

# ***LEDA2 : Un outil en Sas pour recodifier des variables et produire des tableaux sur de gros fichiers***

*Michel De Bie*

L'Insee utilise depuis vingt ans un outil spécifique nommé LEDA. Cet outil a vieilli, sa maintenance est devenue délicate et son transport sur micro-ordinateur est trop onéreux.

LEDA2 a pour but de remplacer LEDA dans ses fonctions Castor, Pollux et Hélène .

LEDA2 se présente comme une surcouche du logiciel SAS, déjà largement utilisé à l'Insee et fonctionne à la fois sur machine MVS et sur micro Windows.

Les données traitées constituent un **groupe de fichiers détails** (par exemple Immeubles, Logements, Individus) reliés entre eux par des identifiants. Les fichiers (data Sas) du groupe sont lus et écrits simultanément par LEDA2.

Les **recodifications** sont spécifiées :

- par des FORMULES de calcul ;
- par des TABLES DE DÉCISION : une table est un tableau de formules traitant des cas alternatifs (SI ALORS SINON)
- par des enchaînements de tels calculs (LISTES DE FONCTIONS)

Les variables recodifiées peuvent être "remontées" (*i.e.* calcul du nombre d'enfants par logement), ou "descendues" (*i.e.* copier la variable confort du logement dans l'observation Individu). Cette fonction rend transparente à l'utilisateur la gestion de fichiers hiérarchisés.

---

1. Castor (structure des données), Pollux (recodification) et Hélène (tabulation) sont les composants de LEDA.

Les **tableaux** sont plus généraux que ceux de Proc Tabulate de Sas :

- possibilité de définir un filtre pour chaque tableau d'une demande LEDA2 alors que Tabulate ne permet qu'une clause WHERE commune à tous les tableaux créés lors d'une étape Tabulate) ;

- recodification "à la volée" des critères (variables de classification) et des cumuls (variables sommées) juste avant la tabulation et dans la même étape.

Plusieurs recodifications et tableaux peuvent être produits **par une seule lecture** d'un groupe de fichiers (économie pour de gros fichiers).

Les spécifications introduites et testées interactivement, sont archivées dans un dictionnaire pour documenter les résultats et les reproduire si nécessaire.

## Les choix techniques

### *Les données détaillées*

On considère que la collecte et saisie des données est terminée et qu'il n'y a plus d'insertion ou de destruction d'observations; l'exploitation ici consiste à retraiter les données pour produire des résultats.

Les données sont dans des fichiers Sas pour faciliter le travail des statisticiens et sont organisées en unités statistiques (US) ou "niveaux" selon une structure arborescente telle que District, Immeuble, Logement, Famille, Individu. Les niveaux de cette structure constituent un GROUPE de fichiers. Chaque niveau est un fichier Sas indépendant dont les observations sont identifiées par une clé. Un niveau fils contient la clé du père (un moyen de trouver les données du père).

La technique Sas retenue consiste, dans une étape Data, à lire **séquentiellement** les différents niveaux **triés** sur les clés, en comparant ces clés. Ce choix est encapsulé dans une méthode de lecture et peut assez facilement être modifié. Il a été pris après divers essais rejetés et qui sont les suivants :

- "merges" Sas en cascade pour "aplatir" les fichiers: temps des merges élevé, et finalement la taille disque passe de 1 à 1,70 ;

- indexation des datas: la taille disque passe de 1 à 1,10 ou 1,15, le temps de lecture de 1 à 4 ;

- "View Sas" sur fichiers indexés le temps de lecture passe de 1 à 10 parce que la "view" est recalculée à chaque appel du groupe.

Les fichiers LÉDA anciens peuvent être lus en état (sans recopie), il en sera de même des fichiers FIS (norme Insee de format d'échange et d'archivage).

## ***Les traitements ou spécifications***

Pour définir ces traitements, les dialogues sont écrits en Sas/SCL : la spécification d'un traitement particulier est conservée dans un petit data Sas, mis à jour en SCL.

Chaque traitement est traduit dans une macro contenant un texte en Sas de base et cette macro est elle aussi conservée. Un petit test d'exécution est fait dès la génération.

L'exécution réelle est faite en assemblant la méthode de lecture (Sas, Leda ou Fis), les macros de travail et les choix de sortie (indépendants des formats d'entrée) en une "grosse" étape data, qui donne les fichiers détails recodifiés (GROUPE en sortie) et les tableaux "résumés" (format Sas/Summary). Ces tableaux sont repris et édités par Sas (Proc Summary, Tabulate, Etape data ...).

## ***Les programmes***

Pour l'utilisateur, le mode opératoire sera identique devant les différents affichages. Une première version dite DINO(saure) sert de prototype pour la méthode et la mise au point. Une deuxième version dite SYMPA, utilisera plus complètement les possibilités d'affichage de Windows.

Cette évolution est possible grâce à la normalisation des programmes et en particulier à la séparation des actions et des affichages.

Des souhaits un peu "futuristes" tels que spécifier un tableau en le dessinant ont été écartés (provisoirement ?).

## **Organisation d'une base LEDA2**

### ***Les données (groupe de fichiers)***

Un groupe de fichiers réunit les fichiers qui "vont ensemble", par exemple les cinq US d'un recensement qui contiennent ensemble l'information disponible pour l'exploitation.

LEDA2 peut traiter un groupe unique ou un groupe éclaté en sous-groupes (les régions dans le cas d'un recensement).

Une base LEDA2 de test (pour la mise au point) contient un groupe d'essai en Sas. Les données réelles (pour l'exécution) peuvent être en format Sas, en format LEDA ou en format FIS.

### ***Les spécifications***

#### **Organisation des données**

Trois data Sas définissent les données :

- DICBASE décrit le type de base LEDA2 (nombre de sous-groupes, format Sas, LEDA, FIS) ;
- DICLIENS décrit les liens entre les fichiers d'un groupe ;
- DICGRP donne le chemin physique des sous-groupes, si la base a plus d'un sous-groupe.

#### **Le catalogue des spécifications**

Deux data Sas définissent les traitements :

- DICSPE recense les objets de la base et en assure l'intégrité (mise à jour, duplication, sécurité ...)
- DOCSPE donne pour chaque objet la liste des variables utilisées en entrée et en sortie.

## Les objets de spécification

Trois types principaux :

### *Recodifications*

Table de décision (D) [si ligne=vraie alors action sinon voir la ligne suivante]

Formule de calcul (F) [ ZEAT=SUBSTR(REGION,1,1)]

Remontée d'un code ou d'une quantité (R) [CSM= CS de l'individu qui est chef de ménage].

Comptage d'US (C) [NBENF=nombre d'individus d'âge=15 ans dans le ménage (logement).

Séquence de fonctions ou imbrication paramétrée(S).

### *Tableaux*

Demande de tableau (T) [si LEDA2 utilisait le vocabulaire de TABULATE, on dirait : critères=class, cumuls=var, édition=table]

### *Passages*

- Demande de passage (exécution) (P) [sous-groupe d'entrée, recodifications et tableaux à faire (plusieurs tableaux), sorties à garder]

Chacun de ces objets est enregistré dans un data Sas de la base LEDA2. Ces objets sont créés et testés en mode "mise au point", le test génère une macro de même nom que le data. Les macros sont conservées dans répertoire (MACROGEN) pour être reprises à l'exécution sans régénération.

## Documentation des variables

Deux types d'objets : les variables et les listes.

- DOCVAR (type V)

Une variable d'entrée peut être décrite par un **titre**, un thème (ou mots-clés) pour la

recherche, une **liste de modalités** et une **liste d'intitulés des modalités**, pouvant être repris dans un tableau.

