



1<sup>ER</sup> CENTRE DE FORMATION COMPTABLE  
VIA INTERNET

100%  
GRATUIT



# CORRIGÉS COMPTALIA DCG 2010



Ce corrigé est la propriété exclusive de Comptalia ; toute utilisation autre que personnelle  
devra faire l'objet d'une demande préalable sous peine de poursuites.

**SESSION 2010****UE 8 - SYSTÈMES D'INFORMATION DE GESTION****Durée de l'épreuve : 4 heures - coefficient : 1,5**

---

Aucun document ni aucun matériel autorisé En conséquence, tout usage d'une calculatrice est **INTERDIT** et constituerait une **fraude**.

Document remis au candidat: **le sujet comporte 12 pages numérotées de 1/12 il 15/12.**

**Il vous est demandé de vérifier que le sujet est complet dès sa mise à votre disposition**

---

*Le sujet se présente sous la forme de 5 dossiers indépendants :*

Page de garde .....	page 1
Présentation du sujet .....	page 2
DOSSIER 1 - Gestion des assemblées générales..... (5 points) .....	page 2
DOSSIER 2 - Evolution du système d'information .....	(3 points) ..... page 3
DOSSIER 3 - Evolution du réseau .....	(4 points) ..... page 4
DOSSIER 4 - Gestion des appels de fonds travaux.....	(4,5 points) ..... page 4
DOSSIER 5 - Gestion des lots et tantièmes sur tableur.....	(3,5 points) ..... page 5

---

*Le sujet comporte les annexes suivantes :*

**DOSSIER 1**

Annexe 1 - Procuration à l'assemblée générale .....	page 6
Annexe 2 –Ordre du jour .....	page 6

**DOSSIER 2**

Annexe 3 – Le mode SaaS: bien plus qu'une réduction des coûts .....	page 7
---	--------

**DOSSIER 3**

Annexe 4 - Description du réseau .....	page 8
--	--------

**DOSSIER 4**

Annexe 5 - Modèle relationnel de la gestion des appels de fonds-travaux.....	page 9
--	--------

**DOSSIER 5**

Annexe 6 - Tableaux de gestion des lots et tantièmes d'une copropriété.....	page 9
---	--------

Annexe 7 - Indications de réalisation de la fonction personnalisée .....	page 11
--	---------

<b>Annexe A - Modèle conceptuel des données (à rendre avec la copie) .....</b>	<b>page 12</b>
--	----------------

**NOTA : l'annexe A doit obligatoirement être rendue avec la copie.**

**AVERTISSEMENT**

**Si le texte du sujet, de ses questions ou de ses annexes, vous conduit à formuler une ou plusieurs hypothèses, il vous est demandé de la (ou les) mentionner explicitement dans votre copie.**

## SUJET

La société IMMO-DO (Demandes et Offres) est spécialisée dans la gestion des biens immobiliers. Son activité principale consiste à assurer les fonctions de syndic de copropriétés.

Dans ce cadre, elle est amenée à :

- faire réaliser par les entreprises l'entretien des parties communes, effectuer le suivi des réparations et des travaux;
- gérer le personnel éventuellement employé (concierge, agent de sécurité ...);
- payer les factures relatives aux dépenses engagées (travaux, entretien, électricité, combustibles, eau, ...);
- répartir les charges entre tous les copropriétaires;
- procéder aux appels de fonds;
- gérer les assemblées générales des copropriétaires.

Son siège social est situé à Avignon, mais dans le cadre de son expansion, il est prévu la mise en place d'agences dans plusieurs autres villes de France.

Outil de gestion et vecteur de son développement, le système d'information nécessite une étude complète.

### DOSSIER 1 - MODÉLISATION DES DONNÉES

#### **Description.**

Pour son fonctionnement normal, chaque copropriété est réunie régulièrement en assemblée générale (AG). Tous les ans, un certain nombre d'AG ont lieu. Chacune de ces assemblées générales est numérotée, et cette numérotation est propre à chaque année. A chaque assemblée, tous les propriétaires sont convoqués. Il est nécessaire de mémoriser pour chaque propriétaire, la date d'envoi de chaque convocation effectuée par lettre recommandée. Certains propriétaires étant absents, il faut donc connaître, pour chaque AG la liste des personnes présentes. Ceci est possible grâce à la feuille d'émargement remplie au début de chaque séance, et doit être mémorisée dans le système d'information. Dans une AG, les propriétaires peuvent se faire représenter par un propriétaire de la même copropriété. Dans ce cas, une procuration est nécessaire (voir annexe 1). Une procuration, document numéroté, est utilisée pour une seule représentation à une seule AG.

#### **Les lots.**

Un propriétaire possède, au sein de la copropriété, différents lots. Chaque lot est numéroté et caractérisé par sa description et son tantième (nombre de points attribués à ce lot en fonction de son importance). Le tantième sera pris en compte lors des votes au sein des AG afin de respecter la responsabilité de chaque propriétaire. Une information supplémentaire caractérisant le lot est le type de lot (garage, appartement, place de parking, ...)

#### **Déroulement d'une AG.**

Le déroulement d'une assemblée générale est une suite de votes correspondant à autant de propositions.

Ainsi une AG est composée de propositions numérotées par AG (voir annexe 2).

Une proposition concerne un certain nombre de types de lots. Il faut donc connaître ces derniers. Par exemple, certaines propositions se rapportent aux garages, d'autres aux appartements, d'autres encore à tous les types de lots ...

#### **Les votes.**

D'autre part, chaque proposition fait l'objet d'un vote. Le type de vote associé à chaque proposition peut être différent (par exemple, majorité absolue, majorité simple, unanimité, ...).

Lorsqu'un vote est valide, la proposition fait l'objet d'une et une seule décision. Quand un vote n'est pas valide la proposition n'est reliée à aucune décision. On numérote chaque décision, et une séquence de numéro sera propre à chaque AG. En dehors de ce numéro, une décision est caractérisée par son texte, par exemple: accord, refus, descriptif. ..

NB : dans ce dossier le champ d'étude est limité à une seule copropriété.

### Travail à faire

**1. Dans quel contexte une identification relative (liée à la notion d'entité forte/faible) peut-elle être utile dans un Modèle de Données (MCD)? Vous illustrerez votre réponse par un exemple lié avec le cas présenté.**

**2. Sur l'annexe A (à compléter et il rendre avec la copie), compléter le modèle conceptuel des données permettant de gérer les assemblées générales de la copropriété.**

## DOSSIER 2 - ÉVOLUTION DU SYSTÈME D'INFORMATION

Annexe à utiliser : annexe 3

Pour l'évolution de son système d'information, la société IMMO-DO étudie diverses solutions.

Les fonctions déjà présentes au sein de son système devront exister dans la version future tout en tenant compte du cahier des charges élaboré par l'équipe de projet.

D'autres fonctions, en particulier celles générées par la création des agences réparties sur le territoire français, devront être intégrées.

Actuellement, le système en place est structuré de la façon suivante:

- certaines fonctions (comptabilité, trésorerie) sont assurées par un PGI installé sur des serveurs dans les locaux du siège.

- d'autres fonctions (gestion des locations, gestion des copropriétés) sont également installées sur des serveurs appartenant à la société, mais ont été développées par une SSII qui en assure la maintenance, grâce à un contrat annualisé.

Ces diverses fonctions ne communiquent pas entre elles.

Pour l'évolution envisagée, IMMO-DO doit choisir entre plusieurs solutions:

1. Remplacer l'existant et assurer les nouvelles fonctions grâce à la mise en place d'un seul PGI couvrant tous les besoins fonctionnels.

2. Remplacer son système et assurer les nouvelles fonctions par une ou plusieurs solutions SaaS.

### Travail à faire

**1. Au sein d'un tableau structuré, dresser pour ces deux propositions leurs avantages et leurs inconvénients en tenant compte de la spécificité de IMMO-DO.**

**2. Illustrer par des exemples la notion "d'amélioration continue des fonctionnalités".**

## DOSSIER 3 - EVOLUTION DU RESEAU

Le réseau informatique de la société IMMO-DO est décrit dans l'annexe 4.

Ce schéma présente :

- l'état actuel du réseau du siège avignonnais incluant les différents serveurs, les services administratifs, la DMZ (zone démilitarisée);
- le projet d'une agence reliée au site du siège.

### Travail à faire

1. Décrire le rôle d'une DMZ (zone démilitarisée).
2. Décrire le rôle de chacun des serveurs présents dans l'annexe 4.
3. Quelle est l'adresse du réseau du siège d'Avignon?

La société prévoit d'utiliser le principe des sous-réseaux pour gérer les adresses des agences.

4. Donner les avantages d'un tel choix.
5. Avec les informations présentes dans l'annexe 4 et les critères énoncés ci-dessus, déterminer le nombre d'agences qui peuvent être créées en justifiant votre calcul.

## DOSSIER 4 – GESTION DES APPELS DE FONDS - TRAVAUX

Chaque copropriété doit réaliser des travaux (réfections, aménagements divers, etc.). Le modèle relationnel présenté en annexe 5 décrit la base de données actuellement utilisée. On vous demande d'obtenir certaines informations grâce à des requêtes ciblées.

### Travail à faire

Rédiger, en langage SQL, les requêtes suivantes:

1. Liste des entreprises (noms, sans doublon) ayant effectué des réalisations dans la copropriété "Bel Air" en 2009
2. Liste des propriétaires (nom, prénom, téléphone), de la copropriété "Pelican" classés par ordre alphabétique (nom et prénom), dont le numéro de téléphone a été renseigné.
3. Liste des paiements reçus (montant, date, nom de la réalisation) pour la réalisation nommée "Réfection chaufferie" et pour la réalisation nommée "Mise en place TNT".
4. Montant total des paiements reçus pour chaque réalisation (numéro, description, montant de la réalisation, total des paiements).
5. Liste des appels (Numéro, nom et prénom du propriétaire) pour lesquels un paiement n'a pas encore été reçu.
6. Nom de l'entreprise ayant effectué la réalisation la plus onéreuse concernant la copropriété "Le Bel Air".

**DOSSIER 5 – GESTION DES LOTS ET TANTIEMES SUR TABLEUR**

Dans l'attente d'une refonte complète du système d'information, une solution transitoire doit être mise en place. Dans cet objectif, un tableur a été utilisé afin de permettre la gestion des lots et tantièmes.

**Travail à faire**

**1. Travail sur tableur (annexes 6.A et 6.B)**

**Ecrire les formules permettant d'obtenir les informations suivantes:**

Tableau concerné	Cellule	Information à obtenir	Remarque
<b>6.B</b>	<b>K2</b>	Nombre de lots de nature "parking"(nature en cellule J2)	La formule doit pouvoir être utilisée pour les cellules K3->K8
	<b>L2</b>	Total des tantièmes pour les lots de nature "parking"	La formule doit pouvoir être utilisée pour les cellules L3->L8
	<b>K9</b>	Total du nombre de lots	

**2. La fonction nommée TantiemesTypeProp (nomPropriétaire : chaîne, typeLot : chaîne: decimal), utilisée eu cellule R3 (annexes 6.A, 6.C et 7) permet de calculer le nombre total de tantièmes pour un propriétaire et un type de garage.**

**Rédiger l'algorithme correspondant, en utilisant la fonction *ValeurCellule()* en précisant les noms et types de paramètres et de la valeur de retour.**

**Annexe 1 - Procuration il l'assemblée générale**

IMMO-DO					
<i>Gestion de biens immobiliers</i>					
Assemblée Générale de la copropriété: <i>Le Pré Sergent</i>					
Procuration ....n°	<input type="text" value="14465"/>	En date du:	<input type="text" value="21 avril 2009"/>		
AG .....n°	<input type="text" value="3"/>	de l'année	<input type="text" value="2009"/>	Date:	<input type="text" value="21 avril 2009"/>
Je soussigné,					
<input type="text" value="Marcel BERNARD"/>		Propriétaire n°	<input type="text" value="104"/>		
déclare donner procuration à					
<input type="text" value="Gérard HENRI"/>		Propriétaire n°	<input type="text" value="391"/>		
Pour tous les votes tenus et décisions prises lors de cette réunion,					
Le propriétaire représenté					
Signature					

**Annexe 2 –Ordre du jour**

Assemblée Générale du 2 avril 2009.

L'assemblée délibérera sur l'ordre du jour suivant:

N° 1 : Désignation du président de séance.

N° 2 : Désignation des scrutateurs

N°:3 : Approbation des comptes de l'exercice du 01/01/2008 au 31/12/2008

N° 4: Quitus au Syndic pour sa gestion de l'exercice arrêté au 31/12/2008/

N° 5: Approbation du budget prévisionnel de l'exercice du 01/01/2010 au 31/12/2010 pour un montant de 117 689,00 € TIC.

N° 6 : Désignation des membres du conseil syndical pour une durée d'une année.

N° 7 : Désignation à nouveau de la société IMMO-DO en qualité de syndic et approbation du contrat.

N° 8 : Décision à prendre concernant la réalisation d'une rehausse du mur de la plate forme containers

N° 9 : Décision à prendre concernant la mise en place de la TNT dans la copropriété.

N° 10 : Décision à prendre concernant la rénovation et passage au gaz naturel de la chaufferie avec proposition d'exploitation globale.

