

CORRIGE

Ces éléments de correction n'ont qu'une valeur indicative. Ils ne peuvent en aucun cas engager la responsabilité des autorités académiques, chaque jury est souverain.

E4R : ÉTUDE DE CAS

Durée : 5 heures

Coefficient : 5

CAS CAPDC

ÉLÉMENTS DE CORRECTION

Barème

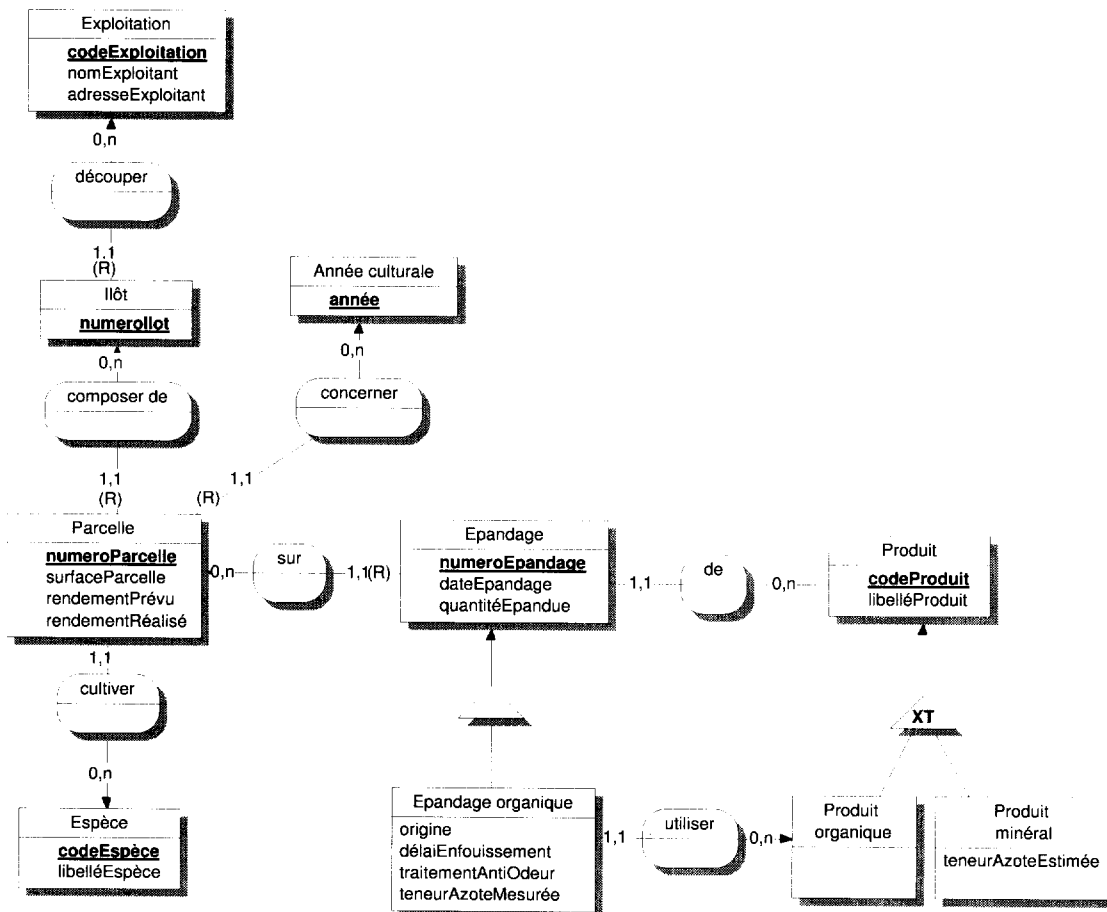
Dossier 1 :		25 points
1.1 Bloc 1	4 points	
1.1 Bloc 2	11 points	
1.1 Bloc 3	10 points	
 Dossier 2 :		 15 points
2.1	3 points	
2.2	2 points	
2.3	4 points	
2.4	6 points	
 Dossier 3 :		 45 points
Partie A		20 points
3.1	4 points	
3.2	4 points	
3.3	4 points	
3.4	3 points	
3.5	5 points	
Partie B		25 points
3.6	3 points	
3.7 a)	3 points	
3.7 b)	5 points	
3.8	7 points	
3.9	7 points	
 Dossier 4 :		 15 points
4.1	5 points	
4.2	3 points	
4.3	2 points	
4.4	5 points	
 Total		 100 points

CODE ÉPREUVE : ISE4D		EXAMEN : BREVET DE TECHNICIEN SUPÉRIEUR	SPÉCIALITÉ : INFORMATIQUE DE GESTION Option Développeur d'applications	
SESSION 2008	SUJET	ÉPREUVE : ÉTUDE DE CAS		
Durée : 5 h	Coefficient : 5	Code sujet : 08DA01N	Page : 1/10	

Dossier 1: Informatisation du cahier d'épandage

- 1.1 Proposer un schéma entité-association représentant les informations nécessaires à l'informatisation de la collecte et du traitement des cahiers d'épandage des exploitations agricoles du département sur plusieurs années.

