

MINI PROJET - SUJET N°2**CORRIGÉ TYPE**

(avec reprise de ce qui est correct dans les rapports remis)

I - CONTEXTE**I-1. L'EXISTANT**

Le domaine considéré est la Gestion des locations dans l'entreprise HIFILOC.

HIFILOC possède plusieurs agences qui offrent aux clients un service de location de matériels audiovisuels (téléviseurs, amplis, haut-parleurs).

Pour la gestion des locations, l'entreprise exploite une base de données support d'une petite application web permettant aux agences et aux clients de consulter en ligne l'offre de l'entreprise, ainsi que leur contrat de location en cours.

Le SCHEMA RELATIONNEL est le suivant :

VILLE (CP, VILL)

CLIENT (NUMCLI, NOMCLI, PRNCLI, ADRCLI, CP#)

AGENCE (CODAGC, NOMAGC, CP#)

MATERIEL (REFMAT, DESMAT, PRILOC)

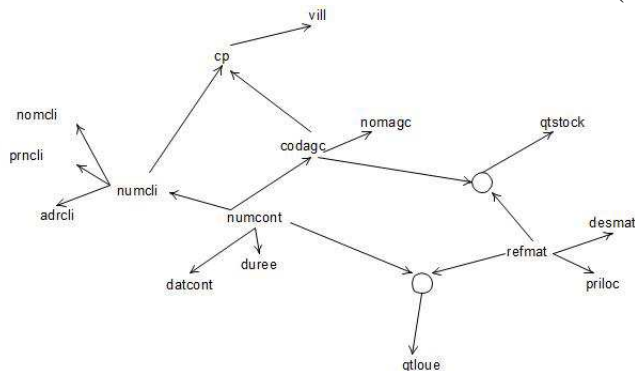
CONTRAT (NUMCONT, DATCONT, DUREE, NUMCLI#, CODAGC#)

DETAIL_CONTRAT (NUMCONT#, REFMAT#, QTELOUE)

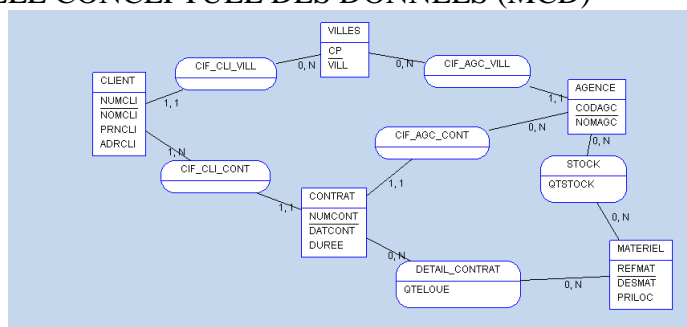
STOCK (CODAGC#, REFMAT#, QTSTOCK)

Il résulte des éléments de conception suivants :

GRAPHE DES DEPENDANCES FONCTIONNELLES (GDF)



MODELE CONCEPTUEL DES DONNEES (MCD)

**I-2. L'ÉVOLUTION A RÉALISER**

Le projet doit permettre de faire évoluer la base de données pour intégrer l'évolution décidée par l'entreprise HIFILOC, et dont l'objectif stratégique est d'intégrer la gestion des fournisseurs dans le domaine.

Présentation de l'évolution

La direction de HIFILOC a décidé de *confier aux agences la relation avec les fournisseurs de matériels*.

Désormais chaque fournisseur sera identifié de façon unique par un numéro sous lequel seront enregistrées les informations suivantes : raison sociale, adresse, code postal, ville, téléphone, mail.

Un ou plusieurs contacts seront identifiés chez un fournisseur donné. Chaque contact est identifié par un numéro unique renseignant son nom, son prénom, son téléphone direct, son mail direct ; sa fonction, exemple : commercial, gestionnaire.

Chaque agence doit intégrer le fait qu'une référence de matériel donnée est délivré par un fournisseur unique ; un fournisseur donné peut approvisionner l'entreprise pour plusieurs références de matériel distinctes.