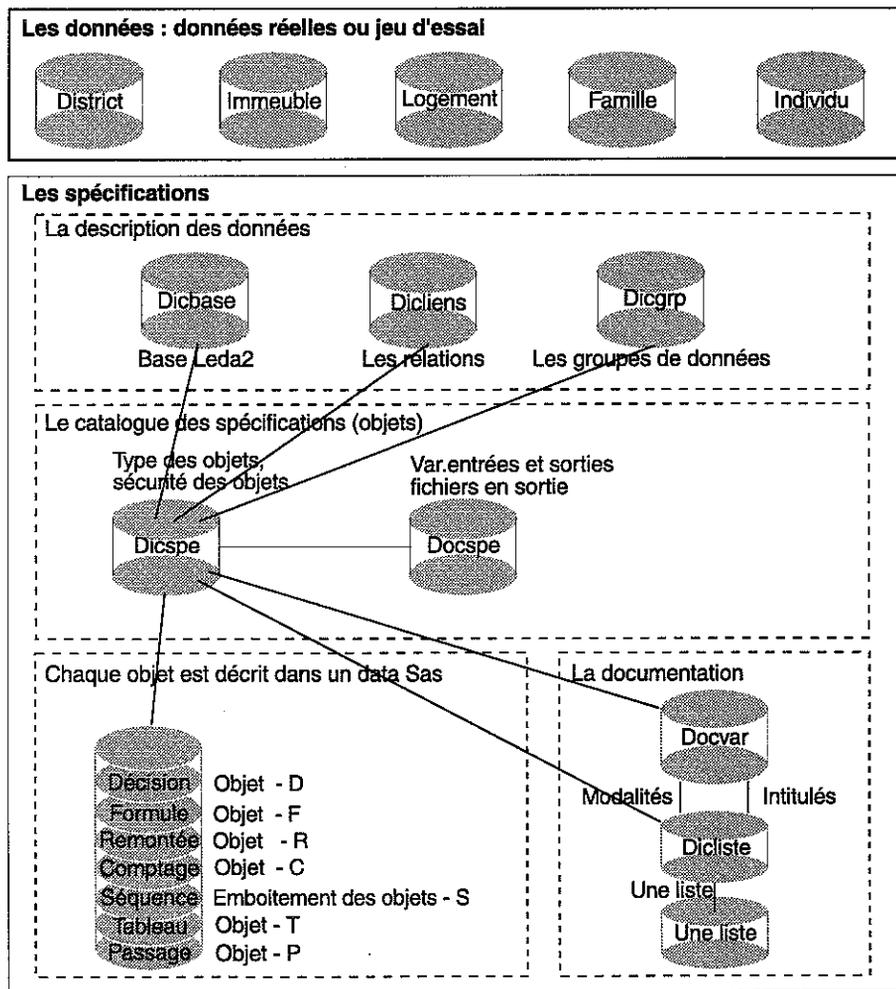
Une variable de sortie doit en plus être définie en nature (caractère ou numérique) et en longueur ;

- DICLISTE (type L) est un catalogue des listes déjà définies (nom et titre)

La liste elle-même est un data SAS de nom égal à celui de la liste et contenant une variable : un élément de la liste par observation.

Graphique 1 :

**Les composants d'une base LEDA2**



## La description des données

### DICBASE

Ce fichier a toujours une seule observation, extraite des inventaires PRIVE ou PUBLIC des bases LEDA2, qui ont le même contenu, pour chaque base inventoriée.

FSVIEW : LEDA2.DICBASE (B)				
OBS	NOM	TYPE	TITRE	CHEMIN
1	RP902LD2	SAS1	TEST LD2	NA90.L02.RP902LD2

### DICLIENS

À noter la cascade de liens ; ici l'arborescence n'a qu'une branche.

FSVIEW:	LEDA2.	DICLIENS (B)			
OBS	US	ID	USUP	LIEN	FILTRE
1	DISTRICT	IDDST			
2	IMMEUBLE	IDIMM	DISTRICT	IDDST	
3	LOGEMENT	IDLOG	IMMEUBLE	IDIMM	
4	FAMILLE	IDFAM	LOGEMENT	IDLOG	
5	INDIVIDU	IDIND	FAMILLE	IDFAM	

### DICGRP

Ici on a un seul groupe situé dans la librairie SAS NA90.L02.ENQUETE et contenant les cinq data SAS District à INDIVIDU.

FSVIEW: LEDA2.DICGRP (B)				
OBS	GROUPE	FICHER	US	CHEMIN
1	ENQUETE	DISTRICT	DISTRICT	NA90.L02.ENQUETE
2	ENQUETE	IMMEUBLE	IMMEUBLE	NA90.L02.ENQUETE
3	ENQUETE	LOGEMENT	LOGEMENT	NA90.L02.ENQUETE
4	ENQUETE	FAMILLE	FAMILLE	NA90.L02.ENQUETE
5	ENQUETE	INDIVIDU	INDIVIDU	NA90.L02.ENQUETE

## Un fichier détail

FSVIEW:LEDA2.FAMILLE (B)											
OBS	TYPE	IDENT	DEP	COM	ST	STUPC	TU	CLR	NPR	ENGFA	MOISCA
1	01	111000490	75	107	9	907	8	01	01	0356	06
2	01	111000560	75	107	9	907	8	01	08	0356	05
3	01	111000630	75	107	9	907	8	01	03	0356	05
4	01	111000700	75	107	9	907	8	01	03	0356	06
5	01	111001190	75	107	9	907	8	01	00	0356	06

## Le catalogue des spécifications (SPE)

### DICSPE

FSVIEW:									
OBS	TYPE	OBJET	US	USUP	NBT	NBAF	FILTRE	TITRE	
1	Décision	TESTEM	INDIVIDU		3	1			
2	Remontée	CSM	INDIVIDU	LOGEMENT			L1=1	cs chef	
3	Comptage	NBENF12	INDIVIDU	LOGEMENT			AGD<=12		
4	Tableau	TESTTAB1							
5	Passage	TESTPAS1							

La colonne Objet donne le nom d'un élément de spécification. Ce nom est celui d'un data Sas de la base LEDA2 qui contient la définition du calcul et celui d'une macro qui contient le texte Sas réalisant le calcul dans une étape DATA.

Les colonnes Usup et filtre servent pour les remontées ou les comptages, les colonnes nbt et nbaaf pour les tables de décision (nombre de colonnes de test et d'affectation).

DICSPE contient en plus plusieurs variables de gestion pour assurer l'intégrité des données, telles que les dates de création et mise à jour, un identifiant de l'auteur, un numéro de version, un compteur d'utilisations en cours (dans des passages ouverts et non encore terminés).

### DOCSPE

Le rôle E/S définit l'emploi (entrée ou sortie) d'une variable pour un objet. Param est le nom du paramètre, Variable est le nom de la variable à utiliser.

FSVIEW: LEDA2.DOCSPE (B)				
OBS	OBJET	ROLE	PARAM	VARIABLE
1	TESTEM	E	AGE	AGE
2	TESTEM	E	SEXE	SEXE
3	TESTEM	E	EMBRUT	EMBRUT
4	TESTEM	S	EMCOR	EMCOR
5	UNTEST	E	AGE	AGE

Le texte de macro généré est le suivant :

```
%macro TESTEM(age=age,sexe=sexe,embrut=embrut,emcor=emcor);
```

```
.....
```

```
%mend;
```

Et l'exécution peut être :

- %testem() avec les valeurs par défaut, ou bien
- %testem(age=aged,emcor=emcor2) pour une autre exécution.

C'est ce mécanisme Sas/macro de paramètres nommés et ayant une valeur par défaut qui est utilisé pour construire les séquences (fonctions imbriquées).

Le choix des paramètres se fait sur une copie de Docspe.

LEDA2 peut vérifier dans une séquence (et surtout un passage), que toute variable utilisée à un rang donné de la séquence a déjà été calculée ou lue.

## ***La documentation***

### **DOCVAR**

Les informations en entrée viennent du dictionnaire LEDA ou d'un DDS. le fichier est complété par Proc contents des fichiers du groupe. Les champs Nature et Longueur sont obligatoires pour les variables à créer.

FSVIEW:	LEDA2.DOCVAR (B)						
OBS	VARIABLE	NATURE	LONGUEUR	MODALITE	INTITULE	TITRE	THEME
1	SORTIE1	C	3				
2	SORTIE2	N	8				
3	ENTREE1			MODAL1	INTIT1		
4	ENTREE2			MODAL2		Type d'activité	
5	TACT						

Les champs Modalité et Intitulés sont très utiles pour simplifier la définition de tableaux. Les champs Titre et Thème sont documentaires, ils aident à trouver la "bonne" variable.

## DICLISTE

Ce catalogue des listes est consulté avant de créer une liste, pour voir si elle n'existe pas déjà ou pour en copier une qui ressemble à ce que l'on veut.

Le titre est documentaire.

FSVIEW : OBS	LEDA2 LISTE	DICLISTE (B) TITRE
1	AGEQ	AGE QUINQUENNAL
2	MAQTR1	Limite de tranche d'âge
3	LAQTR1	Libellés de tranches d'âge

## UNE LISTE ou plusieurs :

En sélectionnant une liste ou plus (ici MAQTR1 et LAQTR1) on obtient l'affichage suivant en mode Edit de FSVIEW et on peut mettre à jour les listes par paire, ...

La mise à jour est faite sur une copie des listes (dans LEDA2USR) et est enregistrée après confirmation.

LES AUTRES OBJETS sont présentés plus loin.