Exemple de solution :



Ce schéma constitue une solution possible de modélisation : toute autre représentation respectant le sens sera acceptée.

Bloc 2 :

L'historisation des parcelles peut être modélisée par une agrégation entre les entités "Ilôt" et "Année".

Bloc 3 :

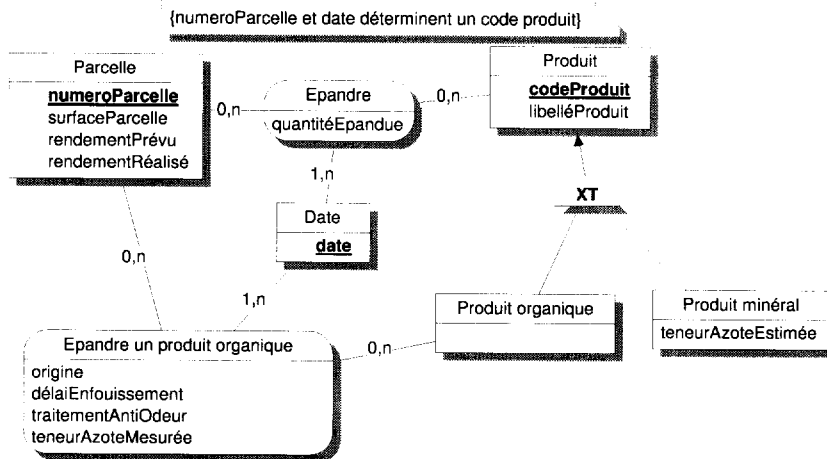
On n'exigera pas la présence de l'association "utiliser", ni de l'entité "Produit organique".

L'association "utiliser" peut être un sous-type de l'association "de".

On acceptera une seule spécialisation sur "Epandage" et pas de sous-typage sur l'entité "Produit", la propriété "teneurAzoteEstimée" devant se trouver dans l'entité "Epandage minéral".

On acceptera une entité "Epandage" identifiée par la date d'épandage en lieu et place du numéro.

Le schéma suivant (extrait) illustre une solution à base d'associations :



On peut trouver une contrainte d'inclusion entre "Epandre un produit organique" et "Epandre" sur les 2 entités pivot Parcelle et Date

Bloc 1 : Exploitation, Ilôt, découper

4 points

Bloc 2 : Année culturale, Parcelle, Espèce, composer, concerner, cultiver

11 points

Pénalité de 3 points dans le cas où l'historisation des parcelles n'est pas gérée.

Bloc 3 : Epanchage, Produit, Epanchage organique, Produit organique, Produit minéral, sur, de, utiliser

10 points

Pénalité de 3 points si les propriétés caractérisant un épandage organique sont attachées au produit organique.

Dossier 2 : Évolution de l'architecture informatique à la chambre d'agriculture

2.1 Donner la configuration IP - adresse IP, masque réseau et passerelle - du serveur « SRV-WEB ».

Adresse IP	179.170.12.151
Masque	255.255.0.0
Passerelle	179.170.12.150

2.2 Justifier le déplacement du serveur « SRV-SQL » dans le réseau local.

Aucun accès externe direct auprès du SGBDR n'est prévu. Les services du SGBDR ne sont sollicités que par le serveur d'application Web, lorsque ce dernier exécute les scripts ou programmes requis par les internautes. **Par raison de sécurité**, l'administrateur décide donc de limiter les services de la DMZ à ceux directement accessibles de l'extérieur.

2.3 Proposer une nouvelle règle sur l'interface 192.168.0.254 qui permet au serveur « SRV-WEB » de communiquer avec le serveur « SRV-SQL », désormais dans le réseau local, avec 192.168.0.153 pour adresse IP. Vous préciserez l'ordre d'application de cette règle par rapport à la règle actuelle.

Remarques :

- Dans l'énoncé du sujet (dans l'encadré expliquant les règles de filtrage), lire "sortant de" à la place "entrant sur"
- Par souci de simplification, on considère uniquement les tables de filtrage en sortie.

