a un paiement prédéfini

Un événement à un tarif

Un événement a plusieurs tarifs

Artiste reçoit un règlement après l'événement

L'événement a un statut (passé/annulé)

Un artiste a un statut de règlement

Un artiste a un nom et un métier

Les abonnés ont un compte personnel

Les abonnés font une réservation

La réservation a un statut (réglé/non réglé)

Le ticket a un statut (réservé/revendu/remboursé)

Bénévole a un statut (générique/spécifié)

Bénévole a une durée de mission

Bénévole a un rôle (responsable/subordonné)

Bénévole a une équipe

Salarié a un poste

L'association a des membres de la présidence

Salarié a un salaire mensuel prédéfini

Salarié a un statut (responsable/subordonné)

Un membre de la présidence a un statut (responsable/subordonné)

BDD a une liste de tous évènements/spectacles/artistes (historique)

2. Entités, attributs, associations et cardinalités

- Une salle de spectacle est une entité ayant pour attribut : le numéro de la salle, le nombre de places, le statut de la salle. Cette entité est associée aux entités : spectacle. Une salle de spectacle peut accueillir un ou plusieurs spectacles.
- Un événement est une entité ayant pour attribut : l'id de l'événement, le statut de l'événement, tickets vendus. Cette entité est associée aux entités : spectacle, bénévole, équipe. Un événement peut donner lieu à un ou plusieurs spectacles. Un événement peut avoir zéro ou plusieurs bénévoles. Un événement peut avoir zéro ou plusieurs équipes de bénévoles.
- Un spectacle est une entité ayant pour attributs : id du spectacle, type de spectacle, statut de spectacle, la date et l'heure du spectacle, nombre de tickets. En association avec : événement, salle de spectacle, artiste, ticket, tarif, réservation. Un spectacle fait partie d'un événement. Un spectacle se déroule dans une et une seule salle. Un spectacle accueille un ou plusieurs artistes. Un spectacle vend zéro ou plusieurs tickets. Un spectacle peut avoir un ou plusieurs tarifs. Pour un spectacle, zéro ou plusieurs réservations sont faites.
- Artiste est une entité ayant pour attributs : l'id de l'artiste, le métier. En association avec : Spectacle, Personne. Un artiste peut participer à un ou plusieurs spectacles. Un artiste est une personne.

- Cachet est une entité type association avec comme attribut : cachet de l'artiste. En association avec : Artiste et Spectacle. Un cachet est touché par un artiste intervenant dans un spectacle.

- Reservation est une entité type association ayant pour attribut : l'id réservation, statut de réservation. En association avec : Abonne, Spectacle, Ticket. La réservation peut être faite/modifiée par un abonné. La réservation peut être payée en ligne en avance. Une réservation est faite par un abonné et pour un ou plusieurs spectacles. Une réservation correspond à un ticket.

- Abonné est une entité ayant pour attribut : l'id de l'abonné. En association avec : Personne, Réservation. Un abonné fait une réservation en ligne.

- Ticket est une entité type association ayant pour attributs : l'id du ticket, le prix, le statut du ticket, l'horaire d'achat du ticket. En association avec : Réservation, Spectacle, Personne. Un ticket peut être réservé en ligne. Un ticket peut correspondre à une réservation. Un ticket est acheté pour un spectacle, et par une personne.

- Bénévole est une entité ayant pour attributs : l'id de bénévole, la durée de la mission, le domaine d'activité, le statut de responsabilité. En association avec : personne, événement, équipe. Un bénévole est une personne. Un bénévole prend part à un ou plusieurs événements. Un bénévole peut faire partie d'une équipe de bénévoles et les diriger.

- Équipe de bénévoles est une entité ayant pour attributs : l'id de l'équipe de bénévoles. En association avec : Benevole. Une équipe est composée de plusieurs bénévoles.

- Salarié est une entité ayant pour attributs : l'id de salarié, le poste, le salaire mensuel. Association avec : Personne. Un salarié est une personne. Un salarié peut avoir un statut de responsable d'un autre salarié (il s'agit de l'auto dépendance au sein d'une entité).

- Personne est une entité ayant pour attributs : l'id de la personne, le nom, le prénom, la date de naissance, l'adresse, le numéro de téléphone, l'adresse email, statut. En

association avec : Artiste, Bénévole, Salarié, Membre, Abonné. Une personne peut être ou non un artiste/bénévole/salarié/membre.

