



Travaux Pratiques

COBOL

10/12/2020

ENONCES

Etudes de cas

(DSN: APIIn.COB.ASSURES)

(Pour chaque question on développera un programme)

Une compagnie d'assurances désire réaliser une chaîne batch pour traiter le fichier des assurés (description en annexe).

On demande de créer les programmes suivants :

1. Ecrire un programme contenant la description du fichier ASSURES.
2. Ecrire un programme qui va lire le fichier ASSURES et lister son contenu
3. Ecrire un programme qui va lister le fichier des assurés par rubrique avec un DISPLAY pour chaque zone.
4. Ecrire un programme qui va imprimer sur un fichier d'édition la quittance suivant le modèle fourni en annexe en ajoutant le jour, date du jour et heure et en supprimant les zéros non significatifs des montants affichés.

5. Ecrire un programme qui doit valider le fichier de **mouvements**.

Ne faire que le contrôle de ce fichier en tenant compte du dessin et des contrôles fournis en annexe. (DSN: APIIn.COB.MVT)

6. Maintenance du programme précédent en utilisant le fichier (Cf. P10):

DSN: APIIn.COB.MVTMAJ (Cf. Annexe)

7. Sortir les statistiques sur le fichier des assurés par département et par code type (Cf. P11).

TP COBOL

Description du fichier des assurés : **DSN: APIIn.COB.ASSURES**

Longueur (LRECL) 80

Organisation séquentielle QSAM

Matricule	6	Numériques
Nom Prénom	20	Alphanumériques
Rue Adresse	18	Alphanumériques
Code Postal	5	Numériques
Ville	12	Alphanumériques
Type Véhicule	1	Alphanumérique
Prime de base	4,2	Numériques
Bonus Malus (B.M.)	1	Alphanumérique
Taux	2	Numériques

N-B. Vous pouvez visualiser ce fichier avec l'option 1 de PDF (browse).

Modèle de quittance

QUITTANCE DE PRIME			
***	Nom		***
***	Adresse		***
***	Code postal/Ville		***
Prime de base	xxxxx,xx		
(Majoration)	xxxx,xx	(Malus	nn%
ou Dégrèvement)		Bonus)	
<hr/>			
Total à payer	xxxxx,xx		

Fichier des mouvements

DSN: APIIn.COB.MVT

Longueur 80 Organisation séquentielle QSAM

Matricule 6 N (Numérique)

Code mouvement 1 N

Suite de l'enregistrement en fonction du code :

Code = 1

Signe (+ ou -) 1 AN (Alphanumérique)

Taux 2 N

Code = 2

Nom Prénom 20 AN

Code = 3

Adresse 18 AN

Code postal 5 N

Ville 12 AN

Code = 4

Type 1 AN

Prime 4,2 N

Contrôle des mouvements

Matricule et code mouvement sont obligatoires

Code = 1

Signe doit être + ou -

Taux doit être numérique

Code = 2

Nom doit être différent de blanc

Code = 3

Au moins une des trois zones doit être saisie

Le code postal doit être numérique s'il est saisi

Code = 4

Contrôle en table de type de prime

Le type de prime peut prendre les valeurs 0 à 9.

La prime doit être numérique

Messages d'Erreurs

1. Matricule non numérique
2. Code mouvement inconnu
3. Signe différent de + ou -
4. Taux non numérique
5. Le nom n'est pas renseigné
6. Au moins une zone doit être saisie (adresse, code postal, ville)
7. Code postal invalide
8. Type de prime invalide
9. Prime non numérique

TP COBOL

APInn

jjjjjjjjjj le JJ/MM/AAAA

HH:MM:SS

LISTE DES ERREURS

MATRICULE	TYPE	MESSAGE D'ERREUR
-----	----	-----
AAAAAA	1	01 - Matricule non numerique
111111	C	02 - Code mouvement invalide
222222	1	03 - Signe différent de + ou -
333333	1	04 - Taux non numérique
444444	2	05 - Le nom n'est pas renseigné
555555	3	06 - Au moins une zone doit être saisie
666666	3	07 - Code postal invalide
777777	4	08 - Type de prime invalide
888888	4	09 - Prime non numérique
.....	.	.. -
.....	.	.. -
.....	.	.. -