FSVIEW: LEDA2USR, LISTES (E)	MAQTR1	LAQTR1
1	15	15-24 ANS
2	25	25-29 ANS
3	30	30-39 ANS
4	40	40-59 ANS
5	60	60 OU PLUS

## Les menus de LEDA2

On présente ici un aspect de l'étude détaillée des traitements sous la forme des affichages et des actions engendrées (l'architecture des programmes est un autre aspect, le plan qualité (PAQL) un troisième ...).

## Menu d'accueil

### Présentation du logiciel LEDA2

L'utilisateur remplit d'abord les zones PRIVE et PUBLIC. Faire "entrée" sur une zone vide ou incorrecte équivaut à demander des suggestions. On lui affiche l'inventaire des bases LEDA2 qu'il a déjà utilisées (PRIVE), ou celle des bases présentes sur le serveur (PUBLIC).

Il n'est pas obligatoire de remplir PUBLIC (en mode mise au point on n'en a pas besoin, par exemple). Par contre PRIVE est obligatoire.

Choix d'une base LEDA2 de travail	"PRIVE"
.....(proposer une liste)	
Choix d'une base LEDA2 où lire des données externes	"PUBLIC"
.....(proposer une liste)	

Mode de travail	
Mise au point (des formules de calcul ou des demandes de tableaux)	<input type="radio"/>
Exécution (sur fichiers réels)	<input type="radio"/>
Chargement de données	<input type="radio"/>
Exportation de données	<input type="radio"/>
Utilitaires	<input type="radio"/>

Si l'utilisateur met "+" dans la zone PRIVE et si la base PUBLIC est désignée, la base PRIVE est initialisée à partir de cette base PUBLIC. C'est un chargement de données simplifié qui peut être utile pour un utilisateur occasionnel (SEDDL) novice en LEDA2.

Les choix sont exclusifs (d'où l'utilisation des "boutons radio").

Le choix "Chargement de données" peut être entré plusieurs fois pour construire le groupe par des fichiers de différentes origines ou bien fusionner des spécifications.

Il est alors prudent de passer en mode "utilitaires" et de vérifier la cohérence.

Suite au choix "exécution", on demande d'indiquer la base LEDA2 contenant les spécifications à exécuter (par défaut la base PRIVE désignée). Ces spécifications (et leurs images sous forme de macros) seront alors transportées dans la nouvelle base de travail.

## ***Mode "mise au point"***

On affiche une liste de tous les objets de la base, avec leur type (D, F, R, C, S, T, P, V) et leur titre. Un QBE (Query By Example) permet de faire des sélections. L'affichage présenté ici pour DICSPE (§ 3.2.3) est repris pour DOCVAR et DICLISTE.

### **Deux façons d'entrer des commandes :**

- Le mode "MENU" : on sélectionne une ligne et on clique sur le bouton voulu du menu d'action ;
- il équivaut au mode "commande" mais en plus le choix est filtré (grisé ou non) pour WINDOWS ;
- ce mode est commode avec une souris, le bac "command" ne sert à rien ;
- le mode "COMMAND" (type ISPF, MEMO, ...) ;
- ce mode est commode avec un clavier, le menu d'action ne sert que d'aide-mémoire.

En face de chaque ligne, une zone (bac "command") permet d'indiquer ce qu'on veut faire : M pour Mettre à jour (édit), V pour Voir (browse), C pour Copier (!), I pour Imprimer, D pour Détruire, A pour Ajouter (créer).

La commande "A" déclenche la création d'un objet dont le nom et le type seront définis dans une boîte de dialogue.

La commande "C" copie l'objet avec son type sous le nom donné (et accepté) dans une boîte de dialogue.

La commande "D" détruit l'objet et la macro associée après message de confirmation dans une boîte de dialogue et sous réserve que cet objet ne soit pas en cours d'utilisation et bloqué par sécurité.

### **Recherche d'un objet**

Dans les deux modes la ligne QBE permet de sélectionner des objets par le type, le début du nom ou un mot figurant dans le titre :

Si on met dans la ligne QBE un type et un nom qui n'existe pas le logiciel renverra une liste vide.

## L'écran principal. (affichage DICSPE)

Evidemment toutes les commandes ne sont pas permises. On ne peut pas éditer un fichier ou un dictionnaire. En mode mise au point on ne peut pas éditer des demandes de passage, etc. Mais on a là un canevas unique pour présenter une bonne partie des traitements de LEDA2, ou pour simplement consulter les objets existants. Cela évite d'avoir à monter et descendre dans une hiérarchie compliquée de menus.

## Fonctionnement

MENU	Ajouter	Mettre à Jour	Voir	Copier	Imprime	Détruire
Com.	Type d'objet	Nom	Titre			
QBE						
	Décision	codif1	Codif de l'age Q			
	Tableau	tab1	Tableau sexe*age			
	Groupe	ind	fichier Individu			
	Groupe	diciens	liens entre les fichiers			

Par convention, en mode "mise au point" PRIVE désigne une base contenant des données de test (de faible volume). Aucune option n'est proposée dans ce mode, c'est à dire que, systématiquement, la mise à jour d'une table ou d'une formule provoque son analyse, la génération d'une macro de même nom, l'exécution sur un groupe de fichiers de tests (fichiers détail de sortie non stocké dans la bases LEDA2 sauf demande expresse) avec affichage des data résultants (ou des tableaux éditables).

## Mode "Exécution" de spécifications sur des données réelles (Passage)

Dans le mode exécution on exécute des **spécifications précédemment testées**. L'appel de ce mode suppose donc qu'on indique où trouver ces spécifications (SPE), et qu'on indique les bases (PRIVE et/ou PUBLIC) contenant les données "réelles".

Pour définir un "passage", on assemble les demandes de codification et de tabulation précédemment élaborées en mode "mise au point". On est donc en mode "exécution". En entrant dans ce mode (voir menu "choix de mode de travail"), on a indiqué une base LEDA2 où figuraient les demandes précédemment mises au point (ensemble SPE).

Cet ensemble SPE externe est alors importé dans la base de travail (base PRIVE), et fusionné avec les spécifications déjà contenues dans cette base.

## Les entrées du passage

Pour le choix **a**, si le groupe de données est simple (§ 3.1) il n'y a pas d'ambiguïté. Si le groupe de données est décomposé en sous-groupes (§ 3.2), la liste des sous-groupes est affichée et on demande si il faut les traiter tous ou certains seulement.

## Les calculs du passage

Le choix **b** indique les "demandes": codifications et tableaux à produire dans le passage.

Demande de passage	
<i>Nom du passage :</i> _____ (proposer une liste si blanc, si le nom n'existe pas on considère que c'est une nouvelle demande, on peut reprendre une ancienne demande et la sauvegarder sous un autre nom)	
<b>Titre</b>	
<input type="checkbox"/> a	Description des données en entrée du traitement
	<input type="radio"/> PUBLIC (par défaut)
	<input type="radio"/> ou PRIVE ?
<input type="checkbox"/> b	Liste des codifications et tabulations à exécuter
<input type="checkbox"/> c	Description des données en sortie (résultats, fichiers)
<input type="radio"/> d	Devis
<input type="radio"/> e	Accès au JCL
<input type="radio"/> f	Exécution

### *La liste des calculs*

L'écran suivant est affiché et désigne une séquence de demandes (l'ordre peut avoir de l'importance) et LEDA2 contrôle cet ordre (sans l'imposer).

**On peut enlever une demande en effaçant son nom de la liste**

On peut entrer directement un nom de demande ou positionner le curseur sur une zone vide (ou avec un ?) et faire entrée. Cela provoque l'apparition d'une liste de choix possibles sous la forme suivante, déjà vue (seul le menu change) :

***La sélection des objets du passage. (affichage de DICSPE)***

On indique "S" dans la zone commande (ou bouton "Sélectionner") pour sélectionner les fonctions. Si on met V (ou bouton "Voir") on peut examiner leur contenu. Ce sont les 2 seules opérations permises. On ne peut pas ici renommer les variables.

Passage XXXXXXXX - liste des codifications et tableaux à effectuer			
N°	Demande	Titre	Caractéristiques annexes
10	AGEQ	Codification de l'âge quinquennal	AGE#AGEQ\$IND
20	NAP40	Activité économique agrégée	NAP600#NAP40\$IND
30	TAB1	Tableau age Q et Sexe	

On sélectionne un ou plusieurs objets et on peut annuler la sélection (brillance).

Les demandes retenues (surbrillance) sont transférées dans le premier panneau en faisant PF3, sont ordonnées par défaut par LEDA2 selon les variables utilisées et numérotées pour modifier éventuellement cet ordre.