- Membre de l'association est une entité ayant pour attribut : l'id du membre, le poste. En association avec : Personne. Un membre de l'association est une personne.
- Tarif : entité type association, a pour attributs : type de tarif. En association avec : Spectacle. Un tarif est effectif pour un ou plusieurs spectacles.

3. Dépendances fonctionnelles

Salle

NumSalle

NombrePlaces

StatutSalle

Evénement

IDEvenement

StatutEvenement

Spectacle

IDSpectacle

TypeSpectacle

StatutSpectacle

DateTimeSpectacle

Artiste

IDArtiste

Metier

Reglement (artiste+employé)

IDReglement

StatutReglement

MontantReglement

DateReglement

Paielement (ticket+reservation)

IDPaielement

StatutPaielement

DatePaielement

Réservation

IDReservation

DateReservation

Abonné

IDAbonne

Ticket

IDTicket

StatutTicket

Tarif

TypeTarif

Bénévole

IDBenevole

DureeMission

DomaineActivite

StatutResponsabilite

Équipe

IDEquipe

Salarié

IDSalarie

Poste

SalaireMensuel

Personne

IDPersonne

Nom

Prenom

DateDeNaissance

Adresse

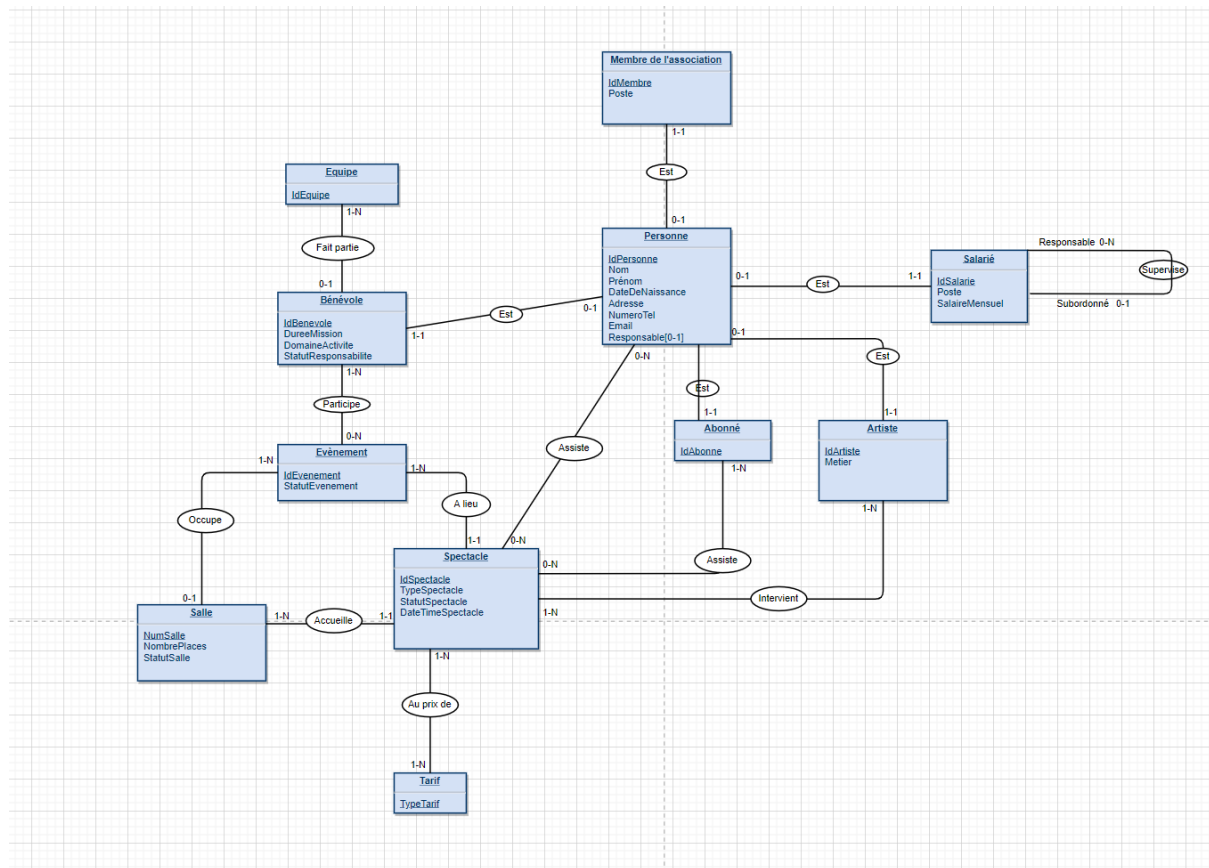
NumeroTel

Email

Statut

Responsable [0-1]