Il faut autoriser les paquets en provenance de la machine d'adresse IP 179.170.12.151 vers la machine d'adresse IP 192.168.0.153 et le port 5432 lié au SGBDR. Afin d'être appliquée avant la règle par défaut qui rejette tout paquet, cette nouvelle règle de filtrage sur l'interface 192.168.0.254 (côté réseau local) est à placer avant la règle par défaut :

N° règle	IP source	Port source	IP destinataire	Port destinataire	Décision
1	179.170.12.151	Tous	192.168.0.153	5432	Accepté
Défaut	Toutes	Tous	Toutes	Tous	Rejeté

1 point pour chaque information (IP, ports) sur la ligne attendue

Pénalité de 1 point si la règle est mal placée.

2.4 Calculer et comparer les capacités utiles de stockage du serveur « SRV-SQL » pour chaque solution RAID1 et RAID5. Justifier les calculs.

La solution RAID1 consiste à faire du "miroitage" de disques, c'est-à-dire à dupliquer les données sur un ou plusieurs disques. Dans la configuration de 4 disques de 300 GO, il est possible de dupliquer 1 fois le volume de données de 2 disques, c'est-à-dire d'avoir une capacité utile de 2×300 GO, c'est-à-dire 600 GO, les 2 autres disques étant la réplique des 2 premiers. On peut aussi choisir de dupliquer 3 fois le volume de données d'un disque, la capacité utile étant alors restreinte à celle d'un seul disque, soit 300 GO.

La solution RAID5 consiste à répartir sur les 4 disques les blocs de données, ainsi que les blocs de parité. Il faut alors réserver une capacité équivalente à la capacité d'un des disques de la grappe pour stocker les informations de parité.

La capacité utile est donc de $(4-1) \times 300$ GO, soit 900 GO.

Le RAID5 est donc plus intéressant en termes de capacité utile de stockage.

3 points pour le RAID1, 3 points pour le RAID5

Dossier 3 : Développement de l'application « Registre phytosanitaire »

3.1 Écrire la requête SQL permettant d'obtenir les informations nécessaires pour ce traitement.

```
SELECT DISTINCT Exploitation.nomExploitant, Exploitation.melExploitant
FROM Exploitation, Parcelle, Traitement, Pulverisation
WHERE Exploitation.id = Parcelle.idExploitation
AND Parcelle.id = Traitement.idParcelle
AND Traitement.id = Pulverisation.idTraitement
AND dateDiff (Pulverisation.datePulverisation, Parcelle.dateRecoltePrevue) < 30
```

La jointure peut être exprimée selon la norme SQL 2 – 1992 :

```
SELECT DISTINCT Exploitation.nomExploitant, Exploitation.melExploitant
FROM Exploitation
INNER JOIN Parcelle ON Exploitation.id = Parcelle.idExploitation
INNER JOIN Traitement ON Parcelle.id = Traitement.idParcelle
INNER JOIN Pulverisation ON Traitement.id = Pulverisation.idTraitement
WHERE dateDiff (Pulverisation.datePulverisation, Parcelle.dateRecoltePrevue) < 30
```

NB1 : On attend au minimum comme attribut projeté le mél des exploitants, on n'exige pas le **DISTINCT** (son absence ne sera pas pénalisée).

NB2 : On acceptera une restriction inutile sur le type de traitement valant 'c' (traitement en champ).

3.2 Écrire la requête en SQL qui retourne le nom des exploitants et l'identifiant des parcelles qui ont fait l'objet de plus de quatre pulvérisations.

```
SELECT Exploitation.nomExploitant, Parcelle.id
FROM Exploitation, Parcelle, Traitement, Pulverisation
WHERE Exploitation.id = Parcelle.idExploitation
AND Parcelle.id = Traitement.idParcelle
AND Traitement.id = Pulverisation.idTraitement
GROUP BY Exploitation.nomExploitant, Parcelle.id
HAVING count(*) >4
```

La jointure peut être exprimée selon la norme SQL 2 – 1992 :

```
SELECT Exploitation.nomExploitant, Parcelle.id
FROM Exploitation
INNER JOIN Parcelle ON Exploitation.id = Parcelle.idExploitation
INNER JOIN Traitement ON Parcelle.id = Traitement.idParcelle
INNER JOIN Pulverisation ON Pulverisation.idTraitement = Traitement.id
GROUP BY Exploitation.nomExploitant, Parcelle.id
HAVING count(*) >4
```

NB : On acceptera une restriction inutile sur le type de traitement valant 'c' (traitement en champ).