Quand on sort de **b** il y a déclenchement d'un traitement de vérification de la validité de la demande (pas de mélange entre remontées et ventilations, toutes les variables nécessaires sont bien présentes ....). La liste des variables qu'on peut conserver est

Menu:	Sélectionner	Voir	
Commande	Type d'objet	Objet	Titre
<b>QBE</b>			
	Décision	codif1	Codif de l'age Q
	Tableau	tab1	Tableau sexe'age
	...	...	

mémorisée pour l'étape c.

## Cohérence des calculs

Des messages signalent l'emploi d'une variable qui n'a pas encore de valeur lue ou calculée.

## Les sorties du passage

Le choix **c** affiche l'écran suivant prérenseigné par les options d'environnement :

À propos du dernier carré :

En mode exécution, on écrit toujours dans la base PRIVE. Si c'est aussi la base lue en entrée il s'agit d'une mise à jour "sur place". Dans certains cas ce n'est pas permis ;

- à chaque niveau on peut conserver une variable de niveau supérieur, mais il faut la renommer (un nom de variable doit être unique dans une base LEDA2) ;
- on ne peut pas ici ajouter un niveau. Voir le menu chargement d'une base LEDA2 à partir d'une autre base LEDA2 ;

Passage XXXXXXX : Description des données en sortie (résultats à visualiser, fichiers détail, fichiers ventilations)

Résultats à imprimer ou visualiser ultérieurement

Sortie sur imprimante (définie dans le menu "environnement")

Sortie sur fichier RESULTAT (idem)

Fichiers ventilations à garder (seulement si la demande en comporte)

OUI  format SAS/Summary

format Quattro/(FIS)?

format SEDDL

NON

Données détail à conserver en sortie (seulement si la demande comporte des codifications)

OUI

NON

→ aiguillage sur un menu QBE proposant la liste des variables qu'on peut garder, avec possibilité de renommage

- il n'y a pas de données Leda, FIS, et encore moins CRT90 en sortie.

Remarque générale : il faut stocker toutes ces informations pour pouvoir reprendre une demande de passage ancienne.

### ***Les fichiers de données détail en sortie (recodification)***

Le choix de garder des données détail en sortie conduit à l'écran suivant :

L'état (Keep=Garder ou Drop=Oter) est déterminé par le menu. Les commandes xxxTout agissent sur toutes les variables sélectionnées par le QBE en cours. On peut ensuite modifier individuellement l'état.

Le groupe des fichiers de sortie a la même structure que le groupe d'entrée, mais on peut supprimer des fichiers en supprimant toutes les variables du fichier.

Pour chaque fichier d'entrée on connaît la liste des variables.

DOCSPE donne pour chaque opération les variables en entrée et en sortie donc on connaît la liste des variables créées par fichier.

La colonne Etat (K/D) peut être remplacée par la surbrillance.

Menu:	Garder	Oter	Garder Tout	Oter Tout
Commande/État	Variable	Fichier	Titre	Theme
QBE →				
K	D	District	Département	
D	CL	Logement	Catégorie de logement	
	AN	Individu	Année de naissance	État Civil
	SEXE	Individu	Sexe	État Civil

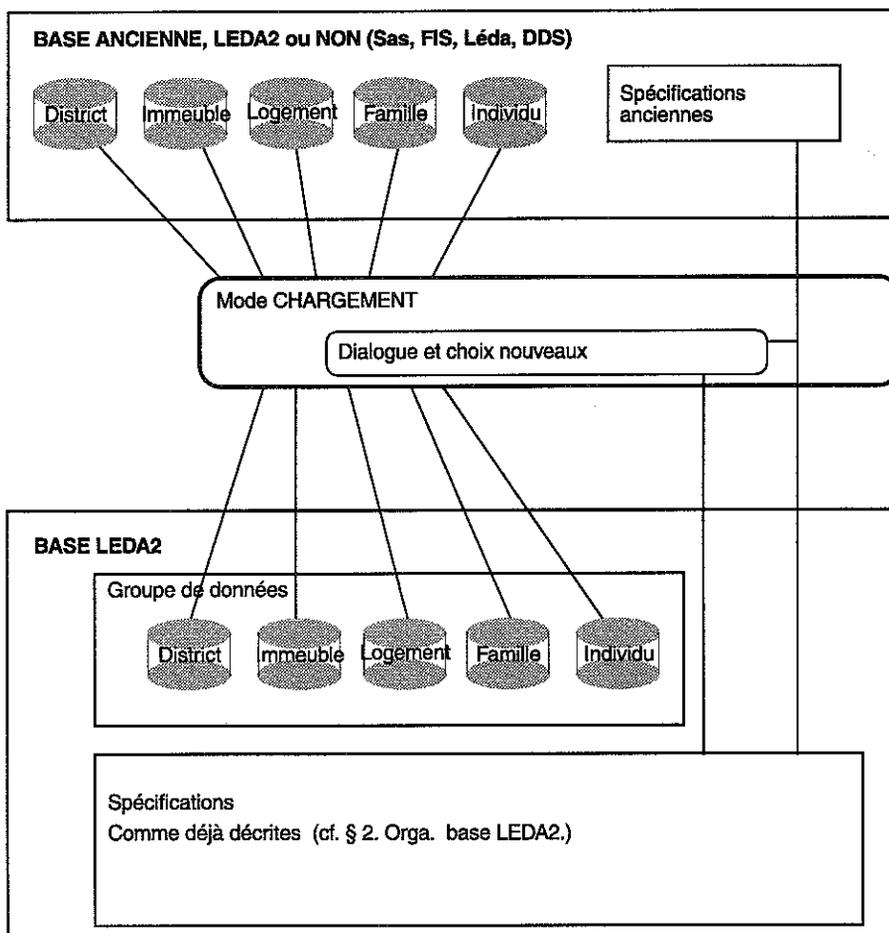
## Chargement de données

Le menu en mode chargement de données :

Dans le **Choix A (LEDA2)** l'utilisateur veut se constituer une **base de test** pour mettre au point des tableaux sur l'enquête Emploi de 1992, par exemple. On pourrait proposer des options par défaut simples : par exemple tirage au 1/1000 sur une US, en gardant tous les descendants.

Graphique 2 :

### Schéma du chargement de données



Ce choix correspond aussi au cas où l'utilisateur veut se constituer une nouvelle base de production, soit par souci de sécurité (ne pas toucher aux données de base), soit parce que il veut changer la structure des groupes de fichiers (aplatir, ou créer de nouveaux niveaux), soit enfin parce qu'il s'intéresse à un seul aspect de l'enquête (il sélectionne alors des fichiers, des thèmes, des variables ...).

Importation dans la base PRIVE		
<b>A</b>	Chargement à partir de la base LEDA2 PUBLIC <i>(extraction d'un échantillon par exemple)</i>	<input type="radio"/>
<b>B</b>	Chargement à partir de fichiers Leda	<input type="radio"/>
<b>C</b>	Chargement à partir de fichiers SAS	<input type="radio"/>
<b>D</b>	Chargement à partir de fichiers FIS	<input type="radio"/>
<b>F</b>	Importation d'informations à partir du DDS d'enquête	<input type="checkbox"/>

Le programme doit donc aiguiller vers des menus qui permettent de sélectionner des variables, des enregistrements et de changer de structure le groupe de fichiers origine. Mais tout cela est rare et assez facile en Sas, il est raisonnable de penser que la personne chargée de ce travail connaît bien Sas.

Le type de base LEDA2 (dans DICBASE) est SAS1 (un seul groupe) ou SASN (plusieurs sous-groupes).

Dans le **choix B (Leda)**, les dictionnaires peuvent être constitués automatiquement, le contenu du dictionnaire Leda étant en général suffisant pour LEDA2.

Le type de base LEDA2 (dans DICDASE) est LEDA1 ou LEDAN.

Dans le **choix C (SAS)** les dictionnaires peuvent être constitués si les identifiants apparaissent clairement. Mais il faudra une intervention manuelle si ce n'est pas le cas.

Le type de base LEDA2 (dans DICBASE) sera SAS1 sauf si l'utilisateur impose un autre choix.

Dans le **choix D (FIS)** le programme se réfère au choix C (par FROMFIS) et les identifiants sont connus en FIS2. On peut garder les types de base LEDA2 FIS21 et FIS2N.

## ***Exportation de données***

Comme les exportations de ventilations brutes au format SEDDL ou au format Quattro sont faites dans la foulée des exécutions de tableaux, il n'y a plus guère à exporter que des données détail en format FIS. Plus exactement on exporte les fichiers et le dictionnaire DICLIENS (la PROC TOFIS est utilisée).