Un fournisseur n'est enregistré qu'à partir du moment où il délivre au moins une référence de matériel à l'entreprise.

Une agence pourra privilégier les relations avec les fournisseurs habitant la même ville qu'elle.

Le projet se déroulera en trois étapes : 1) modélisation conceptuelle de l'évolution. 2) migration de la base de données existante vers la nouvelle version conçue. 3) réalisation de tests.

II – Modélisation Conceptuelle de l'évolution

1) Mise à jour du GDF

De la nouvelle disposition suivante :

chaque fournisseur sera identifié de façon unique par un numéro sous lequel seront enregistrées les informations suivantes : raison sociale, adresse, code postal, ville, téléphone, mail

on déduit la création d'un groupe de Dépendances Fonctionnelles (DF) ayant toutes pour source une nouvelle propriété qui identifie un fournisseur et que nous dénommons "NUMFOURN", et pour but, respectivement les propriétés nouvelles caractérisant un fournisseur : raison sociale que l'on nomme "RSFOURN", adresse que l'on nomme "ADRFOURN", téléphone que l'on nomme "TELFURN", mail que l'on nomme "MAILFOURN".

De l'information suivante :

Une agence pourra privilégier les relations avec les fournisseurs habitant la même ville qu'elle

on déduit que les villes de localisation des fournisseurs ont une intersection non nulle avec l'ensemble des villes de localisation des agences ; on crée donc pas de nouvelles propriétés pour caractériser le code postal et la ville d'un fournisseur ; on utilise les propriétés existantes "CP" et "VILL", avec la DF NUMFOURN → CP qui traduit le lien sémantique entre l'identifiant d'un fournisseur et le code postal de sa ville de résidence ; la connaissance du nom de la ville de résidence étant obtenue par transitivité puisque CP → VILL

De la nouvelle disposition suivante :

Un ou plusieurs contacts seront identifiés chez un fournisseur donné. Chaque contact est identifié par un numéro unique renseignant son nom, son prénom, son téléphone direct, son mail direct ; sa fonction, exemple : commercial, gestionnaire

on déduit la création d'un groupe de Dépendances Fonctionnelles (DF) ayant toutes pour source une nouvelle propriété qui identifie un contact chez un fournisseur et que nous dénommons "NUMCONTAC", et pour but, respectivement les propriétés nouvelles caractérisant un contact chez un fournisseur : nom que l'on nomme "NOMCONTAC", prénom que l'on nomme "PRENCONTAC", téléphone que l'on nomme "TELCONTAC", mail que l'on nomme "MAILCONTAC", fonction que l'on nomme "FONCTION".

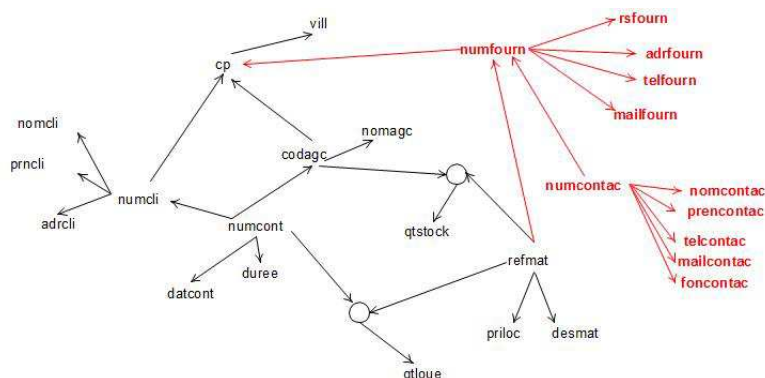
La DF NUMCONTAC → NUMFOURN traduira le lien sémantique entre un contact et le fournisseur chez lequel il exerce.

De la nouvelle disposition suivante :

Chaque agence doit intégrer le fait qu'une référence de matériel donnée est délivré par un fournisseur unique ; un fournisseur donné peut approvisionner l'entreprise pour plusieurs références de matériel distinctes

on déduit la DF REFMAT → NUMFOURN qui traduit le lien sémantique entre une référence de matériel et le fournisseur en charge de son approvisionnement.

