

MAFE Methodological Note 8

Manipulation des bases biographiques

Application à la transnationalité des unions

STATA

Marc Thevenin (INED-SMS)

Janvier 2015



*Funded under the
Socio-economic
Sciences & Humanities
Theme*



TABLE DE MATIERES

INTRODUCTION	3
PARTIE I : Création de la base d’analyse en format individus/années	7
1) Base listant toutes les années de vie du répondant jusqu’ à l’année de l’enquête	7
2) Base individus-années listant les pays de résidence du répondant	9
3) Base individus-années listant les pays de résidence des conjoints	13
a) Pays de résidence des partenaires	13
b) Les périodes d’union du répondant	14
c) Appariement des bases répondants et conjoints.....	18
PARTIE II. Exemples de création de variables d’analyse	20
1) Création des variables de statut matrimonial	20
a) Les motifs et les années de séparation	20
b) Les années de mariages	22
c) Variables dichotomiques de statut matrimonial	23
d) Variable synthétique de statut matrimonial.....	24
2) Variable de comparaison des pays de résidence des répondants et des partenaires	26
a) Comparaison des codes pays entre le répondant et le(les) partenaires.....	26
b) Création des variables synthétiques de localisation	26
ANNEXES	37
Création de la base d’analyse	38
1) Création de la base individus années <i>year.dta</i> à partir d’un module biographique	38
a) Modules dont les trajectoires ne sont pas strictement continues	38
b) A partir d’un module strictement continu, <i>house.dta</i> ou <i>activity.dta</i>	38
2) Pays de résidence du répondant à partir de la base « house »	39
3) Correction de l’overlapping : réduction de la largeur du reshape	39
Création des variables d’analyse	41
1) Report des informations sur les séparations	41
a) Réduire le report aux années sans union.....	41
b) Reporter les informations à partir de la base <i>union.dta</i>	41
2) Comparaison des pays de résidence des co-épouses	42
Programme alternatif	44

INTRODUCTION

L'objectif de ce tutoriel est de donner des outils de manipulation des modules biographiques de l'enquête MAFE avec STATA, en particulier lorsqu'il s'agit de mettre en œuvre une analyse biographique à temps discret. En termes de prérequis avec le logiciel, seules les opérations comme la création de variables et la fusion de bases sont nécessaires, les autres instructions étant brièvement présentées dans le tutoriel avec si nécessaire des liens vers le manuel de référence. Du fait de la complexité des données, il convient impérativement de consulter la documentation méthodologique présente sur le site mafeproject.com, en particulier la note méthodologique décrivant toutes les bases mises à disposition (« [Introduction to the MAFE datasets](#) »).

Objectifs du tutoriel

- I. **Changement de niveau d'observation** : les modules biographiques de MAFE sont livrés en format individu-événement (ex : dans le fichier des unions, pour chaque individu, une ligne par union). Si ce type de format permet déjà de réaliser des analyses, la nécessité de combiner plusieurs modules ou l'utilisation de méthodes en temps discret demande de modifier le niveau d'observation. On va donc créer une base d'analyse qui donne *pour chaque année* des informations sur le pays de résidence du répondant et de son ou ses conjoints, ainsi que sa situation maritale. Le niveau d'observation passe donc de l'événement vécu à l'année vécue.

- II. **Création de « variables du moment »**: On cherchera à créer des variables synthétiques qui permettent de renseigner pour chaque instant d'observation les caractéristiques d'état d'un individu. Ces variables sont souvent appelée TVC pour « Time Varying Covariate » dans la littérature technique relative à ce type de problématique. Par exemple, on cherchera à renseigner une variable donnant pour chaque année le statut matrimonial annuel du répondant et une information sur la transnationalité des unions qui tient compte des situations passées.

D'une base individus-événements à une base individus-périodes

En entrée : on dispose d'un fichier *individus-événements*. A l'exception de la base *general.dta*, tous les modules biographiques de MAFE sont livrés sous ce format avec une variable indiquant le rang lié à l'évènement (num_event), l'année d'entrée dans l'évènement (qxxxd) et l'année de sortie (qxxxf), ainsi que diverses caractéristiques propres à l'évènement.