## *Utilitaires*

Deux opérations principales :

### **- Vérification de cohérence.**

On vérifie que tous les objets cités existent et sont cohérents (nombre de colonnes du data décision et nombre de colonnes de test + nombre de colonnes d'affectation dans Dicspé, date des objets ...).

### **- Transport micro --- MVS et réciproquement**

Ce transfert est déclenché actuellement "à la main". Son automatisation est prévue quand les procédures de réseau seront plus sûres.

## **Les différents objets**

### *Une table de décision*

Une table de décision est une suite de lignes qui jouent le rôle de

**si ... alors, sinon** ligne suivante.

Ces lignes sont dans un fichier (data) Sas.

FSVIEW:	LEDA2.TESTEM (B)			
OBS	SEXE	AGE	EMBRUT	EMCOR
1	1	<18	1	1
2	1	<18	1	1
3	2	<15	1	1
4	2	<15	1	1
5				EMBRUT

Exemple édité par FSVIEW (Sas) :

Il y a 3 colonnes de test et une d'affectation, cette table signifie :

Si SEXE .... et AGE .... et EMBRUT .... alors EMCOR= .... sinon ...

La création de cette table (structure : noms et rôles des colonnes) et la modification de structure est faite par la liste suivante :

Les actions sont Insérer, Détruire, Renommer. Les colonnes sont copiées par le nom, sauf les colonnes "renommées" qui le sont par la position.

Nom de la table		→TESTEM	
Nom de l'unité statistique --		→Individu	
Titre		→Test état matrimonial__	
Action	Colonne	Largeur	Rôle
	SEXE	5	TEST
	AGE	10	TEST
	EMBRUT	5	TEST
	EMCOR	20	AFFECTATION

L'exemple cité vérifie "l'âge légal du mariage" (ici 18 ans pour les garçons et 15 ans pour les filles).

Il crée une variable EMCOR ne comprenant ni mariés, veufs ou divorcés en dessous de l'âge légal.

Le texte généré sera :

```
%macro testem(sexe=sexe,age=age,embrut=embrut,emcor=emcor);
array testem(5) 8 _temporary_ (0,0,0,0,0);
  if &sexe='1' & &age'18' & &embrut='1' then do;
&emcor='1'; testem(1)=testem(1)+1;
  end;
else if &sexe='1' & &age'18'          then do;
```

```

&emcor='1'; testem(2)=testem(2)+1;
    end;
else if &sexe='2' & &age'15' & &embrut='2' then do;
&emcor='1'; testem(3)=testem(3)+1;
    end;
else if &sexe='2' & &age'15'                then do;
&emcor='1'; testem(4)=testem(4)+1;
    end;
else
emcor=&embrut; testem(5)=testem(5)+1;
    end;
%mend;

```

L'array testem sera conservé en fin de travail, il donne le nombre de fois où une ligne (règle) a servi; Les règles 1 et 2 (ainsi que 3 et 4) donnent le même EMBRUT mais les règles 2 et 4 comptent le nombre de fois où EMBRUT est modifié. Les règles 1, 3 et 5 comptent le nombre de fois où EMCOR=EMBRUT. Les tables de décision génèrent ainsi un "compte rendu" du calcul.

À noter également, la règle "balai" (obs 5 de la table SAS TESTEM) qui intercepte tous les cas non encore traités.

La somme des 5 compteurs correspond au nombre d'individus traités.

Les trois types suivants (Formule, Remontée et Comptage) *ont beaucoup de points communs : ils calculent une variable VARSUP dans le niveau USUP, à partir de variables disponibles au niveau US, par un calcul CAL à réaliser si le filtre FIL est vrai.*

USUP est au dessus de US dans la hiérarchie et on a donc une séquence d'événement : Début-USUP, US est rencontré, [ N fois ], Fin-USUP à laquelle correspond une séquence d'actions: mise à vide de VARSUP, pour chaque US rencontrée, si FIL est vrai alors effectuer CAL (qui modifie VARSUP), lorsque Fin-USUP est vrai, noter la valeur de VARSUP.

- Pour une *formule*, *USUP=US*.

Exemple de formule *ZEAT=substr(REGION,1,1);*

ou bien *DEPCOM=DEPARTEMENT || COMMUNE*.

- Pour une *remontée*, *USUPUS* et la *variable à remonter est VAR*, alors :

- soit VAR est un code (variable de caractère) et le filtre doit désigner une seule US  
Exemple CSM (cs du chef de ménage dans l'us logement) = CS de l'individu du logement qui a le code L1=1 (filtre) L1 est le lien dans le ménage)

- soit VAR est une quantité SAL (salaire) et le filtre peut désigner plusieurs US

Exemple 1 analogue au précédent : *SALM = SAL* si L1=1 (salaire du seul et unique chef de ménage)

Exemple 2 *SALC=somme (SAL)* si L1=1,2 (salaire du chef + celui du conjoint éventuel)

Pour un *comptage*, *USUPUS* et la *variable à remonter est la variable indicatrice US*.

Exemple de comptage :

*IMPUBERE=somme(INDIVIDU)* si (sexe=1 et age ) ou (sexe=2 et age)

- mais on peut effectuer après tabulation et avant édition des calculs entre sous-cases et supprimer l'édition de sous-cases devenues inutiles

- en sortie de l'étape data chaque tableau est un fichier Sas de format Summary

- la mise en page est de style Tabulate (elle est faite par un proc TABULATE générée)

## *Une formule*

ZEAT est le nom de l'objet.

**FORMULE ZEAT**  
**TITRE (facultatif)- Calcul ZEAT**  
**La FORMULE doit être appliquée à l'US DISTRICT**  
**est définie ci-dessous par un TEXTE SAS:**  
**ZEAT=SUBSTR(REGION,1,1)**

sous la condition du **FILTRE**

→

## *Une remontée*

CSM est le nom de l'objet

**La REMONTEE CSM doit copier la VARIABLE- CS**  
**de l'unité statistique- INDIVIDU**  
**vers l'UNITE SUPERIEURE LOGEMENT**  
**sous la condition du FILTRE(obligatoire):**  
**FILTRE → L1= '1'**

**TITRE(facultatif) → CS chef de ménage**

## *Un comptage*

IMPUBERE est le nom de l'objet

<b>Le COMPTAGE IMPUBERE</b>	
des occurrences de l'unité statistique →	<b>INDIVIDU</b>
dans l'UNITE statistique SUPERIEURE →	<b>LOGEMENT</b>
est fait: sous la condition éventuelle du FILTRE:	
FILTRE- (SEXE=1 & AGE > 15) (SEXE=2 & AGE<15)	
TITRE(facultatif) → AGE légal du mariage	

## *Une séquence de fonctions*

Dans son principe la création d'une séquence est analogue à l'opération § 3.3.2. mais en plus on peut modifier la valeur par défaut des paramètres des fonctions pour enchaîner des calculs entre variables.

Par exemple la formule ZEAT a été définie par REGION d'où la fonction ZEAT(REGION).

Dans le fichier on peut avoir plusieurs instances de région telles que la région de naissance REGNAI, la région de domicile REGION, la région de lieu de travail REGLT, la région de résidence antérieure REGRAN pour un individu, et autant pour son père, sa mère, son frère aîné ...

Ici on peut renommer les paramètres et utiliser ZEAT(REGNAI), etc. En fait les macros générées sont des procédures ou "fonctions implicites".

La macro ZEAT est :

```
%MACRO ZEAT(REGION=REGION,ZEAT=ZEAT);  
  &ZEAT=SUBSTR(&REGION,1,1);  
%MEND; et les séquences autorisent la modification des paramètres d'où:  
  
%ZEAT(REGION=REGNAI,ZEAT=ZEATNAI);  
  et plus loin l'utilisation de ZEATNAI  
%TRUC(ZEAT=ZEATNAI ...);
```

## *Une demande de tableaux*

Voici un aperçu de la **mise en page** de l'édition du tableau W4A qui sera défini ensuite.

**En SAS, usuellement, les tableaux sont fabriqués par proc FREQ ou TABULATE (tableaux édités), ou proc SUMMARY si on n'édite pas (tableaux bruts).**

*Les concepts SAS utilisés sont :*

- **la population = le fichier en entrée, il a**

+ une unité statistique (c'est un fichier d'individus ou (exclusif) de logement)

+ une appartenance ou champ (le contenu du fichier et la clause WHERE appliquée) ;

- la pondération (WEIGHT )

- les critères ou variables de classement (CLASS)

+ on peut recalculer une variable de classement par un format

- les cumuls ou variables d'étude (VAR)

+ on applique une ou plusieurs statistiques

à une variable d'étude

- les groupes d'observations triées (BY), équivalents aux critères "ouverts" de LEDA2

- la mise en page (TABLES de freq et TABLE de tabulate, OUTPUT de summary)

Sas applique ces définitions à la population et crée un fichier intermédiaire dans WORK puis il restitue les résultats selon les mises en page demandées.