D'où le nouveau GDF :



2) Mise à jour du MCD

L'ajout du groupe de DF qui a pour source NUMFOURN se traduit dans le MCD par la création d'une entité FOURNISSEUR qui a pour identifiant NUMFOURN et pour autres propriétés les buts dans le groupe de DF

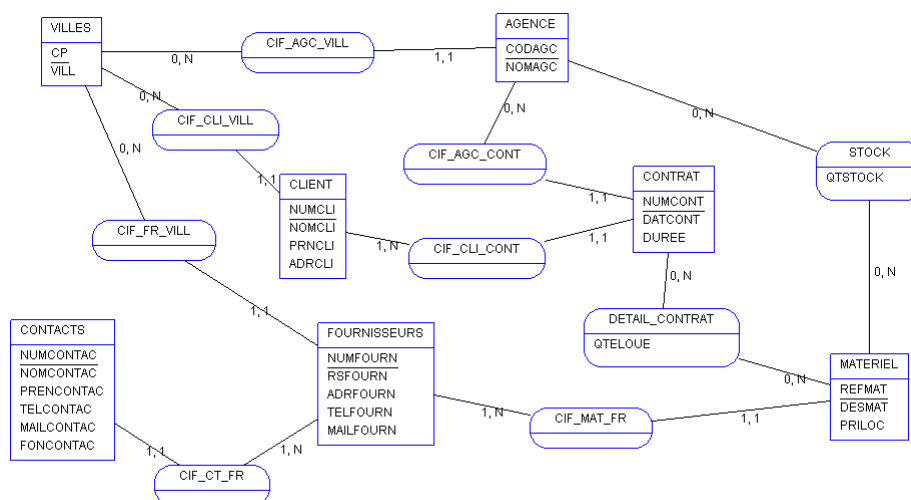
La DF NUMFOURN → CP se traduit par l'ajout d'une association de type CIF entre l'entité FOURNISSEUR et l'entité VILLES

L'ajout du groupe de DF qui a pour source NUMCONTAC se traduit dans le MCD par la création d'une entité CONTACTS qui a pour identifiant NUMCONTAC et pour autres propriétés les buts dans le groupe de DF

La DF NUMCONTAC → NUMFOURN se traduit par l'ajout d'une association de type CIF entre l'entité CONTACTS et l'entité FOURNISSEUR

La DF REFMAT → NUMFOURN se traduit par l'ajout d'une association de type CIF entre l'entité MATERIEL et l'entité FOURNISSEUR

D'où le nouveau MCD :



3) Mise à jour du schéma relationnel

Les entités FOURNISSEURS et CONTACTS sont traduites par deux relations de même nom qui ont pour clé primaire l'identifiant, et pour autres attributs les autres propriétés des entités correspondantes :

FOURNISSEURS (NUMFOURN, RSFOURN, ADRFOURN, TELFOURN, MAILFOURN)

CONTACTS (NUMCONTAC, NOMCONTAC, PRENCONTAC, TELCONTAC, MAILCONTAC, FONCONTAC)

L'association CIF entre FOURNISSEURS et VILLES se traduit par une clé étrangère dans la table FOURNISSEURS.

FOURNISSEURS (NUMFOURN, RSFOURN, ADRFOURN, TELFOURN, MAILFOURN, CP#)

De même, l'association CIF entre FOURNISSEURS et CONTACTS se traduit par une clé étrangère dans la table CONTACTS.

CONTACTS (NUMCONTAC, NOMCONTAC, PRENCONTAC, TELCONTAC, MAILCONTAC, FONCONTAC, NUMFOURN#)

Traduction identique pour l'association CIF entre FOURNISSEURS et MATERIEL se traduit par une clé étrangère dans la table MATERIEL.