### Annexe 3

#### **Le mode SaaS : bien plus qu'une réduction des coûts**

Argument principal en faveur du mode SaaS, la réduction des coûts est partout présentée comme le principal avantage de ces solutions. Pour être exact, une précision s'impose.

Pour reprendre la définition de **Guillaume Plouin** (Cloud Computing et SaaS, Editions Dunod), un Software as a Service « est un logiciel fourni sous la forme de service et non sous la forme de programme informatique. Les utilisateurs des applications SaaS accèdent à ce service via Internet. » Principale conséquence de ce modèle de distribution, le fameux « pay per use » : le paiement s'effectue suivant le nombre d'utilisateurs, sans coût fixe forfaitaire, et mensuellement. Pour être exact, le modèle SaaS permet donc un lissage des coûts informatiques, qui deviennent prédictibles.

Le passage d'un coût fixe à un coût variable unique est donc le point clé à retenir en termes d'avantages financiers. Dans le modèle traditionnel, il fallait prendre en compte un ensemble d'éléments pour évaluer les coûts, comme l'achat des licences, de support, de mises à jour et de frais d'exploitation. Aujourd'hui ces éléments sont agrégés dans un prix unique forfaitaire.

Aujourd'hui, nombre d'organisations voient dans le mode SaaS un moyen de réduire les coûts. Bien que ce soit effectivement un bénéfice majeur, il en est bien d'autres à considérer, notamment celui-ci : l'amélioration continue des fonctionnalités.

Dans un mode pur SaaS, l'application est fournie comme un service, par le biais d'un portail web. Aucune acquisition de licence, et de fait les mises à jour sont incluses dans le prix (à l'usage) : l'utilisateur n'a plus à changer de version tous les deux ou trois ans, et profite des derniers updates. Cette différence amène les éditeurs à revoir constamment leur logiciel, et donc à l'améliorer, en étant plus à l'écoute du marché et des utilisateurs. Le feedback de ces derniers quant à l'usage quotidien de l'application permet l'amélioration continue des fonctionnalités.

Dans le système on-premise (déploiement des applications sur site), chaque client travaille sur sa petite île, et le feedback est hors d'atteinte. Avec le SaaS, la facilité de la rapidité de déploiement entraînent de facto une plus grande proximité avec le client, qui devient acteur et pilote de l'amélioration. Chacun trouve ainsi son compte dans ce modèle qu'est le « Software as a service » : client, écouté et influent, et l'éditeur, qui améliore leur satisfaction et peut transformer ce processus en un puissant message de communication. La notion de « service » parle d'elle-même, et modifie la relation avec le client d'un bout à l'autre de la chaîne: on doit désormais servir le client. Si vous n'êtes pas satisfait de votre opérateur téléphonique, vous en changez. Si vous jugez qu'il répond à vos attentes et que la qualité de service s'améliore, vous le conservez.

Le SaaS permet effectivement de réaliser des économies, mais ce n'est qu'une partie des avantages. Disposer d'un outil continuellement amélioré, c'est encore mieux. Le SaaS, c'est aussi simple que ça !

May 28th, and Jun 12th, 2009 by Cyril Laguiseray  
[www.mysaas.fr](http://www.mysaas.fr)



**Annexe 4 – Réseau : situation actuelle avec projet agence****1 – Siège de la société IMMO-DO****Serveurs internes :**

- Serveur Proxy Linux Debian  
adresse 172.30.32.11 / 255.255.255.0
- Serveur de fichiers Windows serveur 2003  
adresse 172.30.32.12 / 255.255.255.0
- Serveur WEB intranet Linux Debian & Apache  
adresse 172.30.32.13 / 255.255.255.0
- Serveur DHCP Linux Debian  
adresse 172.30.32.14 / 255.255.255.0
- commutateur 3com
- Routeur accès Agence (projet) : adresse inetrne 172.30.32.254 / 255.255.255.0

**Poste de travail**

- Poste de travail 1 : secrétaire : Windows seven pro
- Poste de travail 2 : directeur : Windows seven pro
- Poste de travail 3 : comptable : Windows seven pro
- commutateur 3com

**DMZ (zone démilitarisée)**

- Pare-feu interne  
adresse 172.30.32.2 / 255.255.255.0
- Serveur DNS Windows serveur 2003  
adresse 172.30.32.3 / 255.255.255.0
- Serveur WEB internet Linux Debian & Apache  
adresse 172.30.32.4 / 255.255.255.0
- Serveur messagerie Exchange Windows serveur 2003  
adresse 172.30.32.5 / 255.255.255.0
- Pare-feu externe
- commutateur 3com
- Routeur accès internet

**2 – Agence (Projet)**

- Serveur DHCP agent relais Linux Debian  
adresse 172.30.64.3 / 255.255.255.0
- commutateur 3com
- Routeur accès siège : adresse interne 172.30.64.2.254 / 255.255.255.0
- Poste de travail 1 : secrétaire : Windows seven pro
- Poste de travail 2 : responsable agence : Windows seven pro

**Annexe 5 - Modèle relationnel de la gestion des appels de fonds – travaux**

 COPROPRIETE(CopNum,CopNom,CopRue, CopCP,CopVille)

 ENTREPRISE(EntNum,EntNom,EntTel)

 REALISATION(RealNum, RealNom,RealDescription, RealDate, RealMontant, EntNum#, CopNum#)

 APPEL(AppelNum,RealNum#,PropNum#,date,montant)

 PAIEMENT(PmtNum, AppelNum#,PmtDate,PmtMontant)

 PROPRIETAIRE(PropNum,PropNom,PropPrenom,PropRue,CPProp,VilleProp,TelProp,EmailProp,  
 copNum#)

**Règles de gestion**

Les réalisations font l'objet d'appels de fonds spécifiques.

Chaque réalisation est réalisée par une entreprise. Chaque appel de fond concerne un propriétaire et une réalisation particulière. Il peut y avoir plusieurs appels pour une même réalisation.

Chaque paiement est relatif à un appel de fond. Une occurrence de paiement est créée lors de la réception de celui-ci.

**Annexe 6 - Tableaux de gestion des lots et tantièmes d'une copropriété**

Nota

-les cellules contenant des formules comportent une trame grisée

-les tableaux sont tous situés sur une même feuille.

**Tableau 6A : lots et tantièmes de la copropriété**

	A	B	C	D	E	F	G	H
	N° Lot	Bâtiment	Type Lot	Etage	N°	Tantièmes	Tantièmes asc	Propriétaire
1								
2	1		Parking			1	194	POUILLE
3	2		Parking			2	194	JOLIVET
4	3		Parking			3	194	DUBARD
5	4		Parking			4	194	RENVOISE
6	5		Parking			5	194	JOLIVET
...								
48	46	1	Garage			20	1939	RICHARD
49	47	1	Garage			21	1939	VERDIER
50	48	1	Garage			22	1939	SIMARD
51	49	1	Garage			23	1939	PETIT
52	50	1	Garage			24	1939	VILLAINES
53	51	1	Garage			25	1939	BOURBON
...								
65	87	1		5	1	4	24 740	56 JUBAULT
66	88	1		1	1	1	9 612	14 DURAND
67	89	1		1	1	22	9 612	14 HELIERE
...								
74	100	1		1	2	14	9 612	28 COULBOIS
75	101	1		2	2	11	13 665	49 JAMEAU
76	102	1		1	2	8	9 612	28 ROBLLOT
77	103	1		5	3	6	24 740	169 GORON
78	104	1		1	3	3	9 612	42 GIROUX
79	105	1		1	3	24	9 612	42 LORIN
...								
154	152	2		2	7	41	13 666	35 LURIER
155							1 000 000	2 000

*Bâtiment* : la copropriété comporte 2 bâtiments

*N°* : numéro porté sur le lot (numéro de parking, numéro de garage, numéro d'appartement).

*Type lot* : nature du bien (parking, garage, Type: 1,2,3,4,5 pour les appartements).

*Tantièmes asc* : (non utilisé ici).

**Tableau 6B : récapitulatif des tantièmes par nature de lot**

J	K	L
1	Type Lot	Nombre de lots
2		Tantièmes
2	Parking	45
3	garage	41
4	1	26
5	2	22
6	3	7
7	4	8
8	5	3
9	Total	152
		1 000 000

**Tableau 6C : récapitulatif des tantièmes par propriétaire et par nature**

Q	R
1	Nom
2	BURLOY
2	Total tantièmes
3	parking
4	garage
5	1
6	2
7	3
8	4
9	5

**Formule :**

Q	R
1	Nom
2	Total tantièmes
3	parking
4	garage
	JOLIVET
	16 089
	= TantièmesTypesProp(R1;Q3)

Remarque : ici pour JOLIVET et parking, on trouve 388 en R3

**Annexe 7 - Indications de réalisation de la fonction personnalisée**Traitement opéré par la fonction *TantiemesTypeProp()*

La fonction doit:

- rechercher tous les lots correspondant au nom du propriétaire indiqué et au type de lot;
- cumuler le total des tantièmes pour les lots trouvés;
- retourner ce cumul.

Documentation de la fonction *ValeurCellule()* :Le tableur utilisé dispose d'une fonction *ValeurCellule()* décrite ci-après.Syntaxe*ValeurCellule*(ligne, *colonne*)*Ligne* : numéro de la ligne (de type entier)*Colonne*: numéro de la ligne (de type entier)

Exemple d'utilisation de la fonction *ValeurCellule* : ici on recherche la valeur de la cellule de coordonnées ligne = 2 colonne = 6

variables

a: entier

i: entier

j:entier

début

i&lt;- 2

j&lt;- 6 ...

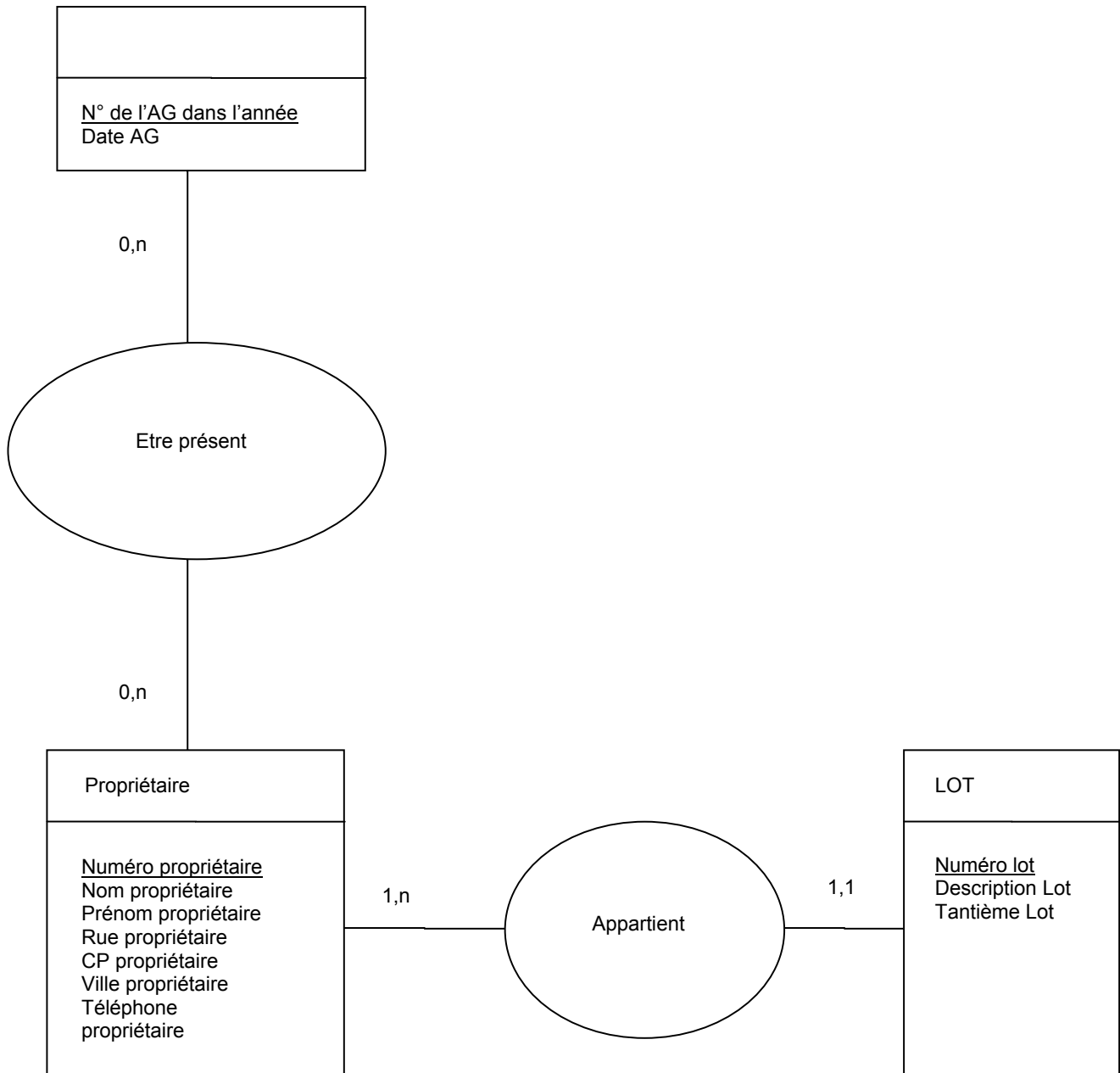
a←*ValeurCellule*(i,j)

// soit ici la valeur 194 pour a

...

fin

**Annexe A – Ebauche du MCD (à compléter et à rendre avec votre copie)**



INSCRIPTION  
TOUTE L'ANNÉE  
DÉMARRAGE  
SOUS 48 H



## COMPTALIA FORMATION

Préparations aux DCG-DSCG  
Formations en Comptabilité, Gestion de la paye,  
Spécialisation IFRS... en ligne.