*Ce fichier intermédiaire interdit de traiter de trop gros fichiers.*

Les concepts de LEDA2 sont comparables à ceux de Sas.

**La principale différence est qu'une action spécifiée en Sas s'applique à tous les tableaux demandés :**

par exemple la clause WHERE s'applique à toute la demande et on ne peut pas produire en même temps un tableau sur la population totale et un autre sur les actifs ; autre exemple : par un FORMAT, on peut regrouper l'âge détaillé en trois tranches, mais alors tous les tableaux de la demande qui utilisent l'âge auront cet âge en trois tranches; encore un exemple: le fichier en entrée est un fichier d'individus ou bien de logements, en Sas on ne peut pas compter simultanément les individus et les logements (pour calculer une taille moyenne des ménages).

Donc *Sas nécessite de nombreuses lectures du fichier et des appariements des résultats.*

Ou inversement le calcul préalable de variables "ad hoc".

**L'organisation de LEDA2 permet des calculs spécifiques à chaque tableau et autorise le calcul de plusieurs tableaux en une seule lecture du gros fichier dans la limite de la mémoire disponible.**

L'accumulation des données est faite en mémoire dans des arrays définis par les **critères fermés (ayant une liste pré définie de modalités), et non dans une WORK.**

La lecture du fichier est faite dans une étape data qui calcule des *variables indicatrices d'unités statistiques; on a ainsi une variable LOGEMENT qui vaut 1 une fois pour chaque logement même vide et 0 autrement, et une variable INDIVIDU qui vaut 1 si il y a un individu et 0 dans un logement vide.* La variable logement à 1 signale le début du logement, une variable FINLOG signale la fin du logement (et à ce moment les remontées au niveau logement sont disponibles). LEDA2 peut ainsi compter en même temps les logements et les individus. L'étape data autorise d'utiliser toutes les recodifications déjà définies et des *recodifications spécifiques à un tableau.* Ces dernières sont :

- **l'appartenance, équivaut à un filtre where par tableau**

- **les critères équivalent à une instruction CLASS par tableau et à des FORMATS spécifiques.**

- *les cumuls, équivalent aux variables d'étude cependant :*  
 + LEDA2 ne dispose que des statistiques  
 COUNT et SUM  
 + mais il autorise une formule de calcul:  
 + exemple : (SALAIRE + PRIME / 12)\* SOND \* INDIVIDU  
 la somme est impossible en Sas (la faire avant ou après la tabulation)  
 le facteur Sond remplace WEIGHT ou FREQ, pour ce tableau  
 le facteur Individu (variable indicatrice) désigne l'unité statistique du cumul, deux  
 cumuls d'un même tableau peuvent avoir des US différentes (non en Sas)  
 + autre exemple: (TACT='11') \* SOND \* INDIVIDU  
 En Sas le test TACT='11' a une valeur logique 0 ou 1 utilisable en numérique  
 On exprime ainsi la sélection d'une sous-population, ici les actifs occupés.  
 - *Les sous-cases ressemblent aux variables d'étude pour le format d'édition*  
 + par défaut on a une sous-case éditée par cumul, comme en Sas  
 + mais on peut effectuer après tabulation et avant édition  
 des calculs entre sous-cases  
 + et supprimer l'édition de sous-cases devenues inutiles  
 - *en sortie de l'étape data chaque tableau est un fichier Sas de format Summary*  
 - *la mise en page est de style Tabulate (elle est faite par une proc TABULATE générée).*

Cette édition (ligne de 132 c, page de 58 lignes) est rétrécie pour voir l'ensemble.

On aurait pu demander l'édition de marges (somme des sexes; somme des tranches d'âges ...) ou bien des pourcentages. Les options d'édition sont celles de TABULATE (ici avec NOSEPS).

### **Le guide (check-list)**

(Ici le tableau s'appelle W4A dans le menu général des objets).

Des utilisateurs ont demandé cette liste dont les boutons s'allument pour indiquer ce qui est déjà fait. Les écrans suivants apparaissent en cliquant sur les boutons de cette liste.

Voici les affichages successifs obtenus par la check liste.

**GUIDE**

Un tableau contient au moins un critère ou un cumul. Les autres choix sont facultatifs (options par défaut)

Choisissez une action en cliquant sur une ligne.

Définition du tableau W4A

Titres du tableau (trois lignes)

Appartenance ou sous-population

Critères ou répartition croisée de la population

Cumuls ou quantités sommables accumulés

Sous-cases éditées après calcul entre cumuls

Contrôle de cohérence (génération en Sas) et ensuite:

Edition du tableau, produit sur un jeu d'essai

Devis partiel

JCL MVS partiel (place disque et temps Cpu)

Retour

Abandon

Quitter LEDA2

Population des 15 ans et plus			11:58, Tuesday, July 18, 1995									
			Population active (y compris militaires) par sexe, âge et type d'activité ACT 1 RES									
			Sexe									
			Hommes					Femmes				
			Actifs	Taux d'activité	Dont actifs occupés	Dont chômeurs	Taux de chômage	Actifs	Taux d'activité	Dont actifs occupés	Dont chômeurs	Taux de chômage
Département	Com mune	Tranche d'âge en années										
02	001	15-24	24	46.2	12	8	33.3	8	22.2	8	0	0.0
		25-29	16	100.0	16	0	0.0	16	100.0	16	0	0.0
		30-39	20	100.0	20	0	0.0	12	100.0	8	4	33.3
		40-59	40	76.9	36	4	10.0	24	35.3	24	0	0.0
	>60	4	6.3	4	0	0.0	4	5.6	4	0	0.0	
	002	15-24	40	76.9	24	4	10.0	8	15.4	4	4	50.0
		25-29	12	100.0	12	0	0.0	4	100.0	4	0	0.0
		30-39	40	100.0	40	0	0.0	40	76.9	28	12	30.0
		40-59	64	84.2	56	8	12.5	44	57.9	36	8	18.2
	> 60	0	0.0	0	0	0.0	0	0.0	0	0	0.0	
	003	15-24	32	38.1	20	8	25.0	28	29.2	16	12	42.9
		25-29	16	100.0	16	0	0.0	4	50.0	0	4	100.0
		30-39	72	94.7	64	8	11.1	48	66.7	44	4	8.3
		40-59	104	76.5	104	0	0.0	64	51.6	64	0	0.0
	60 ou	8	10.5	4	4	50.0	16	15.4	16	0	0.0	
	004	15-24	4	20.0	4	0	0.0	4	50.0	0	4	100.0