D'où le nouveau schéma relationnel :

VILLE (CP, VILL)

CLIENT (NUMCLI, NOMCLI, PRNCLI, ADRCLI, CP#)

AGENCE (CODAGC, NOMAGC, CP#)

MATERIEL (REFMAT, DESMAT, PRILOC, QTSTOCK, NUMFOURN#)

CONTRAT (NUMCONT, DATCONT, DUREE, NUMCLI#, CODAGC#)

DETAIL_CONTRAT (NUMCONT#, REFMAT#, QTELOUE)

FOURNISSEURS (NUMFOURN, RSFOURN, ADRFOURN, TELFOURN, MAILFOURN, CP#)

CONTACTS (NUMCONTAC, NOMCONTAC, PRENCONTAC, TELCONTAC, MAILCONTAC, FONCONTAC, NUMFOURN#)

Les modifications apportées ne changent rien à la conformité du schéma relationnel à la 3^{ème} forme normale.

III – Migration de la Base de Données

Mise à jour de la structure de la base de données

Deux nouvelles tables doivent être créées, FOURNISSEURS en premier lieu, puis CONTACTS qui référence la précédente ; ce que l'interface graphique de phpmyadmin permet de réaliser très simplement.

Table FOURNISSEURS :

Serveur: localhost Base de données: SGC2018_P2LOCMAT Table: FOURNISSEURS "InnoDB free: 16384 kB; ('CP') REFER 'SGC2018_P2LOCMAT.FOURNISSEURS' ('CP')"

Afficher Structure SQL Rechercher Insérer Exporter Importer Opérations Vider Supprimer

Champ	Type	Interclassement	Attributs	Null	Défaut	Extra	Action
NUMFOURN	int(11)			Non			
RSFOURN	varchar(30)	latin1_swedish_ci		Non			
TELFURN	bigint(20)			Non			
MAILFOURN	varchar(50)	latin1_swedish_ci		Non			
CP	int(11)			Non			

Tout cocher / Tout décocher Pour la sélection :

Version imprimable Gestion des relations Suggérer des optimisations quant à la structure de la table

Ajouter 1 champ(s) En fin de table En début de table Après NUMFOURN Exécuter

Index				Espace utilisé		Statistiques	
Nom de l'index	Type	Cardinalité	Action	Champ	Type	Espace	Valeur
PRIMARY	PRIMARY	0		NUMFOURN	Données	16 384 0	format
FK_FOURNISSEURS_CP	INDEX	0		CP	Index	16 384 0	Interclassement
Créer un index sur 1 colonne(s) Exécuter					Total	32 768 0	Création

Information: Ven 30 Novembre 20

Table CONTACTS :

Serveur: localhost Base de données: SGC2018_P2LOCMAT Table: CONTACTS "InnoDB free: 16384 kB; ('NUMFOURN') REFER 'SGC2018_P2LOCMAT.FOURNISSEURS' ('NUMFOURN')"

Afficher Structure SQL Rechercher Insérer Exporter Importer Opérations Vider Supprimer

Champ	Type	Interclassement	Attributs	Null	Défaut	Extra	Action
NUMCONTAC	int(11)			Non			
NOMCONTAC	varchar(50)	latin1_swedish_ci		Non			
PRENCONTAC	varchar(50)	latin1_swedish_ci		Non			
TELCONTAC	varchar(10)	latin1_swedish_ci		Non			
MAILCONTAC	varchar(70)	latin1_swedish_ci		Non			
FONCTION	varchar(30)	latin1_swedish_ci		Non			
NUMFOURN	int(11)			Non			

Tout cocher / Tout décocher Pour la sélection :

Version imprimable Gestion des relations Suggérer des optimisations quant à la structure de la table

Ajouter 1 champ(s) En fin de table En début de table Après NUMCONTAC Exécuter

Index				Espace utilisé		Statistiques	
Nom de l'index	Type	Cardinalité	Action	Champ	Type	Espace	Valeur
PRIMARY	PRIMARY	0		NUMCONTAC	Données	16 384 0	format
FK_CONTACTS_NUMFOURN	INDEX	0		NUMFOURN	Index	16 384 0	Interclassement
Créer un index sur 1 colonne(s) Exécuter					Total	32 768 0	Création

Information: Ven 30 Novembre 20

Il faut également mettre à jour le schéma de la table MATERIEL pour y ajouter l'attribut NUMFOURN, qui doit être définie comme clé étrangère référençant FOURNISSEUR

Mise à jour des données des tables

Consigne à respecter pour FOURNISSEUR :

Tous les fournisseurs de l'entreprise doivent être intégrés à la base. La Direction fournit pour ce faire, un fichier texte, au format "csv" : (numfourniss ; raisonsocial ; teleph ; mail ; codpost ; ville).