- › Formation complète - sur mesure - à votre rythme
- › Cours en ligne + supports papier + cours-vidéo
- › Assistance permanente de vos formateurs
- › Devoirs corrigés - séances de cours en direct sur Internet

ET TELLEMENT PLUS SUR

**www.comptalia.com**



CONSULTEZ  
NOTRE DOCUMENTATION  
SUR NOTRE SITE



## COMPTALIA TV

La chaîne du savoir comptable.

- › Cours-vidéo à la demande pour tout le programme DCG et DSCG
- › Cours-vidéo sur l'utilisation des logiciels comptables
- › Magazines d'information professionnelle : l'actualité Fiscale, Sociale, Juridique, Comptable, IFRS...

ALLEZ-VOIR SUR

**www.comptalia.tv**

## UNE QUESTION ?

- Comment finaliser mon DCG ou mon DSCG ?
- Quelle formation professionnelle pour mon projet ?
- Comment fonctionnent les formations Comptalia via Internet ?
- Mes frais de formation peuvent-ils être pris en charge ?...

**NOS CONSEILLERS VOUS RENSEIGNENT AU**

**N° Vert 0 800 COMPTA**

APPEL GRATUIT DEPUIS UN POSTE FIXE  
SOIT **0 800 266 782**



## Proposition de corrigé

Remarque préalable.

Le corrigé proposé par Comptalia est plus détaillé que ce que l'on est en droit d'attendre d'un candidat dans le temps imparti pour chaque épreuve. A titre pédagogique le corrigé comporte donc des rappels de cours, non exigés dans le traitement du sujet.

### DOSSIER 1 - MODÉLISATION DES DONNÉES

- 1. Dans quel contexte une identification relative (liée à la notion d'entité forte/faible) peut-elle être utile dans un Modèle de Données (MCD)? Vous illustrerez votre réponse par un exemple lié avec le cas présenté.**

Une entité forte est une entité qui peut être considérée de façon isolée alors qu'une entité faible ne peut être considérée qu'en association avec une entité forte. En d'autres termes, une occurrence d'une entité faible ne pourra être identifiée que par rapport à une occurrence d'une entité forte.

Cela peut être utile pour modéliser le fait que les occurrences des entités faibles sont regroupées au sein de l'occurrence de l'entité forte et qu'elles ne peuvent pas jouer de rôle seules.

Par exemple, dans le présent cas de figure, le lot devrait être considéré en association avec son propriétaire car tout passe par ce dernier (la convocation aux assemblées générales, la procuration, le vote ...). De même, les propositions ou les décisions sont modélisées par des entités PROPOSITION et DECISION qui sont faibles par rapport à l'entité ASSEMBLEE GENERALE.

- 2. Sur l'annexe A (à compléter et il rendre avec la copie), compléter le modèle conceptuel des données permettant de gérer les assemblées générales de la copropriété.**

Pour compléter ce MCD, nous pouvons construire au préalable au brouillon un dictionnaire de données et une couverture minimale. Pour déterminer ce dictionnaire de donnée, nous reprenons d'abord la liste des éléments déjà utilisés dans le MCD partiel.

Mnémonique	Signification
N° de l'AG dans l'année	Numéro de l'assemblée générale dans l'année
Date AG	Date de l'assemblée générale
Numéro propriétaire	Numéro du propriétaire
Nom propriétaire	Nom (patronyme) du propriétaire
Prénom propriétaire	Prénom du propriétaire
Rue propriétaire	Numéro et rue du propriétaire
CP propriétaire	Code postal du propriétaire
Ville propriétaire	Ville du propriétaire
Téléphone propriétaire	Téléphone du propriétaire
Numéro lot	Numéro de lot
Description Lot	Description du lot
Tantième Lot	Tantième associé au lot

On complète ensuite cette liste en analysant les document fournis en annexe.

Annexe 1 : La procuration il l'assemblée générale

IMMO-DO

*immobiliè*  
*Gestion de biens rs*

Assemblée Générale de la copropriété: *Le*  
*Pré Sergent*

Procuration .....n°  En date du:

AG .....n°  de l'année  Date:

Je soussigné,

Propriétaire n°

déclare donner procuration à

Propriétaire n°

Pour tous les votes tenus et décisions prises lors de cette réunion,

Le propriétaire représenté

Signature

Les éléments mis en évidence en rouge sont des occurrence des éléments d'information qu'il est pertinent de placer dans le système d'information. On complète donc le dictionnaire de données en vérifiant si ces éléments d'information n'ont pas déjà été considérés (pour éviter les doublons).

Mnémonique	Signification
numProcuration	Numéro de la procuration
dateProcuration	Date de la procuration
N° de l'AG dans l'année	Déjà considéré
Année de l'AG	Déjà considéré (dans la date de l'AG)
Date AG	Déjà considéré
Numéro propriétaire représenté	Déjà considéré
Nom propriétaire représenté	Déjà considéré
Prénom propriétaire représenté	Déjà considéré
Numéro propriétaire représentant	Déjà considéré
Nom propriétaire représentant	Déjà considéré



Prénom propriétaire représentant	Déjà considéré
----------------------------------	----------------

Annexe 2 : Ordre du jour

Assemblée Générale du **2 avril 2009**.

L'assemblée délibérera sur l'ordre du jour suivant:

**N° 1 : Désignation du président de séance.**

N° 2 : Désignation des scrutateurs ...etc

On procède de la même manière pour l'ordre du jour. On met en évidence deux occurrences d'éléments d'information dont un a déjà été considéré.

Mnémonique	Signification
Date AG	Déjà considéré
numProposition	Numéro de la proposition
texteProposition	Texte de la proposition

On termine ce travail en analysant le texte de l'énoncé :

**Description.**

Pour son fonctionnement normal, chaque copropriété est réunie régulièrement en assemblée générale (AG). Tous les ans, un certain nombre d'AG ont lieu. Chacune de ces assemblées générales est **numérotée**, et cette numérotation est propre à chaque année. A chaque assemblée, tous les propriétaires sont convoqués. Il est nécessaire de mémoriser pour chaque propriétaire, **la date d'envoi de chaque convocation** effectuée par lettre recommandée. Certains propriétaires étant absents, il faut donc connaître, pour chaque AG **la liste des personnes présentes**. Ceci est possible grâce à la feuille d'émargement remplie au début de chaque séance, et doit être mémorisée dans le système d'information. Dans une AG, les propriétaires peuvent se faire représenter par un propriétaire de la même copropriété. Dans ce cas, une procuration est nécessaire (voir annexe 1). Une procuration, **document numéroté**, est utilisée pour une seule représentation à une seule AG.

**Les lots.**

Un propriétaire possède, au sein de la copropriété, différents lots. Chaque lot est **numéroté** et caractérisé par sa **description** et son **tantième** (nombre de points attribués à ce lot en fonction de son importance). Le tantième sera pris en compte lors des votes au sein des AG afin de respecter la responsabilité de chaque propriétaire. Une information supplémentaire caractérisant le lot est le **type de lot** (garage, appartement, place de parking, ...)

**Déroulement d'une AG.**

Le déroulement d'une assemblée générale est une suite de votes correspondant à autant de propositions.

Ainsi une AG est composée de propositions **numérotées** par AG (voir annexe 2).

Une proposition concerne un certain nombre de types de lots. Il faut donc connaître ces derniers. Par exemple, certaines propositions se rapportent aux garages, d'autres aux appartements, d'autres encore à tous les types de lots ...

**Les votes.**

D'autre part, chaque proposition fait l'objet d'un vote. Le **type de vote** associé à chaque proposition peut être différent (par exemple, majorité absolue, majorité simple, unanimité, ...).

Lorsqu'un vote est valide, la proposition fait l'objet d'une et une seule décision. Quand un vote n'est pas valide la proposition n'est reliée à aucune décision. On **numérotera** chaque décision, et une séquence de

numéro sera propre à chaque AG. En dehors de ce numéro, une décision est caractérisée par son **texte**, par exemple: accord, refus, descriptif. ...

On met alors en évidence les éléments d'information ci-dessous :

Mnémonique	Signification
N° de l'AG dans l'année	Déjà considéré
Date envoi convocation	Date d'envoi de la convocation
Numéro propriétaire présent	Déjà considéré
NumProcuration	Déjà considéré
Numéro lot	Déjà considéré
Description Lot	Déjà considéré
Tantième Lot	Déjà considéré
Type lot	Type de lot
numProposition	Déjà considéré
Type vote	Type de vote
numDécision	Numéro de la décision
texteDécision	Texte de la décision

Après avoir fait ce travail de recensement, en veillant à éliminer les doublons, on obtient le dictionnaire de données suivant

Mnémonique	Signification
N° de l'AG dans l'année	Numéro de l'assemblée générale dans l'année
Date AG	Date de l'assemblée générale
Numéro propriétaire	Numéro du propriétaire
Nom propriétaire	Nom (patronyme) du propriétaire
Prénom propriétaire	Prénom du propriétaire
Rue propriétaire	Numéro et rue du propriétaire
CP propriétaire	Code postal du propriétaire
Ville propriétaire	Ville du propriétaire
Téléphone propriétaire	Téléphone du propriétaire
Numéro lot	Numéro de lot
Description Lot	Description du lot
Tantième Lot	Tantième associé au lot
numProcuration	Numéro de la procuration
dateProcuration	Date de la procuration
numProposition	Numéro de la proposition
texteProposition	Texte de la proposition
DateEnvoiConvoc	Date d'envoi de la convocation
TypeLot	Type de lot
TypeVote	Type de vote

numDécision	Numéro de la décision
texteDécision	Texte de la décision

On peut alors déterminer la couverture minimale en 4 étapes :

- Étape 1 : détermination des éléments qui peuvent être utilisés comme sources

On sélectionne déjà les éléments qui étaient les identifiants des 3 entités proposées dans l'annexe A :

- N° de l'AG dans l'année
- Numéro propriétaire
- Numéro lot

On cherche ensuite, parmi les nouveaux éléments d'information, qu'on a déterminés, ceux qui peuvent jouer le rôle de sources. On obtient :

- numProcuration
- numProposition
- numDécision
- TypeLot

- Étape 2 : détermination des éléments cibles associés aux éléments sources de l'étape 1

Comme précédemment, on reprend d'abord les éléments issu du MCD partiel, on obtient donc :

- N° de l'AG dans l'année → Date AG
- Numéro propriétaire → Nom propriétaire, Prénom propriétaire, Rue propriétaire, CP propriétaire, Ville propriétaire, Téléphone propriétaire
- Numéro lot → Description Lot, Tantième Lot

On s'intéresse ensuite aux autres sources, déterminées à l'étape précédente :

- numProcuration → dateProcuration
- numProposition → texteProposition, typeVote
- numDécision → texteDécision
- Type lot → TypeLot

- Étape 3 : détermination des DF ayant comme source une combinaison d'éléments d'information

On recherche les éléments d'information restant dans le dictionnaire de données : il ne reste que « DateEnvoiConvoc ». Pour déterminer cette information, on doit connaître le numéro de l'AG et le numéro du propriétaire ce qui se traduit par la DF suivante :

N° de l'AG dans l'année + Numéro propriétaire → DateEnvoiConvoc

- Étape 4: détermination des DF entre les éléments source mis en évidence à l'étape 1

On essaye de déterminer des DF entre les éléments suivants :

- N° de l'AG dans l'année
- Numéro propriétaire
- Numéro lot
- numProcuration

- numProposition
- numDécision
- TypeLot

Dans le MCD partiel, on a une association 1-N entre LOT et PROPRIETAIRE, ce qui correspond à la DF :

Numéro lot → Numéro propriétaire

L'association N-N entre l'entité ayant comme identifiant N° de l'AG dans l'année et PROPRIETAIRE ne peut pas donner lieu à une DF entre deux entités jouant le rôle de source dans d'autres DF. Elle est néanmoins déjà prise en compte par la DF déterminée à l'étape 3 :

N° de l'AG dans l'année + Numéro propriétaire → DateEnvoiConvoc

On doit maintenant de nouveau analyser le texte de l'énoncé pour déterminer les autres DF

« Dans une AG, les propriétaires peuvent se faire représenter par un propriétaire de la même copropriété. Dans ce cas, une procuration est nécessaire (voir annexe 1). Une procuration, document numéroté, est utilisée pour une seule représentation à une seule AG. »

On constate qu'un numéro de procuration est associé à un seul numéro d'AG, on a donc :

numProcuration → N° de l'AG dans l'année

A partir d'un numéro de procuration, on peut également retrouver les numéro des propriétaires représenté et représentant, on a donc :

numProcuration → Numéro propriétaire représenté  
numProcuration → Numéro propriétaire représentant