		25-29	4	100.0	4	0	0.0	12	100.0	8	4	33.3
		30-39	12	100.0	12	0	0.0	0	0.0	0	0	0.0
		40-59	16	66.7	16	0	0.0	16	50.0	12	4	25.0
		60 ou	4	16.7	4	0	0.0	0	0.0	0	0	0.0
	005	15-24	8	50.0	4	4	50.0	0	0.0	0	0	0.0
		25-29	8	100.0	8	0	0.0	12	75.0	8	4	33.3
		30-39	24	100.0	24	0	0.0	28	70.0	24	4	14.3
		40-59	36	100.0	36	0	0.0	16	50.0	16	0	0.0
		> 60	4	20.0	4	0	0.0	0	0.0	0	0	0.0
	006	15-24	4	33.3	4	0	0.0	4	50.0	4	0	0.0
		25-29	16	100.0	16	0	0.0	12	100.0	12	0	0.0
		30-39	8	66.7	8	0	0.0	8	100.0	8	0	0.0
		40-59	12	75.0	12	0	0.0	16	80.0	12	4	25.0
		> 60	4	14.3	4	0	0.0	8	22.2	4	4	50.0
	007	15-24	12	50.0	8	0	0.0	4	33.3	0	4	100.0
		30-39	12	100.0	12	0	0.0	0	0.0	0	0	0.0
		40-59	8	100.0	4	4	50.0	4	33.3	4	0	0.0
		> 60	4	33.3	4	0	0.0	0	0.0	0	0	0.0
	008	15-24	4	33.3	4	0	0.0	4	16.7	0	4	100.0
		25-29	12	100.0	4	8	66.7	4	100.0	4	0	0.0
		30-39	16	100.0	16	0	0.0	8	50.0	8	0	0.0
		40-59	12	60.0	12	0	0.0	4	20.0	4	0	0.0
		> 60	4	16.7	4	0	0.0	0	0.0	0	0	0.0
	009	15-24	16	50.0	8	0	0.0	16	40.0	16	0	0.0
		25-29	8	100.0	4	4	50.0	4	100.0	0	4	100.0
		30-39	44	91.7	44	0	0.0	44	78.6	40	4	9.1
		40-59	60	83.3	52	8	13.3	20	38.5	20	0	0.0
		60 ou	0	0.0	0	0	0.0	0	0.0	0	0	0.0
	010	15-24	0	0.0	0	0	0.0	0	0.0	0	0	0.0
		25-29	8	100.0	4	4	50.0	4	100.0	0	4	100.0
		30-39	16	100.0	16	0	0.0	16	100.0	16	0	0.0
		40-59	4	100.0	4	0	8	0.0	100.0	8	0	0.0
		60	0	0.0	0	0	0.0	0	0.0	0	0	0.0
	011	15-24	16	26.7	12	4	25.0	16	23.5	16	0	0.0
		25-29	28	77.8	16	12	42.9	28	63.6	20	8	28.6
		30-39	116	96.7	104	12	10.3	80	71.4	76	4	5.0
		40-59	104	89.7	104	0	0.0	76	59.4	76	0	0.0
		> 60	4	4.5	4	0	0.0	0	0.0	0	0	0.0
	012	15-24	8	66.7	8	0	0.0	0	0.0	0	0	0.0
		30-39	12	100.0	8	4	33.3	0	0.0	0	0	0.0
		40-59	8	66.7	8	0	0.0	4	50.0	4	0	0.0
	> 60		4	50.0	4	0	0.0	0	0.0	0	0	0.0
013	15-24		12	42.9	8	0	0.0	20	50.0	12	8	40.0
		25-29	12	100.0	12	0	0.0	12	100.0	12	0	0.0
		30-39	28	100.0	28	0	0.0	24	75.0	24	0	0.0
		40-59	32	100.0	32	0	0.0	16	50.0	16	0	0.0

## Les titres

Titre 1 Population des 15 ans et plus
2: Population active (y compris militaires)
3:            par sexe, age et type d'activité ACT 1 RES

## L'appartenance ou sous-population

appartenance d'un tableau Un tableau peut avoir une appartenance (sous-population) Cette formule logique "tient" sur 200 car. au plus. les opérations sont &(et) !(ou) parmi entre sauf hormis Aide=F1    Retour=F3
Demande de tableau W4A : Appartenance.
Texte de la condition d'appartenance: AGEQ PARM 15 à 95

## Les critères de ventilation (variables de classification)

### *La construction d'un critère : suite d'opérations*

Cette construction est guidée par une check-liste de trois opérations.

**critères d'un tableau brut**  
**Un critère est construit à partir d'une variable**  
**Précisez sa mise en page dans Cadre (Page.Ligne...) et donnez lui un nom pour l'édition (Libellé).**  
**Aide=F1**                      **Retour=F3**                      **et le menu**

**Demande de tableau W4A : liste des critères**

Critères	Cadre	Libellé(label pour tabulate)
D	Ligne	Département
C	Ligne	Commune
AGEQ	Ligne	Tranches d'âge
SEXE	Colonne	Sexe

**GUIDE**  
**Un critère de ventilation est une variable recodifiée. Les modalités du critère et celles de la variable.**

**Choisissez une action en cliquant sur une ligne.**

**Définition d'un critère de ventilation AGEQ**

Définir la codification du critère  
 Vérifier les modalités du critère (croisement)  
 Intitulés des modalités du critère pour l'édition

Retour  
 Abandon  
 **Quitter LEDA2**

***La construction d'un critère : Définir la codification du critère***

Cette première grille reprend la ligne du critère dans l'écran précédent.

La liste de modalités à retenir et le type Tranches définissent la correspondance indiquée à l'écran suivant.

## La construction d'un critère : Vérifier les modalités du critère

C'est cette correspondance affichée ici qui sert. On peut la modifier arbitrairement (commandes P, T, PN, TN), indépendamment de sa construction qui a été faite à partir de la liste citée donnant les limites de tranches d'âge (15, 25, 30, 40, 60).

Si les modalités de la variable en entrée avaient des intitulés, ceux-ci seraient affichés.

La tranche 0-14 ans est exclue (voir l'appartenance).

définition d'un critère: le calcul  
 Un critère est une recodification de la variable  
 Deux opérations sont standardisées: Parmi et Tranches  
 Le résultat est améliorable ensuite.  
 Aide=F1                      Retour=F3                      et le menu

Ventilation W4A    Critère AGEQ

Type de critère: T (Ouvert, Parmi, Tranches)  
 Liste de modalités à retenir:  
 15 25 30 40 60

---

Abréviation désignant les intitulés d'édition  
**LAQTR1**

définition d'un critère: le calcul  
 Correspondance variable-critère déjà définie ou non  
 Cette table sert à la production de la ventilation.  
 Colonne C:    P=parmi    T=tranche  
 Aide=F1                      Retour=F3    et le menu

Tableau W4A : modalités du critère AGEQ

C	Variable	Intitulés/variable	Critère
	'00'		
	'05'		
	'10'		
	'15'		'15'
	'20'		'15'
	'25'		'25'
	'30'		'30'
	'35'		'30'
	'40'		'40'
	'45'		'40'

'50'		'40'
'55'		'40'
'60'		'60'
'65'		'60'
'70'		'60'
'75'		'60'
'80'		'60'
'85'		'60'
'90'		'60'
'95'		'60'

### *La construction d'un critère : Vérifier les intitulés des modalités pour l'édition*

C'est la correspondance affichée ici qui sert. On peut la modifier arbitrairement, indépendamment de sa construction qui a été faite à partir de la liste nommée (LAQTR1) et sans effet sur cette liste.

**les intitulés d'un critère**  
**Correspondance critère-intitulé déjà définie ou non**  
 Elle sert à l'édition de la ventilation.  
 Elle vient de l'abréviation(liste) fournie; mise a jour

Aide=F1                      Retour=F3                      et le menu

**Tableau W4A : intitulés du critère AGEQ**

Critère	Intitulés du critère
15	15-24 ANS
25	25-29 ANS
30	30-39 ANS
40	40-59 ANS
60	60 OU PL

### **Les Cumuls (Variables d'étude)**

Chaque cumul a un filtre qui se combine à l'appartenance, et un calcul qui est une expression numérique quelconque (salaire+primes), multipliée par le nom d'une unité statistique. Chacun de ces champs fait 200 caractères, visibles en cliquant sur ce qui est affiché.

Les constantes ('11' et 12) peuvent être quotées ou non, car Sas au prix d'une conversion "se débrouille" correctement. LEDA2 est capable de vérifier les quotes dans les cas simples, mais cela n'est pas programmé car il faudrait vérifier toutes les syntaxes d'expression SAS pour être général.

définition des cumuls		
Liste des cumuls définis		
Vous pouvez voir les textes complets(clic ligne)		
Aide=F1	Retour=F3	et le menu
Demande de ventilation W4A : Liste des cumuls		
Cumuls	Filtres	Us et calculs
POPACT	TACT PARMi 11 à 13	INDIVIDU * SOND
POPTOT		INDIVIDU * SOND
POPOCC	TACT = '11'	INDIVIDU * SOND
CHOMEUR	TACT=12	INDIVIDU * SOND

## Les Sous-cases (Variables calculées éditées)

Les sous-cases sont calculées par combinaison des cumuls. Par défaut il y a une sous-case par cumul et un format d'édition. Le libellé est alors le nom du cumul. La formule de calcul fait 200 caractères, visibles en cliquant sur le champ.

définition des sous-cases calculées			
Les sous-cases résultent d'un calcul entre cumuls			
Le libellé est le titre (en-tête) à l'édition			
En cliquant sur une ligne vous voyez les textes entiers			
Aide=F1	Retour=F3		
Demande de tableau W4A : liste des sous-cases			
Scases	Calcul	Libellé	Format
POPACT	POPACT	actifs	7
TAUXACT	POPACT*100/POPTOT	taux d'activité	7.1
POPOCC	POPOCC	dont actifs occupés	10
CHOMEUR	CHOMEUR	dont chômeurs	9
TAUXCHO	CHOMEUR*100/POPACT	taux de chômage	7.1

## **La génération en Sas**

Par convention la génération en Sas des objets de recodification est faite systématiquement à la fin du dialogue de mise à jour d'un objet et est suivie d'un test sur un petit fichier pour détecter d'éventuel messages "ERROR" de Sas à voir dans la LOG de SAS.