TP COBOL

APInn

jjjjjjjjjj le JJ/MM/AAAA

HH:MM:SS

Statistique sur contrôle du fichier mouvement

10 - Nombre d'enregistrements lu sur FMVT : nnnnn
11 - Nombre d'enregistrements en erreur : nnnnn
 12 - Nombre d'erreur de type 1 : nnnnn
 13 - Nombre d'erreur de type 2 : nnnnn
 14 - Nombre d'erreur de type 3 : nnnnn
 15 - Nombre d'erreur de type 4 : nnnnn
 16 - Nombre d'erreur mat invalide : nnnnn
 17 - Nombre d'erreur code invalide : nnnnn

Maintenance du programme de contrôle des mouvements

(Utiliser le fichier **APIIn.COB.MVTMAJ**)

Dans le fichier Mouvement a été rajouté un Code 5 dont le format est le suivant

Code = 5

Matricule	6	N	(Numérique)
Code mouvement	1	N	
Nom	12	A	
Prénom	10	A	
Tél fixe	10	N	
Tél port	10	N	
Salaire	4	N packé	dont 2 décimales
Qualification	19	AN	
Date d'embauche	8	N	(JJMMAAAA)

Contrôles à effectuer :

1. Le numéro de téléphone fixe doit commencer par : 01,02,03,04,05,06 ou 07
2. Le numéro de téléphone port doit commencer par : 06 ou 07
3. Le salaire doit être numérique et afficher en sysout (matricule nom et salaire)
4. Vérifier la date d'embauche (JJ compris entre 01 et 31; MM entre 01 et 12)

Statistique par département

DEPARTEMENT : xx

TYPE : x

NOM	PRENOM	MONTANT
xxx	xxx	xxxx,xxx
.....