« Une information supplémentaire caractérisant le lot est le type de lot (garage, appartement, place de parking, ...) »

A partir d'un numéro de lot, on peut déterminer son type, on a donc :

Numéro lot → typeLot

« Le déroulement d'une assemblée générale est une suite de votes correspondant à autant de propositions. Ainsi une AG est composée de propositions numérotées par AG (voir annexe 2). »

En d'autres termes, on peut remonter au numéro de l'AG à partir d'un numéro de proposition

numProposition → N° de l'AG dans l'année

« Une information supplémentaire caractérisant le lot est le type de lot (garage, appartement, place de parking, ...) »

On peut donc connaître le type du lot en connaissant son numéro :

Numéro lot → typeLot

« Une proposition concerne un certain nombre de types de lots. Il faut donc connaître ces derniers. Par exemple, certaines propositions se rapportent aux garages, d'autres aux appartements, d'autres encore à tous les types de lots ... »

Autrement dit, le numéro de la proposition ne permet pas de déduire le type de lot et inversement : il n'y a donc pas de DF entre numProposition et TypeLot

« Lorsqu'un vote est valide, la proposition fait l'objet d'une et une seule décision. Quand un vote n'est pas valide la proposition n'est reliée à aucune décision. »

Le numéro de la décision doit donc permettre de retrouver le numéro de la proposition qui a donné lieu au débat, au vote et à la décision :

numDécision → numProposition

Après avoir fait ce travail préparatoire, au brouillon, on peut maintenant déterminer les entités en considérant les DF de l'étape 2 :

- N° de l'AG dans l'année → Date AG

Cette DF correspond à une des entités du MCD partiel de l'annexe A. Elle ne porte cependant pas de nom, on doit donc lui attribuer un nom qui évoque son contenu : on choisit par exemple « ASSEMBLEE GENERALE »

ASSEMBLEE GENERALE
<u>N° de l'AG dans l'année</u> Date AG

- Numéro propriétaire → Nom propriétaire, Prénom propriétaire, Rue propriétaire, CP propriétaire, Ville propriétaire, Téléphone propriétaire

Cette deuxième DF correspond à une autre des entités du MCD partiel de l'annexe A : l'entité PROPRIETAIRE représentée ci-dessous.

PROPRIETAIRE
<u>Numéro propriétaire</u> Nom propriétaire Prénom propriétaire Rue propriétaire CP propriétaire Ville propriétaire Téléphone propriétaire

- Numéro lot → Description Lot, Tantième Lot

Cette troisième DF correspond à la troisième entité du MCD partiel de l'annexe A : l'entité LOT représentée ci-dessous.

LOT
<u>Numéro lot</u> Description lot Tantième lot

- numProcuracion → dateProcuracion

Cette DF correspond à une entité ayant comme identifiant numProcuracion car il s'agit de la source de la DF. La cible, dateProcuracion, joue le rôle de propriété au sein de cette entité.

On lui attribue un nom qui évoque son contenu, on choisit donc PROCURATION.

PROCURATION
<u>numProcuracion</u> dateProcuracion

- numProposition → texteProposition, typeVote

Cette DF correspond à une autre entité, qu'on choisit d'appeler PROPOSITION. Elle a comme identifiant, la source de la DF, c'est-à-dire numProposition. Les cibles, texteProposition et typeVote, jouent le rôle de propriétés au sein de cette entité.

PROPOSITION
<u>numProposition</u> texteProposition typeVote

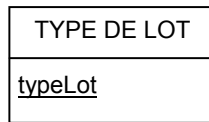
- numDécision → texteDécision

Cette autre DF correspond à l'entité DECISION ci-dessous :

DECISION
<u>numéroDécision</u> texteDécision

- typeLot → typeLot

Cette dernière DF est représentée par une entité qui ne comporte qu'un identifiant typeLot. Nous l'appelons TYPE DE LOT



On reprend ensuite les DF des étapes 3 et 4 pour déterminer les différentes associations

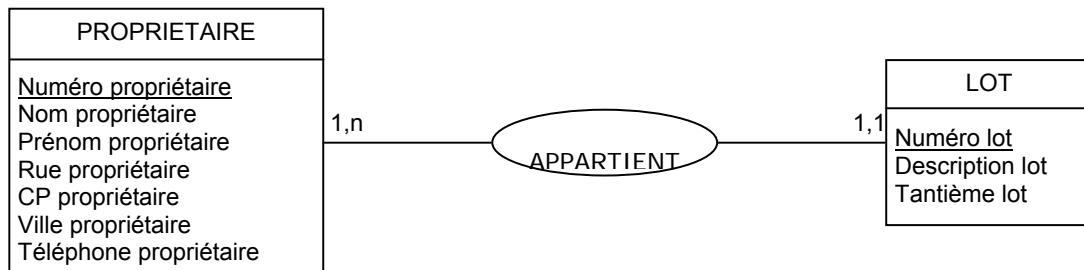
- Numéro lot → Numéro propriétaire

Cette DF correspond à l'association 1-N « Appartient » qui était présente dans le MCD partiel. Elle permet de déterminer les cardinalités maximales :

- 1 du côté de LOT
- N du côté de PROPRIETAIRE

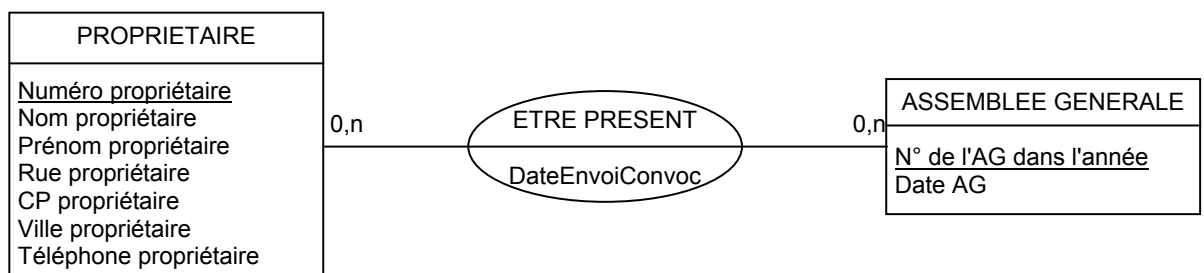
Les cardinalités minimale doivent être déterminées à partir des règles de gestion (de l'énoncé) ou du bon sens :

- un propriétaire a nécessairement un 1 lot donc la cardinalité minimale du côté de propriétaire est 1
- un lot a nécessairement un 1 propriétaire donc la cardinalité minimale du côté de lot est 1



- N° de l'AG dans l'année + Numéro propriétaire → DateEnvoiConvoc

Cette DF correspond à l'association N-N « Etre présent » entre les entités ASSEMBLEE GENERALE et PROPRIETAIRE. Par rapport au MCD, nous ajoutons la propriété « Date envoi convocation »





- numProcuration → Numéro propriétaire représenté
- numProcuration → Numéro propriétaire représentant

Ces deux DF correspondent à deux associations 1-N qui lient les entités PROPRIETAIRE et PROCURATION.

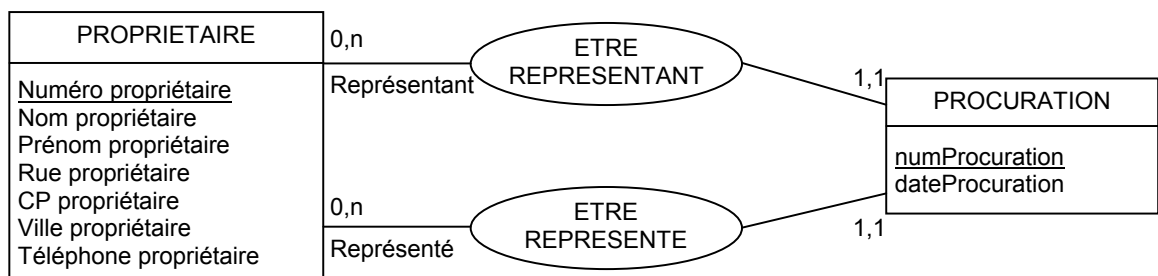
Les cardinalités maximales 1 sont placées du côté de procuration ce qui signifie qu'une procuration ne peut être signée que par un propriétaire qui ne peut désigner qu'un seul représentant pour cette procuration.

A l'inverse, un propriétaire peut représenter plusieurs personnes à des moments différents (voire à la même AG) et un propriétaire peut aussi se faire représenter plusieurs fois à des moments différents : les cardinalités maximales N sont donc placées du côté de PROPRIETAIRE

En ce qui concerne les cardinalités minimale, elles sont les suivantes :

- pour l'association « ETRE REPRESENTANT », on place :
  - 1 du côté de PROCURATION car il y a nécessairement une personne désignée par cette procuration
  - 0 du côté de PROPRIETAIRE car il y peut avoir un propriétaire qui n'a jamais joué le rôle de représentant
- pour l'association « ETRE REPRESENTE », on place :
  - 1 du côté de PROCURATION car il y a nécessairement une personne qui a signé cette procuration
  - 0 du côté de PROPRIETAIRE car il y peut avoir un propriétaire qui n'a jamais manqué d'AG.

On doit enfin ajouter des rôles du côté de l'entité PROPRIETAIRE pour clarifier le MCD car ces deux entités sont liées deux par deux associations différentes (ayant des sens différents). On ajoute donc les rôles « Représenté » et « Représentant ».



- numProcuration → N° de l'AG dans l'année

Cette DF correspond à une association de type 1-N :

- La cardinalité maximale 1 est du côté de PROCURATION, ce qui signifie que la procuration ne sert qu'à au plus une assemblée générale
- La cardinalité maximale N est du côté de ASSEMBLEE GENERALE, ce qui modélise le fait que plusieurs personnes peuvent se faire représenter pour une AG donnée

Pour les cardinalité minimale, on a :

- 1 est du côté de PROCURATION car une procuration est signée pour au moins une AG
- 0 du côté de ASSEMBLEE GENERALE car pour une AG donnée, il peut n'y avoir aucun propriétaire qui se fasse représenter



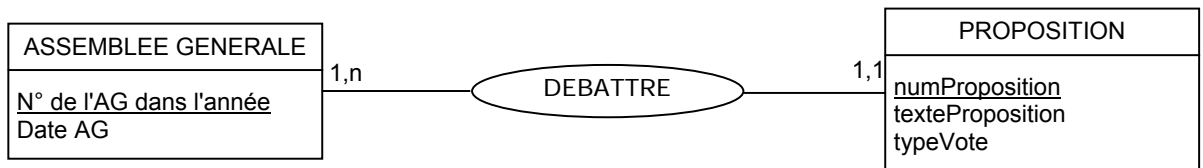
- Numéro proposition → N° de l'AG dans l'année

Cette DF correspond à une association de type 1-N :

- La cardinalité maximale 1 est du côté de PROPOSITION indique qu'une proposition est à l'ordre du jour d'au plus une AG
- La cardinalité maximale N est du côté de ASSEMBLEE GENERALE montre que l'ordre du jour peut comporter plusieurs propositions

Pour les cardinalité minimale, on a :

- 1 est du côté de PROPOSITION car une proposition est débattue durant au moins une AG
- 1 du côté de ASSEMBLEE GENERALE car pour une AG donnée, il doit y avoir au moins une proposition



- Numéro lot → typeLot

Cette DF correspond à une association de type 1-N :

- La cardinalité maximale 1 est du côté de LOT modélise le fait qu'un lot a au plus un type
- La cardinalité maximale N est du côté de TYPE DE LOT montre que plusieurs lots peuvent avoir le même type

Pour les cardinalité minimale, on a :

- 1 est du côté de LOT car un lot a au plus un type
- 1 du côté de TYPE DE LOT car il y a au moins un lot pour former cette catégorie



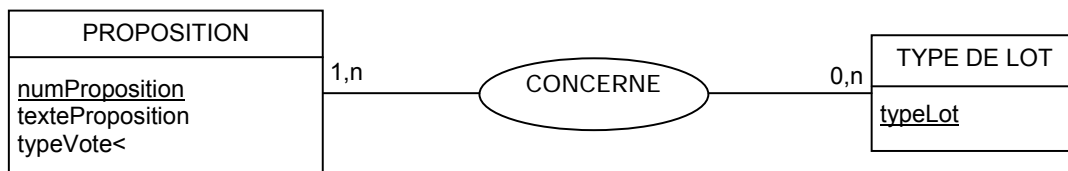
- L'association entre numProposition et TypeLot

Il n'y a pas de DF entre ces éléments, cela correspond à une association de type N-N :

- La cardinalité maximale N est du côté de PROPOSITION car une proposition peut concerner plusieurs types de lot
- La cardinalité maximale N est du côté de TYPE DE LOT montre qu'un type donné peut être concerné par plusieurs propositions

Pour les cardinalité minimale, on a :

- 1 est du côté de PROPOSITION car une proposition concerne au moins un type de lot
- 0 du côté de TYPE DE LOT car pour un type donné, il peut n'y avoir aucune proposition