Dans la base *union.dta* par exemple, le rang de l'union est donné par la variable **num_uni**, l'année de début de l'union par **q102d**, l'année de fin par **q102f**, le motif de rupture de l'union par **q102** et l'année de mariage par **q103**.

Si l'année de fin du dernier épisode est une valeur manquante, cela indique que l'évènement était toujours en cours au moment de l'enquête.

Base individu-évènements

ident	num_event	qxxxd	qxxxf	VarX
SN000000	1	1970	1974	a
SN000000	2	2005	.	b

En sortie: on souhaite disposer d'un fichier *individus-périodes* (ou plus précisément, *individus-années*). Après avoir calculé la durée de l'évènement, on va transformer la base pour à avoir une ligne par année.

Base individu-années

ident	year	num_event	qxxxd	qxxxf	VarX
SN000000	1970	1	1970	1972	a
SN000000	1971	1	1970	1972	a
SN000000	1972	1	1970	1972	a
SN000000	2005	2	2005	2008	b
SN000000	2006	2	2005	2008	b
SN000000	2007	2	2005	2008	b
SN000000	2008	2	2005	2008	b

Ceci implique :

1. Une transformation des bases individus-évènements en bases individus-périodes en tenant compte des trajectoires continues et des fins d'évènement l'année de l'enquête.
2. Une transformation, appelée souvent reshaping dans la littérature technique, des bases lorsque plusieurs évènements de même type ont lieu en même temps (cas de « temporal overlapping »). Ici, le « temporal overlapping » est engendré par la présence d'unions polygames. Dans ce tutoriel, la polygamie est seulement traitée du point de vue d'un mari (le répondant) avec plusieurs épouses. Lorsque le répondant est une femme, bien qu'on dispose dans MAFE d'informations sur les épisodes de migration des autres co-épouses, on ne connaît pas leurs années d'entrée et de sortie de l'union.

Ressources mises à disposition

- **Stata_tuto.do**: la totalité du programme décrit dans ce document (sauf les compléments qui figurent en annexe).
- Extraits des bases de données de MAFE : **general.dta**, **migration.dta**, **house.dta**, **union.dta** et **network.dta**, pour 10 répondants et une sélection de variables.

Ces ressources sont téléchargeables à cette adresse:

http://mafeproject.site.ined.fr/fichier/s_rubrique/20386/mafe.tutorial.files.janvier.2015.fr.zip

Documentation sur les bases

Avant toute manipulation des bases, il est fortement conseillé de consulter la documentation de l'enquête, surtout celle dédiée aux bases de données du document **presentation of the datasets**, et

plus particulièrement pour ce tutoriel, la partie consacrée aux modules biographiques (à partir de la page :

http://nesstar.ined.fr/other_docs/IE0216A_PresentationDatasets.pdf

- Base **general.dta** : page 32-33.
- Base **house.dta** : page 34.
- Base **migration.dta** : page 36.
- Base **union.dta** : pages 34.
- Base **network.dta** pages 42-46.