TOTAL DU TYPE	:	x	xxxxx,xx

TOTAL DEPARTEMENT	:	xx	xxxxxx,xx

TOTAL GENERAL			xxxxx,xx
---------------	--	--	----------

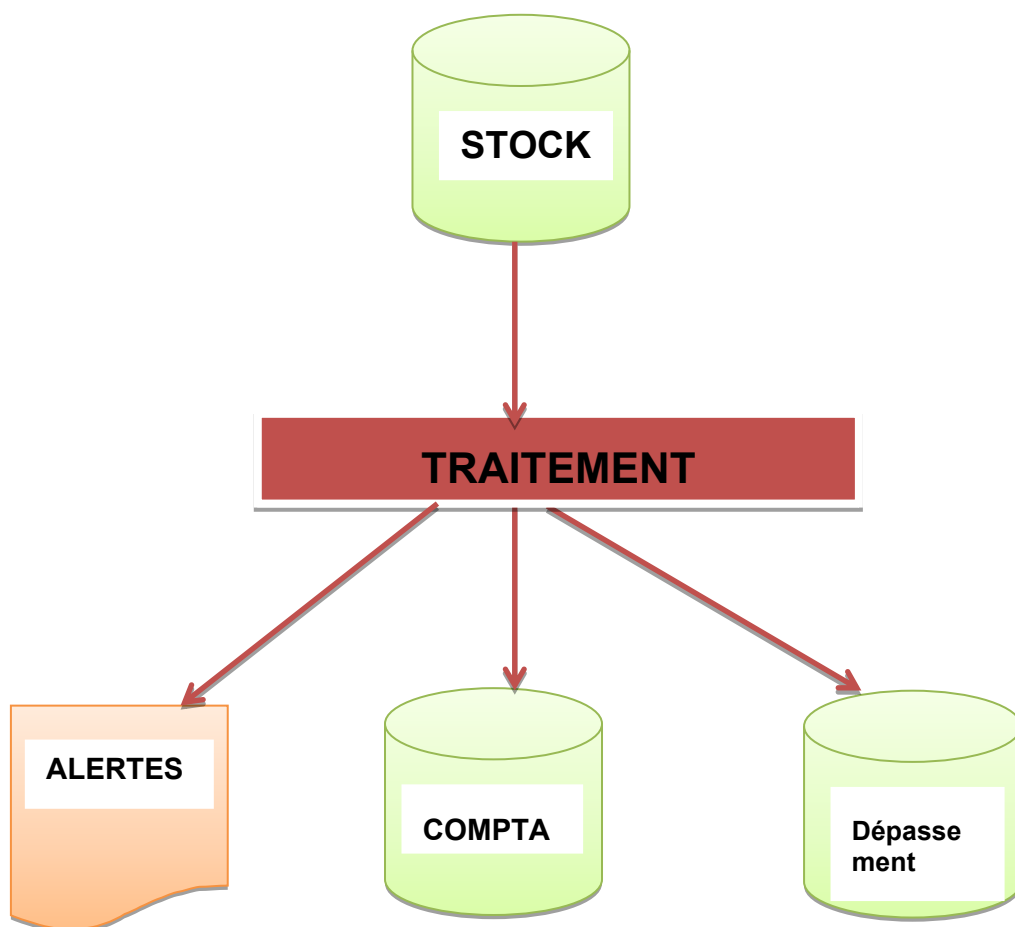
Statistique sur les programmes

Développer un programme permettant de sortir des statistiques sur vos programme en indiquant :

- *Nombre de lignes du programme* : *nnnnn*
- *Nombre de ligne en commentaires* : *nnnnn*
- *Nombre de ligne vides* : *nnnnn*
- *Pourcentages de commentaires* : *nn%*
- *Pourcentage de lignes blanches* : *nn%*

ALGO n°1

GESTION DE STOCK



STRUCTURE DES FICHIERS

Fichier STOCK: **APIIn.COB.FSTOCK**

- *numéro du produit*
- *nombre de produit en stock*
- *prix unitaire*
- *nombre minimum de produits avant alerte*

Fichier COMPTA et DEPASSEMENT:

- *numéro de produit*
- *prix total = prix unitaire * nombre de produit*

TRAITEMENT

Soit un fichier Stock trié par ordre croissant sur le numéro de produit.

Ce fichier va servir à alimenter un fichier comptable, un fichier de dépassement et à imprimer des messages d'alerte.

1) Pour chaque enregistrement du fichier Stock :

créer un enregistrement dans le fichier de sortie "Compta" comptabilisant chaque article

2) Si le nombre de produits est inférieur au nombre de produits minimum :

Imprimer une ligne d'alerte ("Réapprovisionnement du produit N° xxxxXX") par DISPLAY

3) Si le nombre de produit est supérieur ou égal au nombre minimum et que le prix DU PRODUIT est supérieur à 150 € :

créer un enregistrement dans le fichier de sortie "dépassement" avec numéro de produit et prix total

4) Sortir un fichier de statistique sur le :

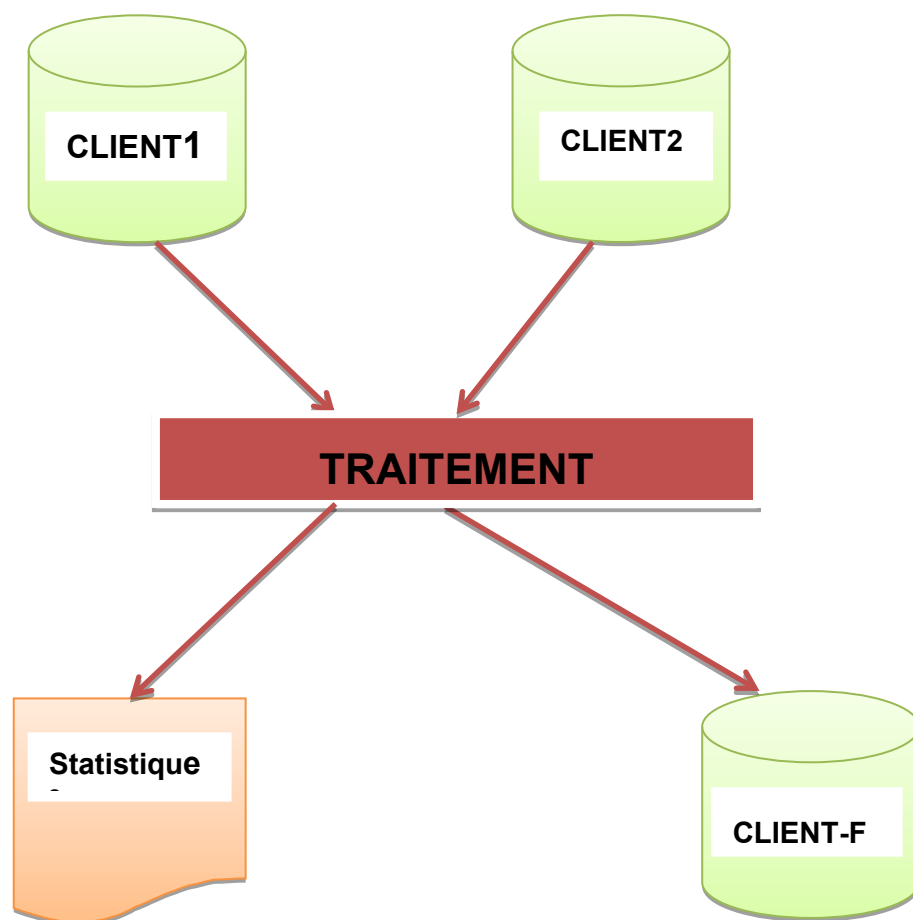
- Nombre d'enregistrement lus dans fichier STOCK*
- Nombre d'enregistrement écrits dans fichier COMPTA*
- Nombre d'enregistrement écrits dans fichier DEPASSEMENT*
- Nombre d'enregistrement écrits dans fichier Anomalie*