4. Schéma Entité-Association (cardinalité Merise)



5. Normalisation

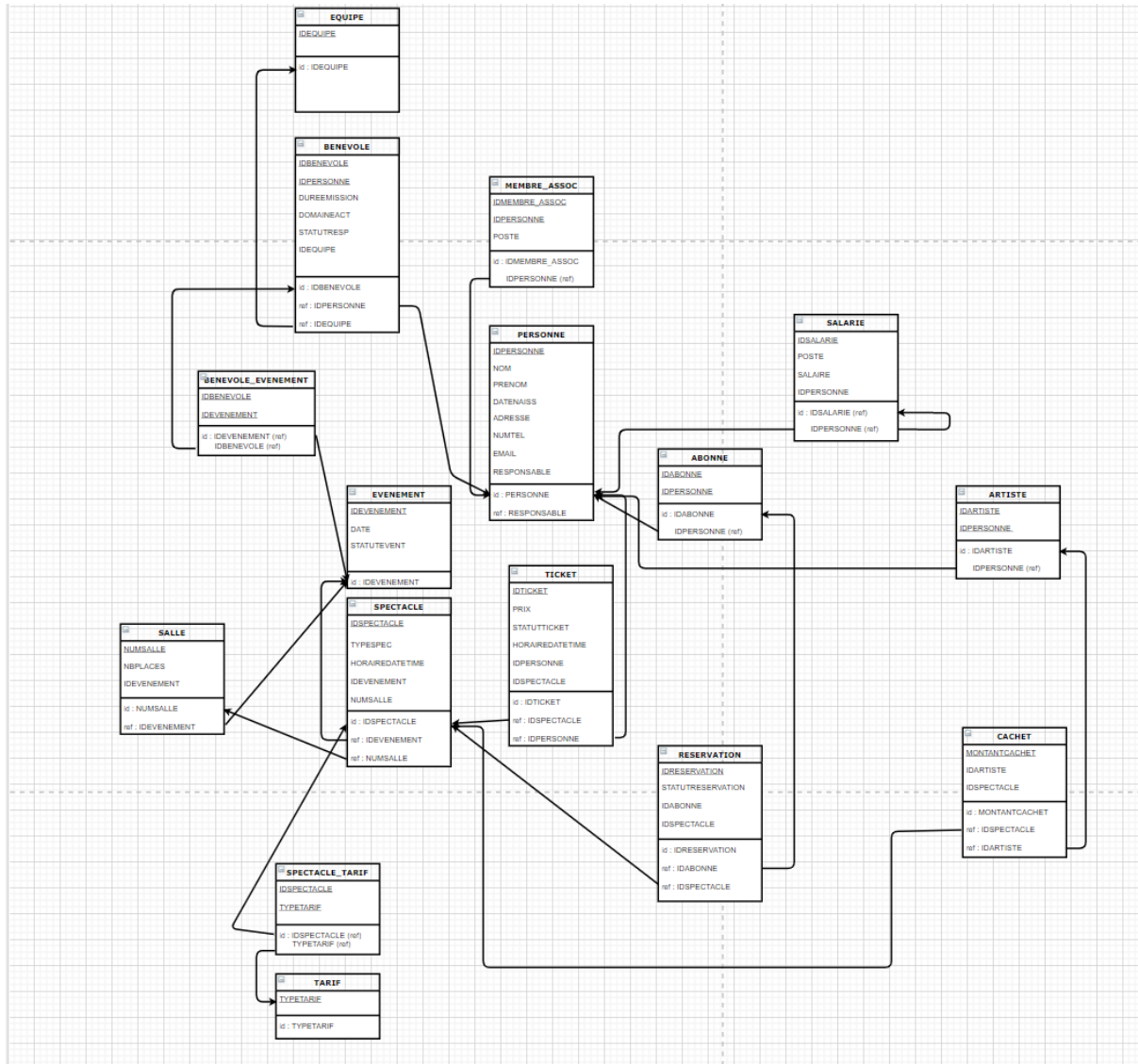
Le schéma est effectué en 1ère forme normale car chaque attribut a une seule donnée associée à lui.