Différences avec les tutoriels SAS et R¹

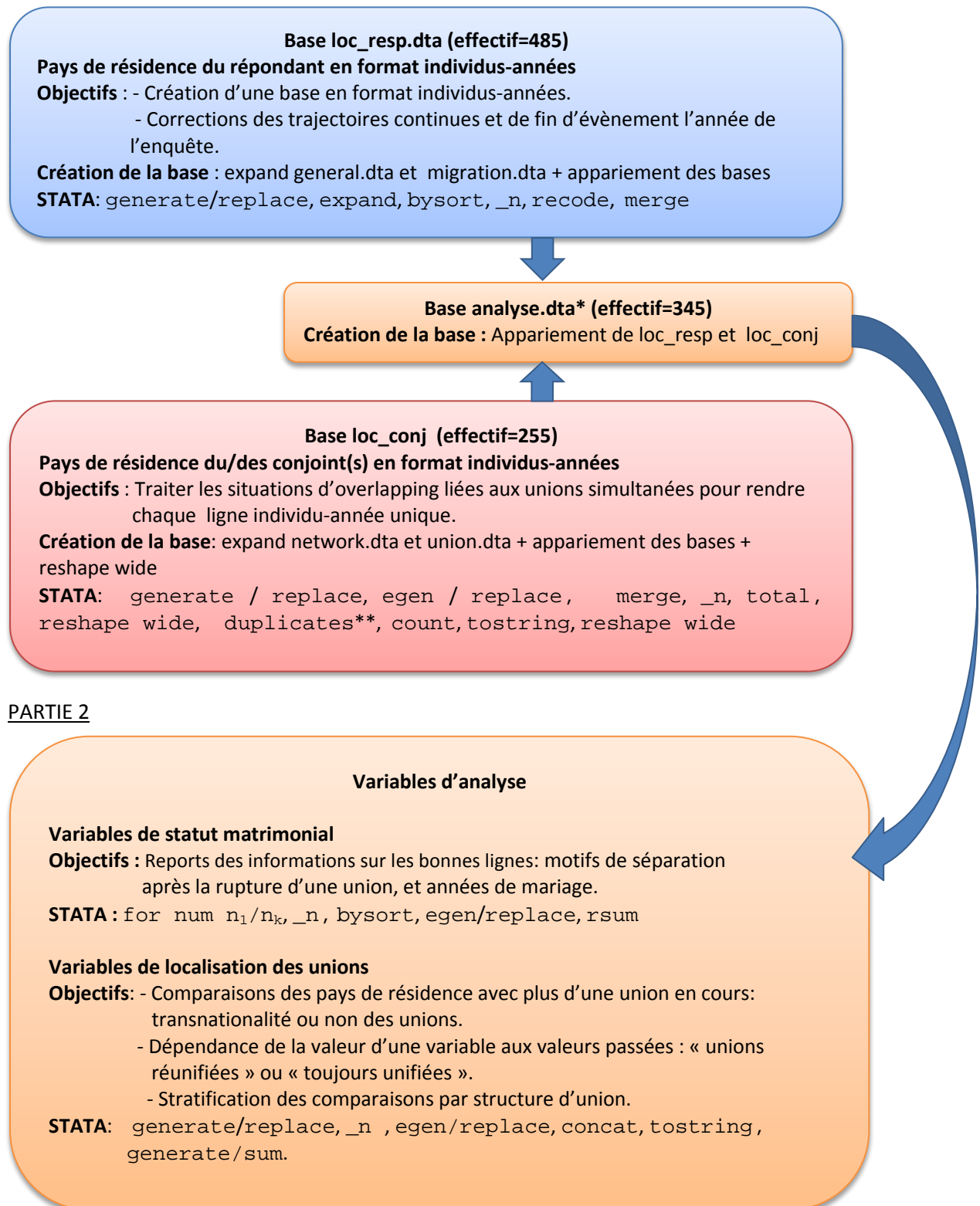
La logique des programmes **STATA** et **SAS** est quasi identique, la seule différence réside dans le report des informations pour la variable q103 (motifs de séparation). En annexe de ce document figure la transcription sous STATA de cette variante.

Avec **R** la logique du programme est différente. La majorité des variables créées ici dans la seconde partie, sont générées sous R dès la constitution de la base d'analyse (partie I) avant la correction de l'overlapping. En fin d'annexe, l'intégralité de ce programme est donnée avec quelques éléments d'explication.

¹ Diffusion prévue pour le début de l'année 2015

Les principales étapes et procédures STATA introduites

PARTIE 1



* Pour un âge supérieur ou égal à 14 ans.