ALGO n°3

FUSION DE FICHIERS

APIIn.COB.FALG3C1A

APIIn.COB.FALG3C2A



TRAITEMENT

1. Soit 2 fichiers clients (CLIENT1 et CLIENT2) triés par ordre croissant sur le numéro de client et comportent un seul enregistrement par client.

Créer un fichier CLIENT-F contenant la fusion des 2 fichiers en entrée.

Afficher les statistiques indiquant :

- le nombre d'enregistrements lus dans CLIENT1
- le nombre d'enregistrements lus dans CLIENT2
- le nombre d'enregistrements lus dans CLIENT-F

2. Refaire le même traitement en utilisant les fichiers :

APIIn.COB.FALG3C1B et **APIIn.COB.FALG3C2B**

Le traitement devra être arrêté si un client est en double dans CLIENT1 ou CLIENT2 ou si il est identique dans dans CLIENT1 ou CLIENT2.

Créer un fichier de statistique indiquant :

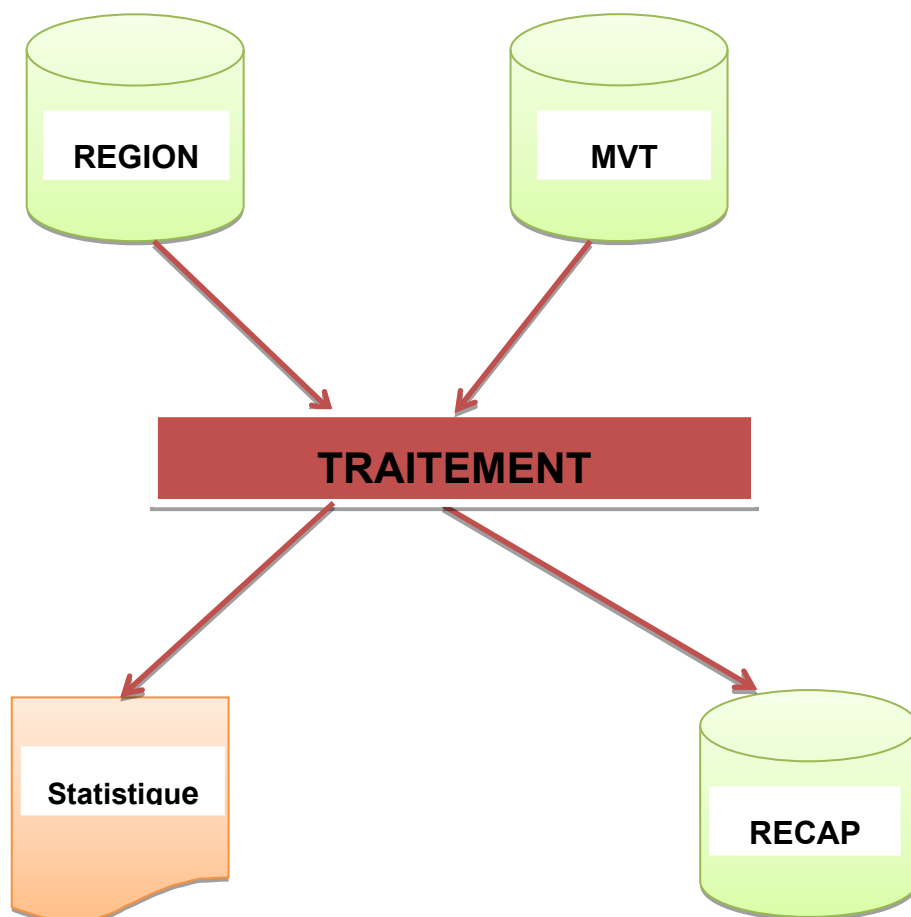
- le nombre d'enregistrements lus dans CLIENT1
- le nombre d'enregistrements lus dans CLIENT2
- le nombre d'enregistrements lus dans CLIENT-F