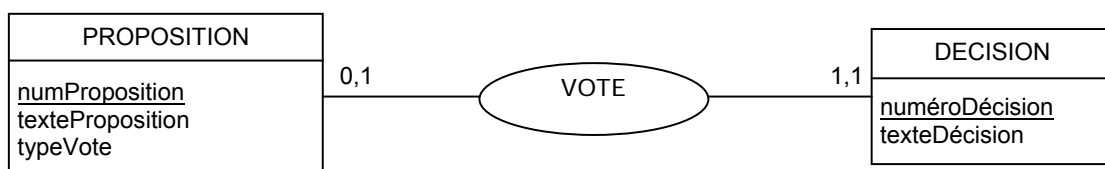
- numDécision → numProposition

Cette DF correspond à une association de type 1-1 :

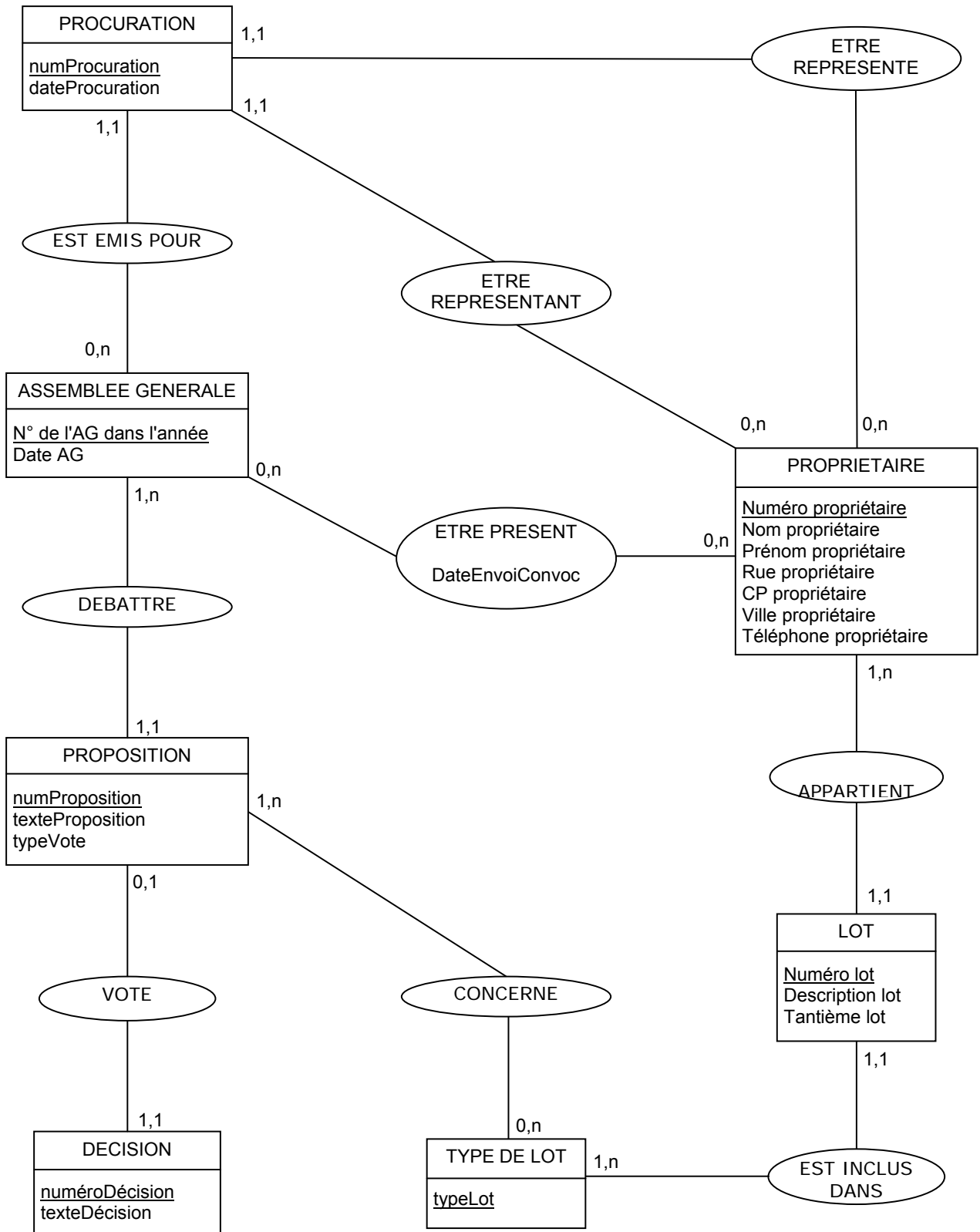
- La cardinalité maximale 1 est du côté de PROPOSITION car une proposition est associée à au plus une décision
- La cardinalité maximale 1 est du côté de DECISION car une décision est issue d'au plus une proposition

Pour les cardinalité minimale, on a :

- 0 est du côté de PROPOSITION car une proposition peut ne pas donner lieu à une décision si le vote est invalide
- 1 du côté de DECISION car une décision donnée est associée à au moins une proposition



On rassemble les différents éléments et on obtient le MCD ci-dessous



**DOSSIER 2 - ÉVOLUTION DU SYSTÈME D'INFORMATION**
**1. Au sein d'un tableau structuré, dresser pour ces deux propositions leurs avantages et leurs inconvénients en tenant compte de la spécificité de IMMO-DO.**

Un certain nombre d'éléments pouvaient être reportés depuis le texte dans le tableau ci-dessous. Les critères utilisés pour comparer ces deux systèmes correspondent aux thèmes abordés dans le texte.

Critère	PGI	Saas
Coût	« L'achat des licences, de support, de mises à jour et de frais d'exploitation »	« Un prix unique forfaitaire »
		« Le paiement s'effectue suivant le nombre d'utilisateurs, sans coût fixe forfaitaire, et mensuellement »
		« Un lissage des coûts informatiques, qui deviennent prédictibles »
Évolution du logiciel	L'évolution du logiciel se fait par « à-coups ». L'éditeur décide d'abord d'inclure des améliorations dans le logiciels qui est modifié pour produire une nouvelle version.	« L'amélioration continue des fonctionnalités. »
	L'utilisateur doit alors acquérir cette nouvelle version pour profiter des améliorations	« les mises à jour sont incluses dans le prix (à l'usage) : l'utilisateur n'a plus à changer de version tous les deux ou trois ans, et profite des derniers updates. »
	Un délai assez long pour profiter de plusieurs améliorations	Un délai beaucoup plus court entre chaque amélioration
Sensibilité aux réseaux	L'application est fournie comme un logiciel indépendant sur un DVDROM	« L'application est fournie comme un service, par le biais d'un portail web »
	Malgré la défaillance du réseau Internet, le PGI est toujours utilisable	Si le réseau Internet est indisponible (défaillance du FAI par exemple), le Saas n'est plus accessible
Prise en compte du feedback	« le système on-premise (déploiement des applications sur site), chaque client travaille sur sa petite île, et le feedback est hors d'atteinte »	« Cette différence amène les éditeurs à revoir constamment leur logiciel, et donc à l'améliorer, en étant plus à l'écoute du marché et des utilisateurs. Le feedback de ces derniers quant à l'usage quotidien de l'application permet l'amélioration continue des fonctionnalités. Le client, qui devient acteur et pilote de l'amélioration »
	Réactivité beaucoup moins bonne et prise en compte des feedback moins importante	Meilleure réactivité et meilleure prise en compte des feedback

## 2. Illustrer par des exemples la notion "d'amélioration continue des fonctionnalités".

L'utilisateur accède au logiciel (au service) par l'intermédiaire d'un portail web. Le logiciel est en réalité installé sur un ordinateur au sein du réseau de l'éditeur :

- un premier ordinateur, situé dans la DMZ du réseau de l'éditeur, contient un serveur web et quelques éléments logiciels secondaires (par exemple un interpréteur PHP)
- un second ordinateur, situé dans la partie privée du réseau de l'éditeur et qui contient le logiciel et sa base de données.

L'éditeur peut modifier directement le logiciel installé sur l'ordinateur au sein de son réseau, l'utilisateur verra l'effet de ces modifications lors de sa prochaine connexion au système.

Ces modifications peuvent concerner par exemple le système d'information par lui-même : on peut ainsi ajouter des nouvelles propriétés à des entités (par exemple, ajouter des propriétés comme la latitude, la longitude ou la surface dans l'entité LOT).

Ces modifications peuvent aussi toucher le fonctionnement du logiciel par lui-même :

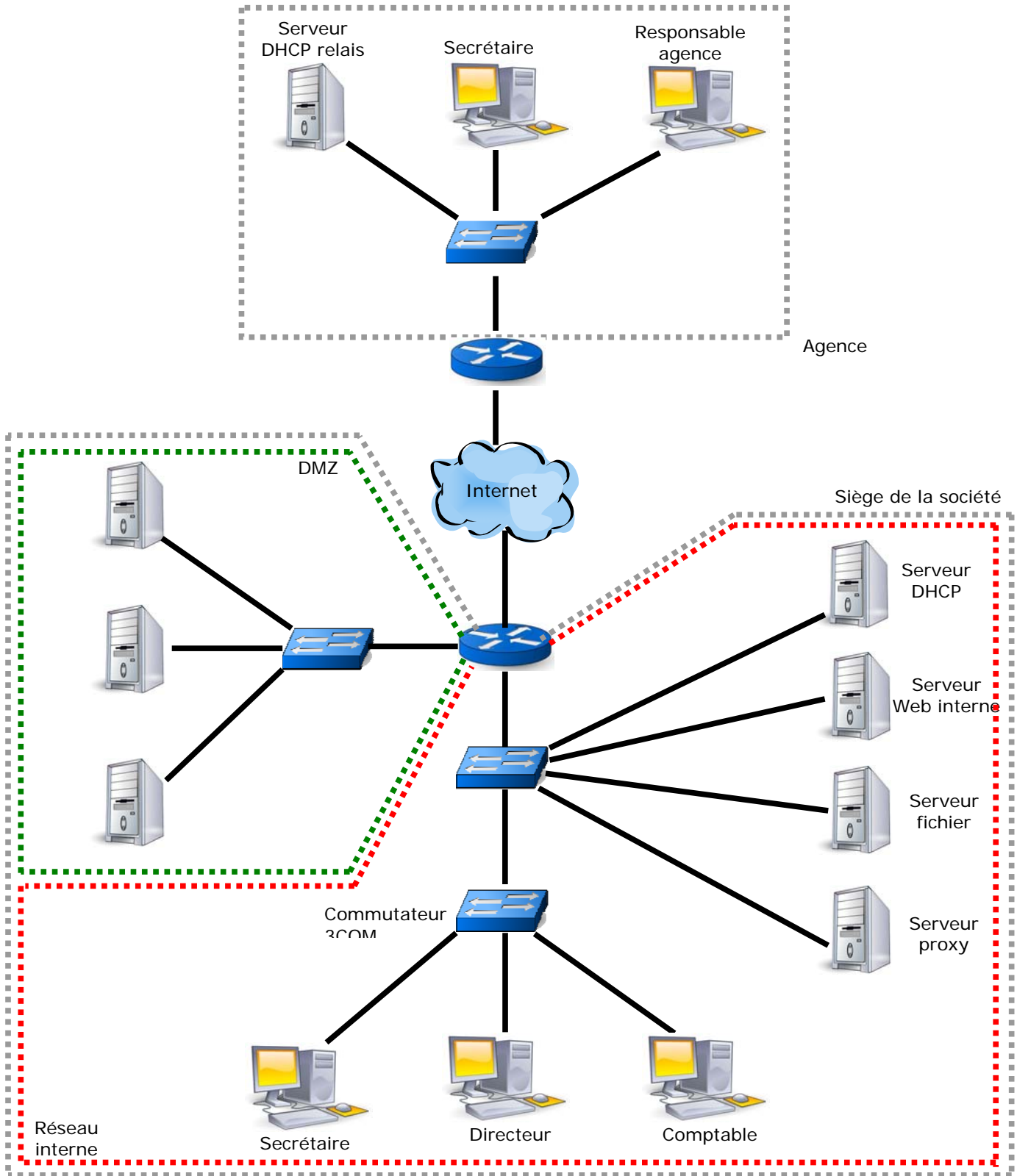
- amélioration de fonctionnalités existantes afin qu'elles prennent en compte des nouvelles propriétés ou qu'elles s'exécutent plus rapidement
- suppression ou ajout de nouvelles fonctionnalités

Ces modifications peuvent enfin impacter l'affichage :

- ajout de nouveaux boutons
- changements dans l'emplacement des éléments
- représentations différentes de certains éléments (par exemple utiliser un diagramme en bâtons à la place d'un diagramme circulaire)

**DOSSIER 3 - EVOLUTION DU RESEAU**

On représente le réseau de façon graphique à partir des éléments d'information présenté à l'annexe 3 afin de clarifier la situation. On obtient alors, par exemple, le schéma ci-dessous.



## 1. Décrire le rôle d'une DMZ (zone démilitarisée).

DMZ signifie « Demilitarized Zone ». Il s'agit d'une partie du réseau local de l'entreprise qui est accessible depuis l'extérieur (internet) et qui contient des serveurs qui doivent pouvoir être atteints par les internautes comme le serveur web de l'entreprise.