** En annexe, dans les démarches alternatives.

PARTIE I : Création de la base d'analyse en format individus/années

1) Base listant toutes les années de vie du répondant jusqu' à l'année de l'enquête

Objectif :

Créer une ligne d'observation par année de vie

1. Création d'une variable durée (entre l'année de naissance et 2008, année de l'enquête).
2. « Allongement » de la base pour chaque individu : les observations vont être dupliquées avec l'instruction `expand`.

On utilisera ici la base *general.dta*, dont la structure est classique, avec autant de lignes que de répondants. Les deux autres options, à partir du module *union.dta* et du module à trajectoires strictement continues *house.dta* figurent en annexe.

Base « general »

variable name	storage type	display format	value label	variable label
ident	str7	%9s		Biographic individual identifier
q1a	int	%17.0g	Q1A	Year of birth

Dans un premier temps, on crée une variable de durée entre 2008 (année de l'enquête) et l'année de naissance (variable **q1a**). On doit ajouter 1 à la différence entre les deux années pour obtenir le nombre exact d'années de la naissance à 2008.

On verra plus tard que cet ajout génère des doublons lorsque la transition entre deux évènements a lieu la même année (trajectoire continue). Ces doublons devront faire l'objet d'une correction.

```
generate dur=2008-q1a+1
```

```
+-----+
|   ident   num_un~n   q1a   a2008   dur   |
+-----+-----+-----+-----+-----+
1. |     E1     1   1972   2008    37   |
2. |    E103    1   1978   2008    31   |
3. |    E147    1   1954   2008    55   |
4. |    F112    1   1951   2008    58   |
5. |    F121    1   1972   2008    37   |
+-----+-----+-----+-----+-----+
6. |    F127    1   1964   2008    45   |
7. |    I172    1   1971   2008    38   |
8. |  S011202   1   1975   2008    34   |
9. |  S079601   1   1936   2008    73   |
+-----+-----+-----+-----+-----+
```

Pour illustrer, on a créé la variable `a2008` qui est toujours égale à 2008

Pour transformer la base en individu-années (une ligne par année) on utilise l'instruction `expand` (<http://www.stata.com/manuals13/dexpand.pdf>). Cette instruction va répliquer la ligne un nombre de fois égal à la valeur de `dur`, soit ici le nombre d'années d'existence du répondant jusqu'en 2008 compris.

```
expand (dur)
```

Par individu (bysort ident), les variables **year** et **age** sont créées avec la variable de comptage **_n**. Il s'agit d'une variable système interne de Stata qui renvoie le numéro (le rang) de l'observation dans la base. Ici pour chaque individu, **_n=1,2,3.....,dur**. La variable **_n** est incrémentée à chaque changement de valeur de la (ou les) variable(s) indiquées en critères de tri (bysort).

Pour initialiser les années à partir de l'année de naissance de l'individu on remonte d'un rang avec **_n-1**, soit l'âge de l'individu au 31 décembre de chaque année.

```
bysort ident: generate year=q1a+(_n-1)
bysort ident: generate age=_n-1
```

Appariée aux modules transformés en format individu-années dont les trajectoires ne sont pas strictements continues, la base *year.dta* permet de récupérer toutes les années « sans évènement »

```
label variable year "Year"
label variable age "Age"
drop num_union q1a a2008
save "year.dta"
```

```
+-----+
| ident  year  age |
+-----+
38. | E103  1978   0 |
39. | E103  1979   1 |
40. | E103  1980   2 |
41. | E103  1981   3 |
42. | E103  1982   4 |
43. | E103  1983   5 |
44. | E103  1984   6 |
45. | E103  1985   7 |
46. | E103  1986   8 |
47. | E103  1987   9 |
48. | E103  1988  10 |
49. | E103  1989  11 |
50. | E103  1990  12 |
51. | E103  1991  13 |
52. | E103  1992  14 |
53. | E103  1993  15 |
54. | E103  1994  16 |
55. | E103  1995  17 |
56. | E103  1996  18 |
57. | E103  1997  19 |
58. | E103  1998  20 |
59. | E103  1999  21 |
60. | E103  2000  22 |
61. | E103  2001  23 |
62. | E103  2002  24 |
63. | E103  2003  25 |
64. | E103  2004  26 |
65. | E103  2005  27 |
66. | E103  2006  28 |
67. | E103  2007  29 |
+-----+
68. | E103  2008  30 |
+-----+
```


2) Base individus-années listant les pays de résidence du répondant

La démarche présentée ici, qui utilise la base *migration.dta*, n'est pas la plus rapide, mais elle exploite *year.dta*. Une option alternative avec la base *house.dta* est présentée en annexe.