ALGO n°4

ASSORTIMENT DE FICHIERS

Trié par Région

Trié par Rg, Ag, Vend



STRUCTURE DES FICHIERS

Fichier REGIONS:

- Numéro (numéroté à partir de 100 avec pas de 1)
- Code enregistrement (RG)
- Numéro de région
- Nom de la région

Fichier MVT :

- Numéro (numéroté à partir de 100 avec pas de 1)
- Code enregistrement (= MV)
- Région
- Code agence
- Code vendeur
- Nombre de dossiers gérés

Fichier RECAP :

- **Entête enregistrement :**

1. Immatriculation (à partir de 1, incrémenté de 1)
2. Code enregistrement (= 400, 500, 999)
3. Région

- **Corps enregistrement de type 400**

1. Code agence
2. Nombre de dossiers

- **Corps enregistrement de type 500**

1. Code agence (=00)
2. Nom de la région
3. Nombre de dossiers

Pas de corps enregistrement de type 999

TRAITEMENT

Soit 2 fichiers :

1. un fichier *REGIONS* (1 Engt/Region)
2. un fichier *MVT.*(1 Engt/vendeur)

Un vendeur fait partie d'une agence qui fait elle-même fait partie d'une région

On veut fournir en sortie un fichier récapitulatif comprenant :

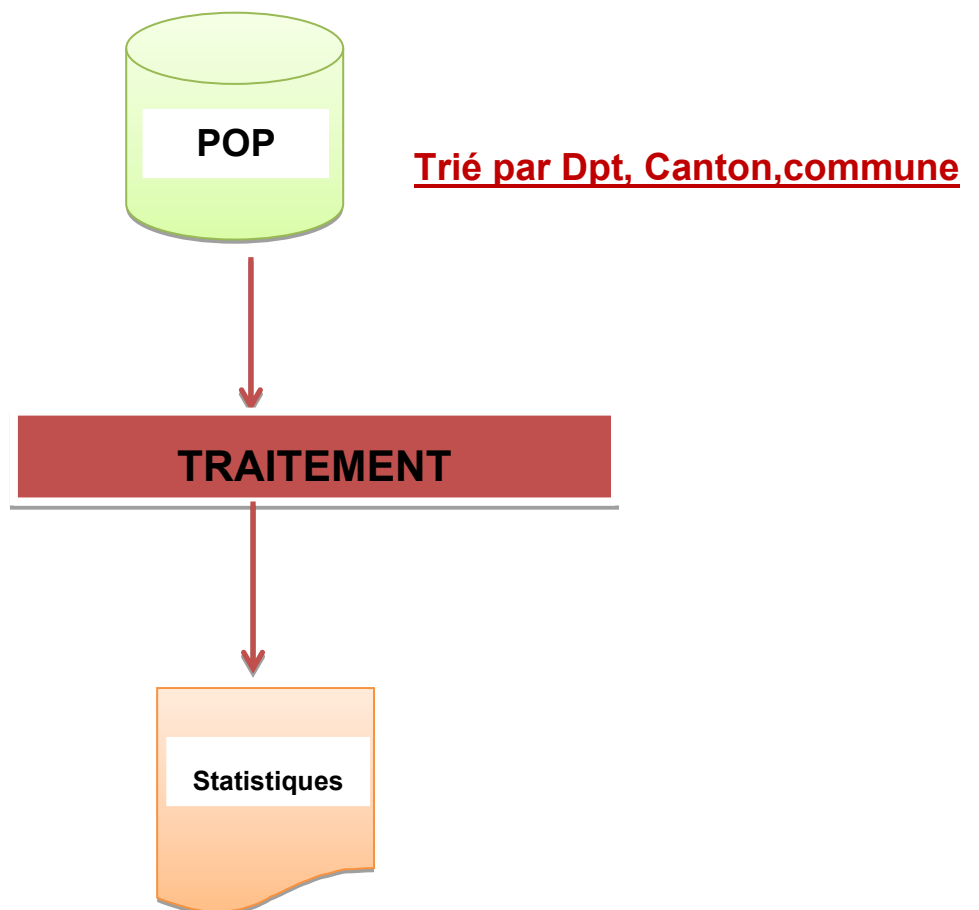
- *Un enregistrement Récap par agence (code 400) contenant le nombre de dossiers géré par l'agence.*
- *Un enregistrement Récap par région (code 500), contenant le nombre de dossiers géré de la région.*
- *Un enregistrement de type Erreur (code 999) lorsqu'une région référencée dans le fichier MVT n'existe pas dans le fichier des régions.*
- *Lors du traitement on vérifiera que les enregistrements de chaque fichier en entrée ont bien un numéro d'immatriculation commençant par 100 et croissant de 1 en 1.*
- *On vérifiera également qu'ils sont bien triés.*