## 2. Décrire le rôle de chacun des serveurs présents dans l'annexe 4.

### - Le Serveur Proxy

Le serveur proxy, encore appelé serveur mandataire, est une machine qui est chargée de relayer les requêtes des ordinateurs du réseau interne vers Internet, et les réponses issues d'Internet vers les ordinateurs du réseau interne. Elle est donc utilisée comme un intermédiaire pour les communications entre les ordinateurs du réseau interne et Internet.

L'utilisation d'un serveur proxy procure les avantages suivants :

- Toutes les requêtes émanant d'une machine du réseau de l'entreprise seront relayées par cette machine intermédiaire : les paquets IP correspondant auront donc tous l'adresse IP source 172.30.32.12. Cela permet de cacher l'organisation interne du réseau de l'entreprise.
- Toutes les requêtes doivent passer par cette machine intermédiaire, il est donc aisé pour un administrateur réseau de mettre en place un filtre afin d'interdire l'accès à certains services (interdire le FTP par exemple) ou certains sites (Facebook ...etc).

### - Le serveur de fichiers

Le serveur de fichiers est utilisé par les différents employés de la société Immo-Dopour pour stocker de façon centralisée des documents qui peuvent être partagés entre les différents utilisateurs

### - Le Serveur WEB intranet

Ce serveur web n'est accédé qu'au sein du réseau de l'entreprise par des employés de la société Immo-Do. Il permet d'avoir accès à des pages web ou des applications utilisant les pages web comme interfaces graphiques. Cela peut être par exemple une application de gestion des biens immobiliers

### - Le Serveur DHCP

Le serveur DHCP (*Dynamic Host Configuration Protocol*) est un serveur qui renvoie des éléments de configuration (adresse IP de la machine, adresse IP de la passerelle, adresse IP du serveur DNS) lorsqu'une machine en fait la demande lors de son démarrage, par exemple (l'adresse IP de cette machine est alors placée à 0.0.0.0 dans le panneau de configuration).

### - Le serveur DNS

Le serveur DNS (*Domain Name Service*) a pour principale tâche de retourner l'adresse IP correspondant à un nom de machine ou le nom d'une machine correspondant à une adresse IP à une machine qui en fait la demande.



Par exemple, lorsqu'on saisit « <http://www.google.fr> » dans la barre d'adresse de notre navigateur et qu'on appuie sur « Go », une requête est envoyée vers le serveur DNS pour connaître l'adresse IP correspondant à la machine [www](http://www.google.fr) du domaine [google.fr](http://www.google.fr)

- Le serveur WEB internet

Le serveur web de la DMZ, à l'instar du serveur web interne, contient un logiciel tel qu'Apache qui répond à des requêtes du navigateur web comme Internet explorer ou Firefox. Le dialogue s'effectue selon le protocole HTTP (*HyperText Transfert Protocol*).

Dans le cas présent, ce serveur est accessible depuis Internet et il permet à des internautes de consulter, par exemple, les habitations proposées à la vente (et géré par Immo-Do)

- Le serveur messagerie Exchange

Ce serveur est utilisé pour envoyer et recevoir des messages électroniques. Il joue donc à la fois le rôle de MTA (*Message Transfer Agent*) et de MDA (*Message Delivery Agent*)

- Le serveur DHCP agent relais

Le protocole DHCP est conçu de manière à ce que les requêtes soient stoppées par les routeurs : elles restent donc confinées au sein d'un sous-réseau. Dans certains cas de figure, il est nécessaire de pouvoir utiliser le protocole DHCP sur plusieurs sous-réseau séparés par des routeurs.

La requête qui parvient à un relai DHCP est alors transmis directement au serveur DHCP (central). Ce dernier répond alors au serveur DHCP relai qui peut retransmettre l'information à la machine à l'initiative de la demande.

Certains routeur dispose de la fonctionnalité « relai DHCP »

### 3. Quelle est l'adresse du réseau du siège d'Avignon?

Les adresses des différentes machines commencent toutes par 172.30. Si on convertit le premier nombre en binaire, on obtient « 10101100 ». Comme ce nombre commence par « 10 », cela signifie qu'il s'agit d'une classe B dont le masque usuel est 255.255.0.0.

Cependant, dans l'annexe 3, on constate que les adresses IP sont accompagné du masque 255.255.255.0, cela signifie que l'administrateur réseau a agrandi le masque pour inclure le 3ème octet dans la partie « réseau » de l'adresse IP.

D'après cette annexe, les machines du siège de l'agence ont une adresse IP du type 172.30.32.X. Comme le masque réseau est 255.255.255.0, l'adresse réseau de l'agence est donc 172.30.32.0

### 4. Donner les avantages d'un tel choix.

L'intérêt d'utiliser les sous-réseaux est de pouvoir mettre en place, grâce au routeur et au mur de feu, des règles de routage et de filtrage. Ces règles peuvent, par exemple, interdire, pour des raisons de sécurité, certains accès provenant du réseau des agences vers le réseau du siège (par exemple, interdire l'accès à la comptabilité).

L'utilisation de ces sous-réseaux est également nécessaire pour mettre en place le réseau privé virtuel (Virtual Private Network ou VPN) entre les différentes agences et le siège : les routeurs ont besoin de sous-réseau différent pour véhiculer les paquets IP (appliquer les règles de routage)

**5. Avec les informations présentes dans l'annexe 4 et les critères énoncés ci-dessus, déterminer le nombre d'agences qui peuvent être créées en justifiant votre calcul.**

Les adresses des différentes machines commencent toutes par 172.30. Si on convertit le premier nombre en binaire, on obtient « 10101100 ». Comme ce nombre commence par « 10 », cela signifie qu'il s'agit d'une classe B dont le masque usuel est 255.255.0.0.

Dans l'annexe 3, on constate que les adresses IP sont accompagnés du masque 255.255.255.0, cela signifie que le 3<sup>ème</sup> octet est inclus dans la partie « réseau » de l'adresse IP.

On peut ainsi créer 256 sous-réseaux différents (numérotés de 0 à 255) au sein du réseau 172.30.0.0 :

- le réseau 172.30.0.0
- le réseau 172.30.1.0
- le réseau 172.30.2.0 ...
  
- le réseau 172.30.254.0
- le réseau 172.30.255.0

Chaque sous-réseau pourra contenir au plus 254 machines (le 0 et le 255 ne sont pas utilisés pour désigner une machine mais respectivement un réseau et une adresse de diffusion)

D'après l'annexe 3, le sous-réseau 172.30.32.0 est utilisé pour le siège de l'agence et le 172.30.32.0 pour un agence, on peut donc créer 254 autres agences.

**DOSSIER 4 – GESTION DES APPELS DE FONDS - TRAVAUX****1. Liste des entreprises (noms, sans doublon) ayant effectué des réalisations dans la copropriété "Bel Air" en 2009**

On analyse la requête afin de déterminer les éléments d'information à manipuler :

- le nom de l'entreprise correspondant au champ EntNom de ENTREPRISE
- le nom de la copropriété stocké dans le champ CopNom de COPROPRIETE
- l'année représentée par le RealDate de REALISATION

Il est donc nécessaire de réaliser une jointure entre ces 3 tables en utilisant comme conditions de jointure :

- REALISATION.EntNum = ENTREPRISE. EntNum
- REALISATION. CopNum= COPROPRIETE. CopNum

Pour réaliser cette jointure, on a trois possibilités :

- la première consiste à placer les conditions de jointure dans la clause WHERE

```
SELECT * FROM REALISATION, ENTREPRISE, COPROPRIETE WHERE  
(REALISATION.EntNum = ENTREPRISE. EntNum) AND (REALISATION. CopNum=  
COPROPRIETE. CopNum)
```

- la deuxième utilise la directive INNER JOIN

```
SELECT * FROM (REALISATION INNER JOIN ENTREPRISE ON REALISATION.EntNum =  
ENTREPRISE. EntNum) INNER JOIN , COPROPRIETE ON REALISATION. CopNum=  
COPROPRIETE. CopNum
```

- la troisième s'appuie sur la directive NATURAL JOIN car il s'agit d'une équijointure et les noms des clés primaires et étrangères sont identiques (il est à noter que cette directive n'est pas supportée par tous les moteurs SQL)

```
SELECT * FROM (REALISATION NATURAL JOIN ENTREPRISE) NATURAL JOIN  
COPROPRIETE
```

On doit modifier ces requêtes SQL afin de tenir compte du fait qu'on ne s'intéresse qu'aux réalisations concernant la copropriété "Bel Air" en 2009 et l'année 2009, en ajoutant ou modifiant la clause WHERE avec le test ci-dessous :

```
(COPROPRIETE.CopNom="Bel Air") AND (YEAR-REALISATION.RealDate)=2009)
```

On peut également utiliser des variantes pour tester le noms de la copropriété ou l'année :

- COPROPRIETE.CopNom LIKE "Bel Air"
- REALISATION.RealDate >= '01/01/2009' AND REALISATION.RealDate <= '01/01/2009'
- REALISATION.RealDate BETWEEN '01/01/2009' AND '01/01/2009'

On obtient donc l'une des 3 solutions ci-dessous :

```
SELECT * FROM REALISATION, ENTREPRISE, COPROPRIETE WHERE (REALISATION.EntNum =
ENTREPRISE. EntNum) AND (REALISATION. CopNum= COPROPRIETE. CopNum) AND
(COPROPRIETE.CopNom="Bel Air") AND (YEAR(REALISATION.RealDate)=2009)
```

```
SELECT * FROM (REALISATION INNER JOIN ENTREPRISE ON REALISATION.EntNum = ENTREPRISE.
EntNum) INNER JOIN , COPROPRIETE ON REALISATION. CopNum= COPROPRIETE. CopNum WHERE
(COPROPRIETE.CopNom="Bel Air") AND (YEAR(REALISATION.RealDate)=2009)
```

```
SELECT * FROM (REALISATION NATURAL JOIN ENTREPRISE) NATURAL JOIN COPROPRIETE
WHERE (COPROPRIETE.CopNom="Bel Air") AND (YEAR(REALISATION.RealDate)=2009)
```

On veut enfin la liste des noms des entreprises sans doublons, on doit donc écrire dans la clause SELECT :  
« DISTINCT ENTREPRISE. EntNom »

On obtient finalement l'une des trois solutions ci-dessous :

```
SELECT DISTINCT ENTREPRISE. EntNom FROM REALISATION, ENTREPRISE, COPROPRIETE
WHERE (REALISATION.EntNum = ENTREPRISE. EntNum) AND (REALISATION. CopNum=
COPROPRIETE. CopNum) AND (COPROPRIETE.CopNom="Bel Air") AND
(YEAR(REALISATION.RealDate)=2009)
```

```
SELECT DISTINCT ENTREPRISE. EntNom FROM (REALISATION INNER JOIN ENTREPRISE ON
REALISATION.EntNum = ENTREPRISE. EntNum) INNER JOIN , COPROPRIETE ON REALISATION.
CopNum= COPROPRIETE. CopNum WHERE (COPROPRIETE.CopNom="Bel Air") AND
(YEAR(REALISATION.RealDate)=2009)
```

```
SELECT DISTINCT ENTREPRISE. EntNom FROM (REALISATION NATURAL JOIN ENTREPRISE)
NATURAL JOIN COPROPRIETE WHERE (COPROPRIETE.CopNom="Bel Air") AND
(YEAR(REALISATION.RealDate)=2009)
```

## **2. Liste des propriétaires (nom, prénom, téléphone), de la copropriété "Pelican" classés par ordre alphabétique (nom et prénom), dont le numéro de téléphone a été renseigné.**

Comme précédemment, on analyse le texte de l'énoncé afin de déterminer les éléments d'information à manipuler :

- le nom du propriétaire (le champ PropNom de PROPRIETAIRE)
- le prénom du propriétaire (le champ PropPrenom de PROPRIETAIRE)
- le numéro de téléphone du propriétaire (le champ TelProp de PROPRIETAIRE)
- le nom de la copropriété (le CopNom de COPROPRIETE)

Cette fois encore, comme on doit manipuler des tables différentes, on doit réaliser une jointure en utilisant la conditions de jointure :

PROPRIETAIRE. CopNum= COPROPRIETE. CopNum

1<sup>ère</sup> Ecole en ligne des professions

On a vu précédemment que cette jointure pouvait être réalisée de 3 manières, on obtient donc les 3 possibilités ci-dessous :

```
SELECT * FROM PROPRIETAIRE, COPROPRIETE WHERE (PROPRIETAIRE. CopNum=
COPROPRIETE. CopNum)
```

```
SELECT * FROM PROPRIETAIRE INNER JOIN COPROPRIETE ON PROPRIETAIRE. CopNum=
COPROPRIETE. CopNum
```

```
SELECT * FROM PROPRIETAIRE NATURAL JOIN COPROPRIETE
```

On s'intéresse aux enregistrements dont le numéro de téléphone a été renseigné, autrement dit, les enregistrements où PROPRIETAIRE. TelProp n'est pas nul. On peut considérer que ce champ est de type « chaîne de caractères » (CHAR, VARCHAR ...), on peut donc utiliser le test ci-dessous dans la clause WHERE (à ajouter ou à modifier)

```
PROPRIETAIRE. TelProp <> "" (telProp doit être différent de la chaîne vide)
```