Objectifs :

1. Rectifier la dernière année d'un évènement qui se termine l'année de l'enquête (p1)
2. Supprimer les doublons générés par les trajectoires continues (p2)

Base *migration.dta*

variable name	storage type	display format	value label	variable label
ident	str7	%9s		Biographic individual identifier
num_mig	byte	%8.0g		Migration number
q601d	int	%17.0g	Q601D	Start year of the stay outside Senegal
q601f	int	%17.0g	Q601F	End year of the stay outside Senegal
q602	float	%41.0g	COUNTRY	Country of stay

	ident	num_mig	q601d	q601f	q602	
1.	E1	1	1991	1992	NIGERIA	
2.	E1	2	2001	.	SPAIN	
3.	E103	1	1992	1995	SPAIN	
4.	E103	2	1997	.	SPAIN	
5.	E147	1	1994	1995	PORTUGAL	
6.	E147	2	1995	.	SPAIN	
7.	F112	1	1968	.	FRANCE	
8.	F121	1	1993	1995	FRANCE	
9.	F121	2	1998	2008	ITALY	(p2)
10.	F121	3	2008	.	FRANCE	
11.	F127	1	1989	1996	FRANCE	(p2)
12.	F127	2	1996	1997	UK	(p2)
13.	F127	3	1997	1997	PORTUGAL	(p2)
14.	F127	4	1997	1998	SERBIA & MTN	(p2)
15.	F127	5	1998	2000	HUNGARY	(p2)
16.	F127	6	2000	.	FRANCE	
17.	I172	1	1995	2004	ITALY	
18.	I172	2	2007	.	ITALY	
19.	S011202	1	2005	2008	SPAIN	(p1)
20.	S079601	1	1954	1956	INDONESIA	(p2)
21.	S079601	2	1956	1958	ALGERIA	
22.	S079601	3	1958	1961	MADAGASCAR	
23.	S079601	4	1961	1963	FRANCE	

Cas (p1) : Fin du dernier évènement l'année de l'enquête

Le temps étant discret avec comme unité l'année, le dernier évènement peut se terminer en 2008 avant la passation de l'entretien.

Si le dernier épisode de migration a eu lieu en 2008, on doit légèrement tronquer l'information pour créer la variable de durée en ramenant la fin de l'évènement à sa dernière année de survie au 31 décembre, soit l'année 2007. On génère tout d'abord la variable `end_08` qui permet d'identifier ces situations et de ne pas tronquer une seconde fois ces observations lorsqu'on corrige p2.

Pour les épisodes non achevés au moment de l'entretien, on remplace la valeur manquante de `q601f` par 2008.

```
generate end_08=1 if q601f==2008
replace q601f=2007 if q601f==2008
recode q601f (.=2008)
```