Le schéma est en 2ème forme normale car chaque attribut représente une clé primaire minimale dans chaque entité et les entités ayant des attributs non-minimaux ont été décomposées en entités séparées. Par exemple, une entité Bénévole ayant pour attribut l'id EquipeBénévole a été décomposée en deux entités Bénévole et EquipeBénévole. Un bénévole n'appartient pas à une équipe en permanence car chaque équipe est formée selon l'événement.

Le schéma est effectué en 3ème forme normale car chaque attribut de chaque entité dépend directement d'une clé primaire de son entité. Par exemple, l'entité Ticket possédait un attribut tarif mais elle était décomposée en deux entités Tarif et Ticket car tarif dépend du spectacle directement et non de ticket.

Partie 2 - Production (MLD)

1. Schéma de table



Nous avons utilisé la version 8.0 de MySQL pour créer la base de données.

Partie 3 - Requêtes

1. Liste des bénévoles et de leur âge, ainsi que les événements auxquels ils ont participés

```
SELECT nom, prenom, datenaiss, idevenement, (DATEDIFF
(YEAR,CURDATE()),(datenaiss))) AS age

FROM PERSONNE, BENEVOLE, EVENEMENT

WHERE PERSONNE.idpersonne = BENEVOLE.idpersonne

AND BENEVOLE.idevenement = EVENEMENT.idevenement;
```

2. Montant des prestations réglées aux artistes entre deux dates données

```
SELECT DISTINCT(cachetartiste), nom, prenom

FROM PERSONNE, ARTISTE, CACHET

WHERE PERSONNE.idpersonne = ARTISTE.idpersonne

AND ARTISTE.idartiste = CACHET.idartiste

AND (DATEDIFF ('2019-12-21', '2020-06-13')) AS periode ;
```

3. Afficher le nombre de places disponibles pour un événement

```
SELECT COUNT(DISTINCT(idticket))

FROM EVENEMENT, SPECTACLE, TICKET

WHERE EVENEMENT.idevenement = SPECTACLE.idevenement

AND SPECTACLE.idspectacle = TICKET.idspectacle

AND statut.ticket = 'disponible';
```

4. Afficher le nombre de minutes, d'heures, de jours, de mois et d'années avant le début d'un événement

```

SELECT idspectacle, horairedatetime, idevenement
FROM EVENEMENT, SPECTACLE
WHERE EVENEMENT.idevenement = SPECTACLE.idevenement
AND TIMESTAMPDIFF(NOW(), (horairedatetime))

Order by horairedatetime ASC

LIMIT 1;

```

5. Afficher les événements par ordre de meilleures ventes, avec les sommes collectées au total d'une part, et le bénéfice réalisé d'autre part (après paiement des artistes)

```

SELECT idevenement, (COUNT(idticket) * typetarif) AS SommesCollectees,
((COUNT(idticket) * typetarif) - SUM(cachetartiste)) AS Benefice
FROM CACHET, SPECTACLE, EVENEMENT, TARIF, TICKET
WHERE EVENEMENT.idevenement = SPECTACLE.idevenement
AND TARIF.typetarif = SPECTACLE.typetarif
AND SPECTACLE.idartiste = ARTISTE.idartiste
AND SPECTACLE.idspectacle = TICKET.idspectacle ;

```

6. Proposer une requête d'insertion d'un événement composé de 7 spectacles qui utilisent les 3 salles. Elle peut être composée de plusieurs requêtes et ne doit pas imposer d'identifiant (== elle doit pouvoir fonctionner sans tenir compte des identifiants des autres événements ou spectacles)

```

INSERT INTO EVENEMENT (idevenement, statutevenement, idspectacle)
VALUES
('3E2_Nuit Blanche', 'programmée', '28AA_aperitif'),
('3E2_Nuit Blanche', 'programmée', '28AB_presentation'),
('3E2_Nuit Blanche', 'programmée', '28AC_ouverture'),

```

```

('3E2_Nuit Blanche', 'programmée', '28AD_concert'),
('3E2_Nuit Blanche', 'programmée', '28AE_diner'),
('3E2_Nuit Blanche', 'programmée', '28AI_after-party'),
('3E2_Nuit Blanche', 'programmée', '28AF_dance')
WHERE EVENEMENT.idevenement = SALLE.idevenement

```

7. Les salaires annuels des salariés pour une année donnée

```

SELECT idsalarie, DISTINCT(salaire)
FROM SALARIE
WHERE DATEDIFF(YEAR('YYYY-MM-DD', 'YYYY-MM-DD'));