Une solution plus élégante consiste à utiliser l'instruction IS NOT NULL. On écrit alors

```
PROPRIETAIRE. TelProp IS NOT NULL
```

On obtient donc l'une des trois solutions suivantes :

```
SELECT * FROM PROPRIETAIRE, COPROPRIETE WHERE (PROPRIETAIRE. CopNum=
COPROPRIETE. CopNum) AND (PROPRIETAIRE. TelProp IS NOT NULL)
```

```
SELECT * FROM PROPRIETAIRE INNER JOIN COPROPRIETE ON PROPRIETAIRE. CopNum=
COPROPRIETE. CopNum WHERE PROPRIETAIRE. TelProp IS NOT NULL
```

```
SELECT * FROM PROPRIETAIRE NATURAL JOIN COPROPRIETE WHERE PROPRIETAIRE. TelProp IS
NOT NULL
```

Enfin pour restreindre l'affichage au nom, prénom et numéro de téléphone des propriétaires, il suffit de remplacer, dans la clause SELECT, l'étoile par PROPRIETAIRE.PropNom, PROPRIETAIRE. PropPrenom, PROPRIETAIRE.TelProp

On obtient finalement l'une des 3 solutions suivantes :

```
SELECT PROPRIETAIRE.PropNom, PROPRIETAIRE. PropPrenom, PROPRIETAIRE.TelProp FROM
PROPRIETAIRE, COPROPRIETE WHERE (PROPRIETAIRE. CopNum= COPROPRIETE. CopNum) AND
(PROPRIETAIRE. TelProp IS NOT NULL)
```

```
SELECT PROPRIETAIRE.PropNom, PROPRIETAIRE. PropPrenom, PROPRIETAIRE.TelProp FROM
PROPRIETAIRE INNER JOIN COPROPRIETE ON PROPRIETAIRE. CopNum= COPROPRIETE. CopNum
WHERE PROPRIETAIRE. TelProp IS NOT NULL
```

```
SELECT PROPRIETAIRE.PropNom, PROPRIETAIRE. PropPrenom, PROPRIETAIRE.TelProp FROM
PROPRIETAIRE NATURAL JOIN COPROPRIETE WHERE PROPRIETAIRE. TelProp IS NOT NULL
```

**3. Liste des paiements reçus (montant, date, nom de la réalisation) pour la réalisation nommée "Réfection chaufferie" et pour la réalisation nommée "Mise en place TNT".**

On analyse l'énoncé et on constate qu'on doit manipuler les éléments d'information suivants :

- le montant du paiement (champ PmtMontant de PAIEMENT)
- la date du paiement (champ PmtDate de PAIEMENT)
- le nom de la réalisation (champ RealNom de REALISATION)

Les éléments utilisés se trouvent dans des tables différentes, on doit donc joindre ces tables. Comme elles n'ont pas d'éléments en commun, on doit trouver une ou plusieurs tables intermédiaires. La table PAIEMENT possède la clé étrangère AppelNum vers la table APPEL qui possède elle-même une clé étrangère RealNum vers REALISATION : on doit donc faire une jointure entre les tables PAIEMENT, APPEL et REALISATION en utilisant les conditions de jointure suivantes :

- REALISATION.RealNum = APPEL.RealNum
- PAIEMENT.AppelNum = APPEL.AppelNum

On obtient donc l'une des trois solutions suivantes :

```
SELECT * FROM PAIEMENT, APPEL, REALISATION WHERE (REALISATION.RealNum = APPEL.RealNum) AND (PAIEMENT.AppelNum = APPEL.AppelNum)
```

```
SELECT * FROM (PAIEMENT INNER JOIN APPEL ON PAIEMENT.AppelNum = APPEL.AppelNum) INNER JOIN REALISATION ON REALISATION.RealNum = APPEL.RealNum
```

```
SELECT * FROM (PAIEMENT NATURAL JOIN APPEL) NATURAL JOIN REALISATION
```

On s'intéresse uniquement aux réalisations appelées "Réfection chaufferie" et "Mise en place TNT", on doit donc ajouter ou modifier une clause WHERE avec la condition ci-dessous :

```
(REALISATION.RealNom = "Réfection chaufferie") AND (REALISATION.RealNom = "Mise en place TNT")
```

On obtient alors l'une des 3 solutions suivantes :

```
SELECT * FROM PAIEMENT, APPEL, REALISATION WHERE (REALISATION.RealNum = APPEL.RealNum) AND (PAIEMENT.AppelNum = APPEL.AppelNum) AND (REALISATION.RealNom = "Réfection chaufferie") AND (REALISATION.RealNom = "Mise en place TNT")
```

```
SELECT * FROM (PAIEMENT INNER JOIN APPEL ON PAIEMENT.AppelNum = APPEL.AppelNum) INNER JOIN REALISATION ON REALISATION.RealNum = APPEL.RealNum WHERE (REALISATION.RealNom = "Réfection chaufferie") AND (REALISATION.RealNom = "Mise en place TNT")
```

```
SELECT * FROM (PAIEMENT NATURAL JOIN APPEL) NATURAL JOIN REALISATION WHERE (REALISATION.RealNom = "Réfection chaufferie") AND (REALISATION.RealNom = "Mise en place TNT")
```

Il ne reste plus qu'à remplacer l'étoile de la clause SELECT par « PAIEMENT.PmtMontant, PAIEMENT.PmtDate, REALISATION.RealNom » pour afficher le montant du paiement, la date et le nom de la réalisation comme demandé dans l'énoncé.

On obtient alors l'une des 3 requêtes ci-dessous :

```
SELECT PAIEMENT.PmtMontant, PAIEMENT.PmtDate, REALISATION.RealNom FROM PAIEMENT,
APPEL, REALISATION WHERE (REALISATION.RealNum = APPEL.RealNum) AND
(PAIEMENT.AppelNum = APPEL.AppelNum) AND (REALISATION.RealNom = "Réfection chaufferie") AND
(REALISATION.RealNom = "Mise en place TNT")
```

```
SELECT PAIEMENT.PmtMontant, PAIEMENT.PmtDate, REALISATION.RealNom FROM (PAIEMENT
INNER JOIN APPEL ON PAIEMENT.AppelNum = APPEL.AppelNum) INNER JOIN REALISATION ON
REALISATION.RealNum = APPEL.RealNum WHERE (REALISATION.RealNom = "Réfection chaufferie")
AND (REALISATION.RealNom = "Mise en place TNT")
```

```
SELECT PAIEMENT.PmtMontant, PAIEMENT.PmtDate, REALISATION.RealNom FROM (PAIEMENT
NATURAL JOIN APPEL) NATURAL JOIN REALISATION WHERE (REALISATION.RealNom = "Réfection
chaufferie") AND (REALISATION.RealNom = "Mise en place TNT")
```

#### **4. Montant total des paiements reçus pour chaque réalisation (numéro, description, montant de la réalisation, total des paiements).**

On détermine les éléments d'information à manipuler en analysant l'énoncé :

- le numéro de la réalisation (champ RealNum de la table REALISATION)
- la description de la réalisation (champ RealDescription de la table REALISATION)
- le montant de la réalisation (champ RealMontant de la table REALISATION)
- le total des paiement (utilisation du champ PmtMontant de la table PAIEMENT)

On met en jeu les mêmes tables que pour la question précédente, on doit donc, de nouveau, utiliser la table intermédiaire APPEL pour faire une jointure entre ces 3 tables en utilisant l'une des requête ci-dessous :

```
SELECT * FROM PAIEMENT, APPEL, REALISATION WHERE (REALISATION.RealNum =
APPEL.RealNum) AND (PAIEMENT.AppelNum = APPEL.AppelNum)
```

```
SELECT * FROM (PAIEMENT INNER JOIN APPEL ON PAIEMENT.AppelNum = APPEL.AppelNum)
INNER JOIN REALISATION ON REALISATION.RealNum = APPEL.RealNum
```

```
SELECT * FROM (PAIEMENT NATURAL JOIN APPEL) NATURAL JOIN REALISATION
```

On veut afficher les informations pour chaque réalisation, en particulier la somme des paiements. Il faut donc :

- utiliser la fonction SUM appliquée au champ PmtMontant de la table PAIEMENT
- regrouper les enregistrements par rapport au numéro de réalisation en utilisant la clause GROUP BY suivi de REALISATION.RealNum

En outre, comme on veut afficher le numéro de la réalisation, sa description, son montant de la réalisation, total des paiements, on remplace l'étoile de la clause SELECT par « REALISATION.RealNum, REALISATION.RealDescription, REALISATION.RealMontant, SUM(PAIEMENT.PmtMontant) »

Finalement, on obtient l'une des trois solutions ci-dessous :

```
SELECT REALISATION.RealNum, REALISATION.RealDescription, REALISATION.RealMontant,  
SUM(PAIEMENT.PmtMontant) FROM PAIEMENT, APPEL, REALISATION WHERE  
(REALISATION.RealNum = APPEL.RealNum) AND (PAIEMENT.AppelNum = APPEL.AppelNum)
```

```
SELECT REALISATION.RealNum, REALISATION.RealDescription, REALISATION.RealMontant,  
SUM(PAIEMENT.PmtMontant) FROM (PAIEMENT INNER JOIN APPEL ON PAIEMENT.AppelNum =  
APPEL.AppelNum) INNER JOIN REALISATION ON REALISATION.RealNum = APPEL.RealNum
```

```
SELECT REALISATION.RealNum, REALISATION.RealDescription, REALISATION.RealMontant,  
SUM(PAIEMENT.PmtMontant) FROM (PAIEMENT NATURAL JOIN APPEL) NATURAL JOIN  
REALISATION
```

### **5. Liste des appels (Numéro, nom et prénom du propriétaire) pour lesquels un paiement n'a pas encore été reçu.**

Les éléments d'information qui doivent être manipulés sont les suivants :

- le numéro de l'appel (champ AppelNum de la table APPEL)
- le nom du propriétaire (champ PropNom de la table PROPRIETAIRE)
- le prénom du propriétaire (champ PropPrenom de la table PROPRIETAIRE)

On doit donc faire une jointure entre ces deux tables en utilisant la condition de jointure :

APPEL. PropNum = PROPRIETAIRE.PropNum

On obtient alors l'une des requêtes ci-dessous :

```
SELECT * FROM APPEL, PROPRIETAIRE WHERE (APPEL. PropNum = PROPRIETAIRE.PropNum)
```

```
SELECT * FROM APPEL INNER JOIN PROPRIETAIRE ON APPEL. PropNum = PROPRIETAIRE.PropNum
```

```
SELECT * FROM APPEL NATURAL JOIN PROPRIETAIRE
```

Ces 3 requêtes permettent d'obtenir tous les appels : ceux pour qui un paiement a été reçu et ceux pour qui un paiement n'a pas été reçu.

Ceux pour qui un paiement a été reçu sont caractérisés par l'existence d'un enregistrement dans la table PAIEMENT. Pour obtenir la liste des numéros d'appels correspondant, on peut utiliser la requête suivante :

SELECT PAIEMENT.AppelNum FROM PAIEMENT.

On peut alors combiner l'une des trois premières requêtes avec cette seconde requête en l'imbriquant dans la clause WHERE (à modifier ou à ajouter). La condition est alors la suivante :

AppelNum NOT IN (SELECT PAIEMENT.AppelNum FROM PAIEMENT)

Cela signifie qu'on s'intéresse à tous les enregistrements de la jointure entre PROPRIETAIRE et APPEL où AppelNum n'appartient pas à l'ensemble des AppelNum retournés par la sous-requête « SELECT PAIEMENT.AppelNum FROM PAIEMENT ».



1<sup>ère</sup> Ecole en ligne des professions

On obtient donc l'une des 3 requêtes ci-dessous :

```
SELECT * FROM APPEL, PROPRIETAIRE WHERE (APPEL. PropNum = PROPRIETAIRE.PropNum) AND  
(AppelNum NOT IN (SELECT PAIEMENT.AppelNum FROM PAIEMENT))
```

```
SELECT * FROM APPEL INNER JOIN PROPRIETAIRE ON APPEL. PropNum = PROPRIETAIRE.PropNum  
WHERE AppelNum NOT IN (SELECT PAIEMENT.AppelNum FROM PAIEMENT)
```

```
SELECT * FROM APPEL NATURAL JOIN PROPRIETAIRE WHERE AppelNum NOT IN (SELECT  
PAIEMENT.AppelNum FROM PAIEMENT)
```

Il ne reste alors plus qu'à remplacer l'étoile du premier SELECT par « APPEL.AppelNum, PROPRIETAIRE. PropNom, PROPRIETAIRE.PropPrenom » pour afficher le numéro de l'appel, le nom et le prénom du propriétaire comme demandé dans l'énoncé. On obtient alors l'une des 3 requêtes ci-dessous :

```
SELECT APPEL.AppelNum, PROPRIETAIRE. PropNom, PROPRIETAIRE.PropPrenom FROM APPEL,  
PROPRIETAIRE WHERE (APPEL. PropNum = PROPRIETAIRE.PropNum) AND (AppelNum NOT IN  
(SELECT PAIEMENT.AppelNum FROM PAIEMENT))
```

```
SELECT APPEL.AppelNum, PROPRIETAIRE. PropNom, PROPRIETAIRE.PropPrenom FROM APPEL  
INNER JOIN PROPRIETAIRE ON APPEL. PropNum = PROPRIETAIRE.PropNum WHERE AppelNum NOT  
IN (SELECT PAIEMENT.AppelNum FROM PAIEMENT)
```

```
SELECT APPEL.AppelNum, PROPRIETAIRE. PropNom, PROPRIETAIRE.PropPrenom FROM APPEL  
NATURAL JOIN PROPRIETAIRE WHERE AppelNum NOT IN (SELECT PAIEMENT.AppelNum FROM  
PAIEMENT)
```

## **6. Nom de l'entreprise ayant effectué la réalisation la plus onéreuse concernant la copropriété "Le Bel Air".**

Selon l'énoncé, on doit manipuler les éléments d'information ci-dessous :

- le nom de l'entreprise (champ EntNom de la table ENTREPRISE)
- le montant de la réalisation pour connaître celle qui est la plus onéreuse (champ RealMontant de la table REALISATION)
- le nom du copropriété (champ CopNom de la table COPROPRIETE)

Cependant, contrairement à la question 1, on ne va pas faire « directement » de jointure entre ces trois tables.