	ident	num_mig	q601d	q601f	q602	end_08
1.	E1	1	1991	1992	NIGERIA	.
2.	E1	2	2001	.	SPAIN	.
3.	E103	1	1992	1995	SPAIN	.
4.	E103	2	1997	.	SPAIN	.
5.	E147	1	1994	1995	PORTUGAL	.
6.	E147	2	1995	.	SPAIN	.
7.	F112	1	1968	.	FRANCE	.
8.	F121	1	1993	1995	FRANCE	.
9.	F121	2	1998	2007	ITALY	1
10.	F121	3	2008	.	FRANCE	.
11.	F127	1	1989	1996	FRANCE	.
12.	F127	2	1996	1997	UNITED KINGDOM	.
13.	F127	3	1997	1997	PORTUGAL	.
14.	F127	4	1997	1998	SERBIA AND MONTENEGRO	.
15.	F127	5	1998	2000	HUNGARY	.
16.	F127	6	2000	.	FRANCE	.
17.	I172	1	1995	2004	ITALY	.
18.	I172	2	2007	.	ITALY	.
19.	S011202	1	2005	2007	SPAIN	1
20.	S079601	1	1954	1956	INDONESIA	.
21.	S079601	2	1956	1958	ALGERIA	.
22.	S079601	3	1958	1961	MADAGASCAR	.
23.	S079601	4	1961	1963	FRANCE	.

```
expand (q601f-q601d+1)
bysort ident num_mig: generate year=q601d+_n-1
```

Cas (p2) : Suppression des doublons générés par les trajectoires continues

Une trajectoire est considérée comme continue si le début d'un évènement a lieu la même année que la fin de l'évènement précédent². Une même année se trouve donc inscrite sur deux lignes

² Ce type de transition est systématique dans les bases *activity.dta*, *house.dta* et *network.dta*, l'individu ayant toujours un statut d'activité et un pays de résidence.

différentes : une fois dans la variable **q601f** en (t-1) et une fois dans la variable **q601d** en (t). L'allongement de la base avec l'instruction `expand` provoque un doublon sur l'année.

Exemple pour `ident= S079601`

	ident	num_mig	q601d	q601f	q602	year
113.	S079601	1	1954	1956	INDONESIA	1954
114.	S079601	1	1954	1956	INDONESIA	1955
115.	S079601	1	1954	1956	INDONESIA	1956
116.	S079601	2	1956	1958	ALGERIA	1956
117.	S079601	2	1956	1958	ALGERIA	1957
118.	S079601	2	1956	1958	ALGERIA	1958
119.	S079601	3	1958	1961	MADAGASCAR	1958
120.	S079601	3	1958	1961	MADAGASCAR	1959
121.	S079601	3	1958	1961	MADAGASCAR	1960
122.	S079601	3	1958	1961	MADAGASCAR	1961
123.	S079601	4	1961	1963	FRANCE	1961
124.	S079601	4	1961	1963	FRANCE	1962
125.	S079601	4	1961	1963	FRANCE	1963

Ici on va de nouveau légèrement tronquer l'information en privilégiant l'année du nouvel évènement. On supprime donc le doublon généré par **q601f**. On considère donc que tous les évènements se sont terminés au 31 décembre de l'année précédente. Les deux conditions (`year !=2008` et `end_08!=1`) évitent de tronquer une seconde fois l'information lorsque le dernier évènement s'est achevé en 2008 (p1).

```
drop if (q601f==year) & year!=2008 & end_08!=1
drop end_08
```

	ident	num_mig	q601d	q601f	q602	year
106.	S079601	1	1954	1956	INDONESIA	1954
107.	S079601	1	1954	1956	INDONESIA	1955
108.	S079601	2	1956	1958	ALGERIA	1956
109.	S079601	2	1956	1958	ALGERIA	1957
110.	S079601	3	1958	1961	MADAGASCAR	1958
111.	S079601	3	1958	1961	MADAGASCAR	1959
112.	S079601	3	1958	1961	MADAGASCAR	1960
113.	S079601	4	1961	1963	FRANCE	1961
114.	S079601	4	1961	1963	FRANCE	1962

On remarque que lorsque la trajectoire continue a lieu en 2008 (F121), le problème a été réglé en amont lorsqu'on a remplacé la valeur de `q601f` par 2007.