3.3 Écrire la requête en SQL qui retourne le nom des exploitants ainsi que l'identifiant des parcelles qui n'ont fait l'objet d'aucun traitement.

```
SELECT Exploitation.nomExploitant, Parcelle.id
FROM Exploitation, Parcelle
WHERE Exploitation.id = Parcelle.idExploitation
and Parcelle.id NOT IN ( SELECT distinct Traitement.idParcelle
                        FROM Traitement )
```

La requête peut être exprimée selon la norme SQL 2 – 1992 :

```
SELECT Exploitation.nomExploitant, Parcelle.id
FROM Exploitation
INNER JOIN Parcelle ON Exploitation.id = Parcelle.idExploitation
WHERE Parcelle.id NOT IN ( SELECT Traitement.idParcelle
                        FROM Traitement )
```

```
SELECT Exploitation.nomExploitant, Parcelle.id
FROM Exploitation
INNER JOIN Parcelle ON Exploitation.id = Parcelle.idExploitation
LEFT OUTER JOIN Traitement ON Traitement.idParcelle = Parcelle.id
WHERE Traitement.id IS NULL
```

Remarque : La clause **DISTINCT** dans la requête imbriquée n'est pas nécessaire, mais sa présence ne sera pas pénalisée.

3.4 Indiquer la règle de gestion prise en charge par le déclencheur.

Le déclencheur permet d'interdire l'ajout d'une pulvérisation à une parcelle qui a déjà fait l'objet d'un traitement en semence. En effet, il est indiqué qu'une pulvérisation ne concerne que des traitements en champ.

3.5 Écrire le déclencheur qui permet de respecter la règle de gestion suivante : « pour une parcelle, il ne peut y avoir qu'un seul traitement en semence puisque le produit phytosanitaire est mélangé au semis ».

```
CREATE TRIGGER trg_insert_Traitement
BEFORE INSERT
ON Traitement
FOR EACH ROW

DECLARE nbTrt INTEGER
IF NEW.typeTraitement = 's'
BEGIN
    SELECT COUNT(*) INTO nbTrt FROM traitement
        WHERE traitement.idParcelle = NEW.idParcelle
        AND typeTraitement = 's'

    IF nbTrt > 0
    BEGIN
        RAISERROR
    END
END
END
```

On peut également trouver une solution basée sur une requête SQL EXISTS :

```
if NEW.typeTraitement = 's' AND EXISTS
  (select * from traitement
   where traitement.idParcelle = NEW.idParcelle
   and typeTraitement = 's')
BEGIN
  RAISERROR
END
```

Remarques :

- le premier test peut être remplacé par une condition supplémentaire dans la condition WHERE du SELECT
- *on acceptera l'absence du test NEW.typeTraitement*
- *on acceptera tout moment de déclenchement du trigger, BEFORE ou AFTER*
- *on acceptera que le déclencheur soit lancé sur un UPDATE*
- *on acceptera une condition dosageTraitementSemence is not null à la place de typeTraitement='s'.*

- *on acceptera toute solution cohérente permettant de savoir si un traitement en semence existait déjà.*

- *on valorisera toute solution plus complexe montrant la volonté de contourner les limites d'Oracle en ce qui concerne les tables mutantes (impossibilité de consulter une table en cours de mise à jour), comme le passage par une table intermédiaire.*

3.6	Écrire le constructeur de la classe <i>TraitementSemence</i> .
-----	--

TraitementSemence :: TraitementSemence (unIdTraitementPhytosan : chaîne,
uneParcelleCultivée : Parcelle, unProduitPhytosanitaire : ProduitPhytosanitaire, unDosage : réel)
Début

 Super(unIdTraitementPhytosan, uneParcelleCultivée, unProduitPhytosanitaire)
 dosageTraitementSemence ← unDosage

Fin

NB1 : *on acceptera tout autre formalisme pour l'appel du constructeur de la classe parent*

NB2 : *L'utilisation explicite de l'objet courant (this) est bien sûr possible (remarque valable pour tout le dossier)*

3.7	Écrire les méthodes <i>quantitéAppliquée()</i> des classes : a) <i>TraitementSemence</i> b) <i>TraitementEnChamp</i> .
-----	--

a) QuantitéAppliquée de la classe TraitementSemence

Fonction TraitementSemence :: QuantitéAppliquée () : Réel

Début

Retourne dosageTraitementSemence * getLaParcelleCultivée().getSurface()

Fin

b) QuantitéAppliquée de la classe TraitementEnChamp

Fonction TraitementEnChamp :: QuantitéAppliquée () : Réel

Var

cumul, qte : Réel

unePulvérisation : Pulvérisation

Début

cumul ← 0

Pour chaque unePulvérisation dans lesPulvérisations

cumul ← cumul + unePulvérisation.getDosage()