Dans le cas contraire, on arrête le traitement à la première anomalie rencontrée

ALGO n°5

STATISTIQUES POPULATION

DSN : APIIn.COB.POP



STRUCTURE DES FICHIERS

Fichier POPULATION: (APIIn.COB.POP)

- département
- canton
- commune
- Nombre d'habitants dans la commune

Un département est composé de cantons, eux même formés de communes

Fichier STATISTIQUES:

STATISTIQUE DE LA POPULATION

Département dd :

Canton aaa :

commune ccc nnnnnn

commune ccc nnnnnn

Total canton aaa nnnnnn

Canton aaa :

commune ccc nnnnnn

commune ccc nnnnnn

Total canton aaa nnnnnn

Total département dd nnnnnn

Département dd :

Canton aaa :

commune ccc nnnnnn

commune ccc nnnnnn

Total canton aaa nnnnnn

Total département dd nnnnnn

Total général nnnnnn

TRAITEMENT

On désire effectuer des statistiques de la répartition géographique des habitants en France.

On désire connaître le nombre d'habitants :

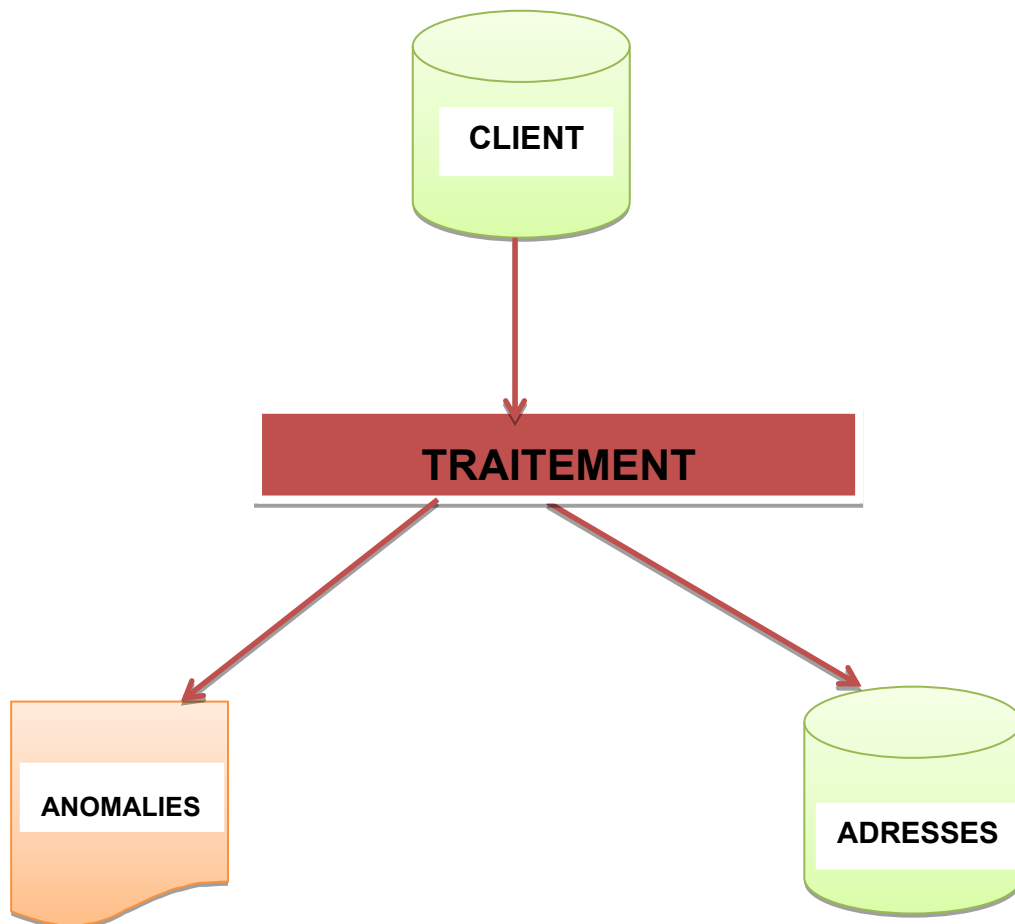
- *par commune*
- *puis par canton*
- *puis par département*
- *et pour la France entière*