Il faut donc d'abord trouver un moyen d'exploiter les données de ce fichier csv. La solution la plus immédiate est de créer une table temporaire dont le schéma est identique au format du csv, soit temp_fourniss (numfourn , rsfourn , telfourn , mailfourn , cpfourn , villfourn). On importe alors le fichier csv dans la table temp_fourniss.

En scrutant ce fichier, on note qu'il comporte des villes qui n'existent pas dans la table VILLES. Avant d'insérer les données dans la table FOURNISSEURS, il faut d'abord rajouter dans VILLES ces nouvelles villes.

La solution choisie doit éviter la re saisie pour limiter le risque d'erreur. On utilise pour cela une jointure avec une condition exprimant l'inégalité :

```
select cpfourn, villfourn
from temp_fourniss t join VILLES V on t.cpfourn <> V.CP;
```

Le résultat :

cpfourn	villfourn
98120	CAP NORD
98170	DES CLOS

Ce résultat peut être enregistré dans une table (possibilité offerte par phpmyadmin) ; appelons cette table temp_vill.

On peut maintenant utiliser cette table temporaire dans un ordre SQL "insert", pour insérer ses données dans la table VILLES.

```
Insert into VILLES
select *
from temp_vill;
```

On peut maintenant utiliser temp_fourniss dans un ordre SQL "insert", pour insérer dans la table FOURNISSEURS les données correspondantes (tous les attributs sauf ADRFOURN qui n'est pas spécifiée).

```
Insert into FOURNISSEURS (NUMFOURN,RSFOURN,TELFOURN,MAILFOURN,CP)
select numfourn,rsfourn,telfourn,mailfourn,cpfourn
from temp_fourniss;
```

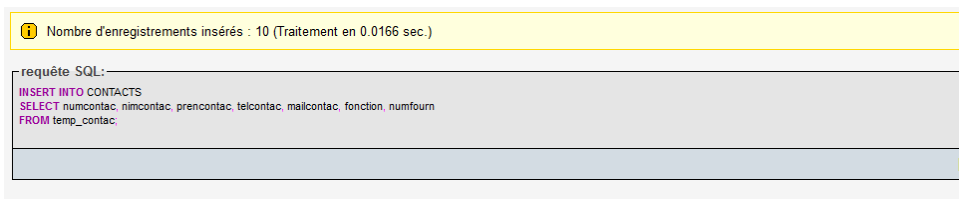
La mise à jour des données de la table FOURNISSEURS étant faite, celle des données de la table CONTACTS peut maintenant être réalisée dans le respect de la consigne donnée, à savoir :

Tous les contacts des différents fournisseurs de l'entreprise doivent être intégrés à la base. La Direction fournit pour ce faire, un fichier texte, au format "csv" : (numfourniss ; numcontact ; nomcontact ; precontact ; telephcontact ; mailcontact ; fonction).

On applique le même principe : création d'une table temporaire

temp_contac(numfourniss , numcontact , nomcontact , precontact , telephcontact , mailcontact , fonction) ; et import dans cette table des données csv.

L'ordre SQL "insert" est ensuite utilisé pour insérer les données de la table temporaire dans la table CONTACTS.



Il reste à mettre à jour la table MATERIEL dans le respect de la consigne donnée :

Toutes les références de matériel présentes dans la base doivent être associées à leur fournisseur. La Direction fournit pour ce faire, un fichier texte, au format "csv" : (refmat ; numfourniss).