FinPour

qte ← cumul * getLaParcelleCultivée().getSurface()

Retourne qte

Fin

Remarque : on acceptera que la quantité soit calculée à chaque fois dans la boucle

Pénalité de 1 point si la multiplication par la surface est absente

On peut également écrire le parcours avec incrémentation de l'index :

Var

cumul, qte : Réel

index, nb : Entier

Début

cumul ← 0

nb ← lesPulvérisations.cardinal()

Pour index de 1 à nb

cumul ← cumul + lesPulvérisations.obtenirObjet(index).getDosage()

FinPour

qte ← cumul * getLaParcelleCultivée().getSurface()

Retourne qte

Fin

3.8	Écrire la méthode <i>chargerLesPulvérisations()</i> de la classe <i>Passerelle</i> .
-----	--

Publique Statique Fonction Passerelle :: chargerLesPulvérisations(unIdTraitement : chaîne) :
Collection de Pulvérisation

Var :

```
texteReq : Chaîne  
jeuPulvérisations : JeuEnregistrements  
unePulvérisation : Pulvérisation  
lesPulvérisationsARetourner : Collection de Pulvérisation
```

Début

```
texteReq ← "select * from Pulvérisation  
          where pulvérisation.idTraitement = ' " + unIdTraitement + " ' "  
jeuPulvérisations ← new jeuEnregistrements(texteReq)  
lesPulvérisationsARetourner = new Collection() de Pulvérisation  
tant que NON jeuPulvérisations.fin()  
    unePulvérisation ← new Pulvérisation(jeuPulvérisations.getValeur(" id "),  
                                          jeuPulvérisations.getValeur(" datePulvérisation "),  
                                          jeuPulvérisations.getValeur(" dosage " ) )  
    lesPulvérisationsARetourner.ajouter(unePulvérisation)  
    jeuPulvérisations.suivant()  
finTQ  
jeuPulvérisations.fermer ()  
Retourne lesPulvérisationsARetourner
```

Fin

3.9	Écrire les instructions manquantes de la méthode <i>chargerLeTraitement()</i> de la classe <i>Passerelle</i> . Le complément demandé est à reporter sur la copie.
-----	---

```
si jeuTraitements.getValeur("typeTraitement") = 's' alors  
    traitementARetourner ← new TraitementSemence( unIdTraitement,  
                                                  laParcelle,  
                                                  leProduit,  
                                                  jeuTraitements.getValeur("dosageTraitementSemence"))  
sinon  
    traitementARetourner ← new TraitementEnChamp( unIdTraitement,  
                                                  laParcelle,  
                                                  leProduit,  
                                                  chargerLesPulvérisations(unIdTraitement))  
fin
```

NB : on acceptera le test *jeuTraitements.getValeur("dosageTraitementSemence") <> null*

Dossier 4	Evaluation des charges du projet AIM
------------------	---

4.1	Indiquer le nombre d'exploitations visées par le taux de couverture. Justifier la charge de la phase de déploiement notifiée ci-dessus.
-----	---

75 % de 15 000 soit $7\,500 + 7\,500/2 = 11\,250$ soit 11 250 exploitations.

A raison d'une charge de 1/2H par exploitation on explique la charge de déploiement prévue soit $11\,250/2 = 5\,625$ JH

4.2	Indiquer le nombre d'exploitations ayant bénéficié de la phase de déploiement fin 2007.
-----	---

4.3	Dresser le bilan de cette phase. <i>Commenter.</i>
-----	--

4.2

625 x 2 soit 1 250 exploitations

4.3

La phase a pris du retard : 1 400-1 250 soit 150 exploitations, n'ont pu être équipées.

4.4	Proposer une planification mensuelle de la phase de déploiement et les ressources humaines minimales à mettre en œuvre chez le prestataire pour atteindre l'objectif de 75 % en 2009.
-----	---

De mai 2008 à décembre 2009 : 20 mois ouvrables

Il reste $11\,250 - 1\,250$ soit 10 000 exploitations à installer.

La charge mensuelle, sous réserve de linéarité, sera de 500 exploitations par mois soit 250 JH mobilisables par mois.

Il y a 20 jours ouvrés par mois ; il faudra donc mobiliser $250/20$ soit 12,5 H (13 H) pour que le projet respecte l'objectif fixé pour 2009.

Remarque : On acceptera que le mois d'avril 2008 soit inclus dans la période de déploiement.

21 mois ouvrables et 10 000 exploitations à installer : environ 477 exploitations par mois soit environ 239 JH mobilisables par mois.

$239/20 = 11,95$ H (12 H)