Les objets TABLEAU génèrent en plus un devis (§4.6.8) et pour examiner ce devis, la génération est explicitement déclenchée par un bouton. De même l'examen du tableau édité (OUTPUT) peut être repris.

## **Les éléments de devis et de JCL**

Au cours de la traduction de la demande en Sas on note des éléments chiffrés utiles au devis tels que le nombre de modalités d'un critère fermé. On peut calculer le nombre de cases livrées, le nombre de recodifications exécutées et estimer la taille du fichier d'entrée. Ces données, combinées avec les règles de facturations fournissent un devis.

Par contre on connaît mal les critères ouverts. Exemple : soit un tableau qui compte les centenaires par communes, si on ne livre que les communes où il y a des centenaires, je n'ai aucune idée de leur nombre, et le programme aussi. Dans ce cas le devis contient un paramètre NC "nombre de communes" à estimer ailleurs.

## **Édition schématique**

Le résultat est visible dans la fenêtre OUTPUT ou sur l'imprimante désignée.

## ***Une demande de passage***

Au § 3.3 on a vu le mode "exécution" qui définit un passage en assemblant des demandes de codification et de tabulation préalablement élaborées en mode "mise au point".

## Conclusion

LEDA2 ne remplace pas SAS mais aide l'utilisateur dans des emplois "pointus" ou fastidieux du langage SAS.

- Accès normalisé à les bases anciennes et documentées.
- Reprise de formats d'archives de données anciens (LEDA) ou nouveaux (FIS).
- Documentation du travail réalisé (c'est nouveau par rapport à SAS).
  
- Conservation des concepts de LEDA.

Les types d'objets gérés permettent de recodifier des variables et de produire des tableaux.

- La méthode de lecture de plusieurs fichiers triés en parallèle est générale.
- Les tables de décisions et les formules sont une présentation claire des recodifications.
- Les tableaux sont équivalents à ceux de Sas et peuvent être produits par lots.

LEDA2 offre plusieurs interfaces :

- chargement de documentation des variables à partir d'un DDS et déchargement.
- chargement de documentation des variables à partir d'un dictionnaire LEDA.
- chargement de données "brutes" en format Sas, venant de Entry Point (EP90).
- lecture directe de données en format LEDA ou FIS (sans recopie intermédiaire).
- présentation d'éléments de DEVIS à un futur logiciel de devis.
- présentation des tableaux produits à SEDDL ou à une chaîne d'édition "propre".

La séparation de la construction d'un objet de spécification et de sa traduction en Sas(génération) permet d'envisager des évolutions telles que la traduction en SQL des demandes pour exécution sur un serveur de données.

Les utilisations les plus nombreuses de LEDA2 seront sans doute des tabulations, par des personnes qui connaissent assez mal Sas, et qui n'ont pas le temps de l'apprendre. Ce sera le cas pour les RP et les utilisations venant de SEDDL. La possibilité de réaliser des recodifications légères est importante, de même que les devis.

LEDA2 doit aussi permettre à un statisticien de codifier une enquête, de fédérer le travail d'une équipe ou de reprendre des spécifications de calcul déjà définies ailleurs.

Un statisticien peut aussi reprendre une enquête ancienne (et documentée) pour une exploitation complémentaire.

Une version DINO(saure) de LEDA2 est en test depuis fin septembre 1995. Une version SYMPA est prévue pour la fin 1995.

## ANNEXE - SYNTAXE DES EXPRESSIONS

*Tout d'abord un exemple*

(entrée) V1	(entrée) V2	(sortie) V3
1	2	'A'
ENTRE 3 5	ENTRE 4 6	'1B'
= 9	PARMI L2	'D'
ENUM 'D' 'F'	PARMI L1	'U'
	PARMI L3	'2'
V2		V1    'F'
		'Z'

Cette table de décision a été entrée en précisant qu'elle comportait 2 variables "de test" (V1 et V2) et une variable calculée (V3). Autrement dit cette table représente une fonction  $V3=f(V1,V2)$ .

La première ligne s'interprète simplement: si  $V1=1$  et  $V2=2$  alors  $V3='A'$ .

La dernière également: dans tous les cas non rencontrés dans les lignes précédentes  $V3='Z'$

Dans les autres cases on voit quelques exemples de **la syntaxe permise par LEDA2**.

- "ENTRE 3 5" en colonne 1 signifie "si V1 est compris entre 3 et 5" (bornes comprises).
- "ENUM" introduit une liste, "PARMI L1" fait référence à une liste nommée L1.

Une case de table (en test ou affectation) peut contenir une expression SAS. En colonne 1 l'avant dernière ligne signifie "si  $V1=V2$  alors" et en colonne 3 " $V3=V1 || 'F'$ " (concaténation Sas)

Les lignes sont examinées dans l'ordre, dès qu'une condition est satisfaite on sort de la table.

Cette syntaxe a d'abord été définie pour les cellules (cases) des tables de décision.

Pour comprendre il faut savoir que le programme teste seulement le premier mot de la cellule, le reste est traduit formellement, sans tester.

### Expression logique, cellule de test :

ENUM énumération  
exemple : enum 1 3 5 (sans virgules, avec des quotes ou non)  
SAUF énumération  
la négation de ENUM  
PARMI nom-liste  
exemple : parmi maqtr1 (un seul nom de liste ou nom d'abréviation)  
HORMIS nom-liste  
la négation de PARMIS  
ENTRE lim-inf lim-sup (sans virgules, avec des quotes ou non)  
exemple : entre 15 25 (valeurs limites comprises)  
HORS lim-inf lim-sup  
la négation de ENTRE (donc valeurs limites exclues)  
joker ou "blanc"  
une cellule vide est toujours vraie  
une comparaison (le premier signe est un opérateur de  
exemple : = 18 comparaison<, <=, =, >=, >, ou leur négation)  
dans une colonne C cela est traduit par C = 18  
un seul mot (interprété comme "= mot")  
exemple1 : 18 ( C = 18)  
exemple2 : toto ( C = toto)  
autres cas (on suppose que c'est un texte SAS)  
exemple: y=substr(z,a,b) (génère if (y=substr(z,a,b) sans analyse)

### Expression scalaire ou cellule d'affectation

La question est encore plus simple : on ne teste quasi rien :  
joker ou "blanc"  
une cellule vide ne génère rien (et surtout pas "C=;" erreur Sas assurée!)  
autrement  
exemple : X+Y!!Z (génère C=X+Y!!Z, en mêlant caractère et numérique, Sas se débrouille)

### Discussion

C'est sommaire et expéditif. On n'a pas l'énumérateur "à" (00 à 99) qui existait en LEDA.

On n'a pas d'expression composée par "and" (&) et "or" (! ou bien |), ni de parenthèses de priorité. C'est acceptable dans les tables dont les cellules de test se composent par "and" et les règles par une forme de "or", le "sinon". C'est insuffisant pour les filtres et appartenances.

Une solution est de n'autoriser que la **syntaxe SAS** dans les **filtres** des objets et donc d'interdire les six mots-clé LEDA2. Une autre serait d'autoriser la référence à des formules, en reportant la difficulté sur les formules.

Une solution plus cohérente est utilisée dans les tableaux, (appartenance et filtres), où on autorise les six mots-clé LEDA2 en les simplifiant, au prix d'une complication des programmes :

PARMI et ENUM font la même chose : on voit bien si ce qui suit est un nom de liste ou une énumération ou un nom de variable.

De même pour HORMIS et SAUF.

On traite l'énumérateur "à" sur des limites numériques de même taille.

On traite "and" et "or" avec des parenthèses de priorité.

Bien sûr il faut examiner tout le texte, et ça prend du temps ; mais on résout toutes les listes et on identifie toutes les variables utilisées.

## **Un autre point délicat**

C'est la question : faut-il "quoter" obligatoirement les constantes caractère ou non ?

En principe Sas demande de quoter les constantes chaîne de caractère ; c'est obligatoire avec l'opérateur IN ( ) (équivalent à ENUM) où on ne peut pas mélanger des constantes quotées et d'autres non quotées. Par contre ailleurs, et au prix de conversions inutiles, Sas rétablit un fonctionnement correct (disons supportable) si on ne quote pas.

Pour comparer une variable numérique et une variable caractère, Sas convertit la variable caractère en numérique et si la conversion est erronée retient la valeur "sans objet".

Ce fonctionnement est "sympathique" mais parfois déroutant.

Ainsi " '3X' 2 " est vrai car " . 2 " mais ça n'est pas intuitif!

Le compromis qui existe dans l'ancien LEDA consiste à exiger les quotes, sauf dans les listes parce que les listes sont traitées par un module qui rajoute les quotes si nécessaire (et qui traite l'énumérateur "à").