La liste éditée indiquera le total par Commune, Canton, Département et pour le pays.

ALGO n°6

GESTION DES ADRESSES et ANOMALIES

API_n.COB.FALGO6



STRUCTURE DES FICHIERS

Fichier CLIENT:

4. Entête enregistrement :

- Type enregistrement (1, 2, 3)
- numéro de compte

5. Corps enregistrement de type 1

- nom client

6. Corps enregistrement de type 2

- Adresse
- Département

7. Corps enregistrement de type 3

- montant facture

Le fichier est trié par numéro de compte et type

Fichier ADRESSES:

- Code enregistrement (= 100)
- Numéro de compte
- Nom
- Adresse
- Département
- Montant total des factures (0 si pas de facture)

TRAITEMENT

L'objectif est de créer un fichier ADRESSE à du fichier CLIENT.

Gestion des anomalies :

Les enregistrements de type 1 et 2 sont obligatoires dans le fichier des mouvements.

Une liste contiendra les anomalies sur les mouvements de type 1 et 2 dans les cas suivants :

- *Type 1 absent.*
- *Type 1 en double.*
- *Type 2 absent.*
- *Type 2 en double.*

En cas d'anomalie, aucun enregistrement correspondant ne sera créé dans le fichier des adresses, mais le traitement se poursuivra avec les enregistrements suivants

Le fichier des adresses comportera un enregistrement par client.

.

ALGO n°7

CALCUL DES MOYENNES

DSN : API_n.COB.ETUDIANT

Soit une classe de n étudiants qui ont 10 notes à différents coefficients.

Calculer :

- 1. la moyenne de chaque étudiant,*
- 2. la moyenne générale de la classe*
- 3. la moyenne la plus élevée*
- 4. la moyenne la plus faible.*

Exercice AAAAMMJJ

Enoncé

Ecrire un programme de vérification de date répondant aux spécifications suivantes :

- demander à l'utilisateur d'entrer une date au format **AAAAMMJJ**
- analyser l'entrée et donner en retour une valeur numérique permettant d'afficher un message d'erreur ou le message " date valide si l'entrée est correcte."
- le programme bouclera sur lui-même pour analyser une autre date et s'arrêtera quand l'utilisateur entrera " 000000 " au clavier.

Consignes spéciales

- on demande un seul message d'erreur par date analysée
- on récupérera la date par un ACCEPT (FROM SYSIN) de longueur 6.

Buts de l'exercice

- utilisation des structures de contrôle (PERFORM, EVALUATE, IF...)
- imbrication de structures
- utilisation de OCCURS (tableaux) et de VALUE
- gestion de messages en Working-Storage
- utilisation des niveaux 88