Sur le même principe que précédemment, on crée une table temporaire

temp_mat_fourn(refmat, numfourniss) ; puis on importe dans cette table les données csv.

L'ordre SQL de mise à jour, "update", peut alors être utilisé pour valoriser l'attribut NUMFOURN de MATERIEL à partir de temp_mat_fourn.

i Nombre d'enregistrements affectés : 20 (Traitement en 0.0574 sec.)

requête SQL:

```
UPDATE MATERIEL SET NUMFOURN = ( SELECT numfourn
FROM temp_mat_fourn
WHERE refmat = MATERIEL REFMAT )
```

Suppression de tables

Il faut ensuite supprimer les tables temporaires.

IV – LES TESTS

Avant la mise en exploitation de la nouvelle version, la Direction demande à avoir le résultat de requêtes de test.

Réponses aux requêtes de test

R2.1. Liste triée par ville de résidence, puis par ordre alphabétique des raisons sociales des fournisseurs. Format [cp | vill | numfourniss | raisonsocial].

i Affichage des enregistrements 0 - 7 (8 total, Traitement en 0.0004 sec.)

requête SQL:

```
SELECT F.CP, VILL, NUMFOURN, RSFOURN
FROM FOURNISSEURS F
JOIN VILLES V ON F.CP = V.CP
ORDER BY VILL, RSFOURN;
LIMIT 0, 30
```

[Modifier] [Expliquer SQL] [Créer source]

Afficher : 30 enregistrement(s) à partir de l'enregistrement n° 0
en mode horizontal et répéter les en-têtes à chaque groupe de 100

CP	VILL	NUMFOURN	RSFOURN
98100	BASSE-MER	60	CTV
98100	BASSE-MER	10	LCLX
98120	CAP NORD	50	GEM
98170	DES CLOS	80	SONDIS
98150	LE GOSSARD	20	DURTY
98150	LE GOSSARD	70	HFSTK
98110	POINTE-PETER	40	HIGHSOUND
98110	POINTE-PETER	30	TVLEAD

Afficher : 30 enregistrement(s) à partir de l'enregistrement n° 0

R2.2. Répartition de la valeur du stock global (i.e toutes agences confondues) par fournisseur. Format [numfourniss | raisonsocial | val_stock].

i Affichage des enregistrements 0 - 7 (8 total, Traitement en 0.0005 sec.)

requête SQL:

```
SELECT F.NUMFOURN, RSFOURN, SUM(QTSTOCK * PRILOC)
FROM STOCKS
JOIN MATERIEL M ON M.REFMAT = S.REFMAT
JOIN FOURNISSEURS F ON F.NUMFOURN = M.NUMFOURN
GROUP BY F.NUMFOURN;
LIMIT 0, 30
```

[Modifier] [Expliquer SQL] [Créer source]

Afficher : 30 enregistrement(s) à partir de l'enregistrement n° 0
en mode horizontal et répéter les en-têtes à chaque groupe de 100

NUMFOURN	RSFOURN	sum(QTSTOCK * PRILOC)
10	LCLX	7160
20	DURTY	11825
30	TVLEAD	11540
40	HIGHSOUND	10995
50	GEM	11185
60	CTV	9275
70	HFSTK	12150
80	SONDIS	2975

Afficher : 30 enregistrement(s) à partir de l'enregistrement n° 0

NOTE : dans l'image d'écran ci-dessus, il manque le renommage de la colonne "sum(QTSTOCK * PRILOC) en "ValStock".

R2.3. Nombre de fournisseurs par ville. Format [cp | vill | nb_fourniss].

```
SELECT F.CP, VILL, COUNT(NUMFOURN) AS 'nb_fourniss'  
FROM FOURNISSEUR F JOIN VILLE V ON F.CP=V.CP  
GROUP BY F.CP, VILL;
```

CP	VILL	nb_fourniss
98100	BASSE-MER	2
98120	CAP NORD	1
98170	DES CLOS	1
98150	LE GOSSARD	2
98110	POINTE-PETER	2