On commence par faire une jointure entre la table COPROPRIETE et REALISATION en utilisant l'une des requêtes ci-dessous :

```
SELECT * FROM REALISATION, COPROPRIETE WHERE REALISATION. CopNum= COPROPRIETE.  
CopNum
```

```
SELECT * FROM REALISATION INNER JOIN COPROPRIETE ON REALISATION. CopNum=  
COPROPRIETE. CopNum
```

```
SELECT * FROM REALISATION NATURAL JOIN COPROPRIETE
```

On ajoute (ou on modifie) ensuite la clause WHERE de manière ajouter la condition sur le nom de la copropriété. On obtient alors l'une des requêtes ci-dessous :

```
SELECT * FROM REALISATION, COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE. CopNum) AND ( COPROPRIETE. CopNom = "Le Bel Air")
```

```
SELECT * FROM REALISATION INNER JOIN COPROPRIETE ON REALISATION. CopNum= COPROPRIETE. CopNum WHERE COPROPRIETE. CopNom = "Le Bel Air"
```

```
SELECT * FROM REALISATION NATURAL JOIN COPROPRIETE WHERE COPROPRIETE. CopNom = "Le Bel Air"
```

Remarque : on peut remplacer COPROPRIETE. CopNom = "Le Bel Air" par COPROPRIETE. CopNom LIKE "Le Bel Air"

On remplace alors l'étoile de la clause SELECT par MAX (REALISATION. RealMontant) pour obtenir la valeur maximale enregistrée dans ce champs pour les enregistrements sélectionnés. On obtient alors les 3 requêtes suivantes :

```
SELECT MAX (REALISATION. RealMontant) FROM REALISATION, COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE. CopNum) AND ( COPROPRIETE. CopNom = "Le Bel Air")
```

```
SELECT MAX (REALISATION. RealMontant) FROM REALISATION INNER JOIN COPROPRIETE ON REALISATION. CopNum= COPROPRIETE. CopNum WHERE COPROPRIETE. CopNom = "Le Bel Air"
```

```
SELECT MAX (REALISATION. RealMontant) FROM REALISATION NATURAL JOIN COPROPRIETE WHERE COPROPRIETE. CopNom = "Le Bel Air"
```

Ces 3 requêtes nous permettent donc d'obtenir le montant de la réalisation la plus onéreuse dans la copropriété Bel Air. On va utiliser l'une d'elles comme sous-requête (ou requête imbriquée) de la requête qu'on va construire maintenant.

On commence d'abord par faire une jointure entre les tables REALISATION et ENTREPRISE en utilisant l'une des requêtes ci-dessous :

```
SELECT * FROM REALISATION, ENTREPRISE WHERE REALISATION.EntNum = ENTREPRISE. EntNum
```

```
SELECT * FROM REALISATION INNER JOIN ENTREPRISE ON REALISATION.EntNum = ENTREPRISE. EntNum
```

```
SELECT * FROM REALISATION NATURAL JOIN ENTREPRISE
```

On sélectionne alors l'enregistrement tel que la valeur de REALISATION. RealMontant soit égale à la valeur de la réalisation la plus onéreuse faite à la copropriété Bel Air en ajoutant ou modifiant la clause WHERE avec l'une des conditions ci-dessous :

```
REALISATION. RealMontant = (SELECT MAX (REALISATION. RealMontant) FROM REALISATION, COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE. CopNum) AND ( COPROPRIETE. CopNom = "Le Bel Air"))
```

```
REALISATION. RealMontant = (SELECT MAX (REALISATION. RealMontant) FROM REALISATION INNER JOIN COPROPRIETE ON REALISATION. CopNum= COPROPRIETE. CopNum WHERE COPROPRIETE. CopNom = "Le Bel Air")
```

```
REALISATION. RealMontant = (SELECT MAX (REALISATION. RealMontant) FROM REALISATION  
NATURAL JOIN COPROPRIETE WHERE COPROPRIETE. CopNom = "Le Bel Air")
```

Remarque : on peut mettre le mot-clé IN à la place du égal et ainsi écrire  
REALISATION. RealMontant IN (SELECT MAX ...etc)

On obtient alors différentes combinaisons possibles de requêtes parmi lesquelles on a :

```
SELECT * FROM REALISATION, ENTREPRISE WHERE (REALISATION.EntNum = ENTREPRISE.  
EntNum) AND (REALISATION. RealMontant = (SELECT MAX (REALISATION. RealMontant) FROM  
REALISATION, COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE. CopNum) AND (  
COPROPRIETE. CopNom = "Le Bel Air")))
```

```
SELECT * FROM REALISATION INNER JOIN ENTREPRISE ON REALISATION.EntNum = ENTREPRISE.  
EntNum WHERE REALISATION. RealMontant = (SELECT MAX (REALISATION. RealMontant) FROM  
REALISATION, COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE. CopNum) AND (  
COPROPRIETE. CopNom = "Le Bel Air"))
```

```
SELECT * FROM REALISATION NATURAL JOIN ENTREPRISE WHERE REALISATION. RealMontant =  
(SELECT MAX (REALISATION. RealMontant) FROM REALISATION, COPROPRIETE WHERE  
(REALISATION. CopNum= COPROPRIETE. CopNum) AND ( COPROPRIETE. CopNom = "Le Bel Air"))
```

Enfin, comme on ne veut afficher que le nom de l'entreprise donc on remplace l'étoile du SELECT par  
ENTREPRISE.EntNom, on obtient donc :

```
SELECT ENTREPRISE.EntNom FROM REALISATION, ENTREPRISE WHERE (REALISATION.EntNum =  
ENTREPRISE. EntNum) AND (REALISATION. RealMontant = (SELECT MAX (REALISATION.  
RealMontant) FROM REALISATION, COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE.  
CopNum) AND ( COPROPRIETE. CopNom = "Le Bel Air")))
```

```
SELECT ENTREPRISE.EntNom FROM REALISATION INNER JOIN ENTREPRISE ON  
REALISATION.EntNum = ENTREPRISE. EntNum WHERE REALISATION. RealMontant = (SELECT MAX  
(REALISATION. RealMontant) FROM REALISATION, COPROPRIETE WHERE (REALISATION. CopNum=  
COPROPRIETE. CopNum) AND ( COPROPRIETE. CopNom = "Le Bel Air"))
```

```
SELECT ENTREPRISE.EntNom FROM REALISATION NATURAL JOIN ENTREPRISE WHERE  
REALISATION. RealMontant = (SELECT MAX (REALISATION. RealMontant) FROM REALISATION,  
COPROPRIETE WHERE (REALISATION. CopNum= COPROPRIETE. CopNum) AND ( COPROPRIETE.  
CopNom = "Le Bel Air"))
```

**DOSSIER 5 – GESTION DES LOTS ET TANTIEMES SUR TABLEUR****1. Travail sur tableur (annexes 6.A et 6.B)****La formule de la cellule K2 permettant d'obtenir le nombre de lots de nature "parking"(nature encellule J2)**

On doit dénombrer les cellules qui appartiennent à la colonne C et qui contiennent le mot inscrit dans la cellule J2.

On utilise donc la fonction NB.SI qui prend deux arguments :

- le premier argument correspond à la plage de cellules sur laquelle la recherche est effectuée (ici il s'agit de la plage de cellules C2:C154
- le deuxième argument est le critère qui doit être respecté par la cellule pour être comptabilisé (ici, il s'agit du contenu de la cellule J2, on écrit donc cette référence dans la formule).

On obtient donc la formule « =NB.SI(C2:C154;J2) »

Il est cependant mentionné que cette formule doit pouvoir être utilisée (copiée) dans les cellules K3 à K8. Il est donc nécessaire de bloquer les numéros de lignes de la plage de cellules (sinon la plage de recherche sera décalée vers le bas).

La formule finale est donc :

=NB.SI(C\$2:C\$154;J2)

**La formule de la cellule L2 permettant d'obtenir le total des tantièmes pour les lots de nature "parking"**

Pour la cellule L2, nous devons cette fois-ci faire une somme conditionnelle des tantièmes contenus dans la colonne F.

On utilise donc la fonction SOMME.SI qui prend trois arguments :

- le premier argument correspond à la plage de cellules sur laquelle la recherche est effectuée (ici il s'agit de la plage de cellules C2:C154 comme précédemment) ;
- le deuxième argument est le critère qui doit être respecté par la cellule pour être comptabilisé (ici, il s'agit du contenu de la cellule J2, on écrit donc cette référence dans la formule comme précédemment) ;
- le troisième argument est la plage de cellules contenant les valeurs à sommer (ici, il s'agit de tantièmes contenus dans la plage de cellules F2:F154.

On obtient donc la formule « =SOMME.SI(C2:C154;J2;F2:F154) »

Comme précédemment, il est indiqué que cette formule doit pouvoir être utilisée (copiée) dans les cellules L3 à L8, il faut donc brider les numéros de lignes des plages de cellules du tableau 6.A.

La formule finale est donc :

=SOMME.SI(C\$2:C\$154;J2;F\$2:F\$154)

### La formule de la cellule K9 permettant d'obtenir le total du nombre de lots

Pour compter le nombre de lots, on peut utiliser la fonction NB qui prend comme argument un ensemble de 30 références sur des cellules ou une plage de cellules afin de retourner le nombre de cellules contenant des nombres.

Il suffit alors de l'appliquer à la première colonne contenant le numéro du lot afin de compter le nombre de lots. On écrit alors

« =NB(A2:A154) »

### 2. L'algorithme correspondant à la fonction TantiemesTypeProp (nomPropriétaire : chaîne, typeLot : chaîne): decimal

L'algorithme demandé correspond à celui de la fonction TantiemesTypeProp :

- qui prend en paramètres :
  - une chaîne de caractères appelée nomPropriétaire
  - une chaîne de caractères appelée typeLot
- qui renvoie comme résultat un entier qu'on choisit d'appeler cumulTantièmes.

Le principe de fonctionnement de cet algorithme est le suivant :

- on crée et initialise cumulTantièmes à 0 et numLig (le numéro de ligne dans la feuille de calcul) à 2
- on utilise une boucle « tant que » dont la condition de continuation est « la cellule de coordonnées (numLig,1) n'est pas vide », ce qui se traduit par :

ValeurCellule (numLig,1) != ""

- à l'intérieur de la boucle est on fait le test suivant :

« Si le contenu de la cellule de coordonnées (numLig,8) correspond au nom du propriétaire ET si le contenu de la cellule de coordonnées (numLig,3) correspond au type de lot alors ajouter dans tantiemesTypeProp la valeur contenu dans la cellule de coordonnées (numLig,6) »

Ce qui se traduit par :

```
SI ( (ValeurCellule (numLig,8) = nomPropriétaire) ET
      (ValeurCellule (numLig,3) = typeLot) )
    ALORS tantiemesTypeProp ← tantiemesTypeProp +
                                ValeurCellule (numLig,6)
FSI
```

Il ne faut naturellement pas oublier d'augmenter la valeur de numLig de 1 pour passer à la ligne suivante (sous peine de rester bloquer à la liste 2 et ainsi se trouver dans une boucle infinie).

- on termine l'algorithme en retournant la valeur de cumulTantièmes

RETOURNER cumulTantièmes

L'algorithme s'écrit alors de la manière suivante :

```
FONCTION TANTIEMESTYPEPROP (nomPropriétaire : chaîne de caractères,  
                             typeLot : chaînes de caractères) : entier  
  
VAR  
  
    tantiemesTypeProp : entier  
    numLig : entier  
  
DEBUT  
  
    tantiemesTypeProp ← 0  
    numLig ← 2  
  
    TANT QUE (ValeurCellule (numLig,1) != "")  
  
        SI ( (ValeurCellule (numLig,8) = nomPropriétaire) ET  
            (ValeurCellule (numLig,3) = typeLot) )  
  
            ALORS tantiemesTypeProp ← tantiemesTypeProp +  
                ValeurCellule (numLig,6)  
  
        FSI  
  
        numLig ← numLig + 1  
  
    FTQ  
  
    RETOURNER cumulTantièmes  
  
FIN
```