```

8. Les ventes d'un événement classées par tarif (tarifs plein, jeune, etc.)

```

SELECT idevenement, typetarif
FROM EVENEMENT, TARIF, SPECTACLE
WHERE EVENEMENT.idevenement = SPECTACLE.idevenement
AND TARIF.typetarif = SPECTACLE.typetarif
ORDER BY typetarif;

```

9. Les pourcentages de types de spectacle pour une année donnée

```

SELECT (COUNT(DISTINCT(typespec)) / COUNT(idspectacle) * 100) AS Pourcentage
FROM SPECTACLE
WHERE DATEDIFF(YEAR('2019-05-21', '2020-05-21'));

```

10. Les événements annulés ainsi que le montant des tickets remboursés

```
SELECT idevenement, SUM(prix)
FROM EVENEMENT, TICKET
WHERE statutevent = 'annule'
AND statutticket = 'rembourse' IN
(SELECT idticket
FROM TICKET
WHERE TICKET.idspectacle = SPECTACLE.idspectacle
AND SPECTACLE.idevenement = EVENEMENT.idevenement);
```

Partie 4 - Contraintes et Triggers

En utilisant des contraintes statiques ou dynamiques, des procédures ou des triggers, proposez des solutions pour vérifier les contraintes suivantes :

1. Un événement peut-être annulé, mais il ne peut pas être supprimé de la base de données, et un ticket vendu peut-être remboursé, mais pas supprimé

```
DELIMITER |
CREATE TRIGGER after_insert_evenement AFTER INSERT
ON Evenement FOR EACH ROW
BEGIN
IF (DELETE from EVENEMENT where statutevenement = 'annulé')
Then abort();
END IF;
End |
DELIMITER ;
```

Pour tester :

```
DELETE FROM EVENEMENT
```

```
WHERE statutevenement = 'annulé' ;
```

2. Vérifier qu'un événement ou un spectacle ne peut pas être créé si un autre événement ou spectacle utilise déjà une des salles sur le même créneau

```
DELIMITER |
```

```
CREATE TRIGGER before_update_spectacle BEFORE UPDATE
```

```
ON Spectacle FOR EACH ROW
```

```
BEGIN
```

```
IF (NEW.horairedatetime = OLD.horairedatetime)
```

```
Then abort();
```

```
END IF;
```

```
End |
```

```
DELIMITER ;
```

Pour tester :

```
INSERT INTO spectacle (horairedatetime, idsalle)
```

```
VALUES
```

```
('2020-12-12 19:00:00', 2),
```

```
('2020-12-12 19:00:00', 2),
```

```
('2020-12-12 19:00:00', 3),
```

```
('2020-12-12 19:00:00', 1) ;
```

3. Un bénévole ne peut-être assigné qu'à un seul responsable par événement

4. On ne peut pas vendre plus de tickets que disponible pour un événement par rapport aux capacités des salles utilisées

```
check(IDSALLE is NULL or IDSALLE in ('S1', 'S2', 'S3')),  
check(IDSALLE<>'S1' or IDSALLE is NULL or NBPLACES <=50),  
check(IDSALLE<>'S2' or IDSALLE is NULL or NBPLACES <=100),  
check(IDSALLE<>'S3' or IDSALLE is NULL or NBPLACES <=300) ;
```

Pour tester :

On peut insérer dans la table TICKET(statutticket = disponible) plus de 50/100/300 lignes en le mettant toutes ou avec une boucle WHILE pour voir que la contrainte statique(prédicat CHECK) arrête l'exécution au moment qu'un nombre de lignes dépasse le nombre de places pour chacune de 3 salles.

5. Les tickets réservés mais non payés doivent redevenir disponibles 15min avant le début d'un événement

```
DELIMITER |  
  
CREATE TRIGGER before_update_ticket BEFORE UPDATE  
  
ON Ticket FOR EACH ROW  
  
BEGIN  
  
IF (OLD.statutticket = 'réservé' AND New.statutticket = 'non réglé'  
  
AND MINUTE(horairedatetime) - CURTIME() <= 15)  
  
Then UPDATE(statutticket = 'disponible');  
  
END IF;  
  
End |  
  
DELIMITER ;  
  
Pour tester :  
  
SELECT CURTIME()  
  
UPDATE ticket  
  
SET statutticket = 'non réglé'
```

Horairedatetime = '20:10:00';

Si l'heure actuelle est différent de l'heure du début de spectacle de moins de 15 minutes, le trigger va se déclencher.