Il ne reste plus alors qu'à apparier la base `migration.dta` avec la base `year.dta` créée plus haut, en triant préalablement les observations par l'identifiant individuel **ident** et la variable **year**.

```
sort ident year
save "loc_resp.dta"
use "year.dta"
sort ident year
merge 1:1 ident year using "loc_resp.dta"
drop _merge
```

```
rename q602 loc_resp
```

La variable renommée **loc_resp** (précédemment **q602**) a maintenant des valeurs manquantes pour les années n'appartenant pas à un épisode migratoire, c'est à dire les années où le répondant résidait au Sénégal. Il suffit de recoder la variable en remplaçant ces valeurs manquantes par le code pays du Sénégal (99341).

```
recode loc_resp (.=99341)  
keep ident year age loc_resp  
save "loc_resp.dta", replace
```

```
+-----+  
|   ident   year   age   loc_resp |  
+-----+  
291. | S079601  1936    0   SENEGAL |  
292. | S079601  1937    1   SENEGAL |  
293. | S079601  1938    2   SENEGAL |  
294. | S079601  1939    3   SENEGAL |  
295. | S079601  1940    4   SENEGAL |  
296. | S079601  1941    5   SENEGAL |  
297. | S079601  1942    6   SENEGAL |  
298. | S079601  1943    7   SENEGAL |  
299. | S079601  1944    8   SENEGAL |  
300. | S079601  1945    9   SENEGAL |  
+-----+  
301. | S079601  1946   10   SENEGAL |  
302. | S079601  1947   11   SENEGAL |  
303. | S079601  1948   12   SENEGAL |  
304. | S079601  1949   13   SENEGAL |  
305. | S079601  1950   14   SENEGAL |  
306. | S079601  1951   15   SENEGAL |  
307. | S079601  1952   16   SENEGAL |  
308. | S079601  1953   17   SENEGAL |  
309. | S079601  1954   18   INDONESIA |  
310. | S079601  1955   19   INDONESIA |  
311. | S079601  1956   20   ALGERIA |  
312. | S079601  1957   21   ALGERIA |  
313. | S079601  1958   22   MADAGASCAR |  
314. | S079601  1959   23   MADAGASCAR |  
315. | S079601  1960   24   MADAGASCAR |  
316. | S079601  1961   25     FRANCE |  
317. | S079601  1962   26     FRANCE |  
318. | S079601  1963   27   SENEGAL |  
319. | S079601  1964   28   SENEGAL |  
320. | S079601  1965   29   SENEGAL |  
321. | S079601  1966   30   SENEGAL |  
|   .       .       .       . |  
|   .       .       .       . |  
|   Idem    .       .       . |  
|   .       .       .       . |  
|   .       .       .       . |  
363. | S079601  2008   72   SENEGAL |  
+-----+
```

3) Base individus-années listant les pays de résidence des conjoints

Objectif

Transformation d'une base de données pour traiter les situations d'overlapping (polygamie) avec l'instruction `reshape` (option `wide`)

a) Pays de résidence des partenaires

Dans un premier temps on va récupérer les informations sur les pays de résidence des conjoints dans la base « network ». On fait comme dans la section précédente après avoir sélectionné les partenaires dans les personnes du réseau (`q901=1`).

Base *network.dta*

variable name	storage type	display format	value label	variable label
ident	str7	%9s		Biographic individual identifier
num_netmig	float	%9.0g		Number of migrant's migration
q904d	int	%17.0g	Q904	Beginning year of the migration
q904f	int	%17.0g	Q904	Ending year of the migration
country	float	%15.0g	COUNTRY	Country
q901	byte	%10.0g	Q901	Link with Ego
id_uni	str10	%10s		Union ID

	ident	num_netmig	q904d	q904f	country	q901	id_uni
1.	E103	1	1979	.	SPAIN	Partner	U_E1032
2.	E103	1	1989	.	SPAIN	Partner	U_E1031
3.	F112	1	1965	.	FRANCE	Partner	U_F1121
4.	I172	1	1986	1990	GUINEA	Partner	U_I1722
5.	I172	2	1990	1991	SENEGAL	Partner	U_I1722
6.	I172	3	1991	.	ITALY	Partner	U_I1722
7.	S011202	1	2000	2002	ITALY	Partner	U_S0112021
8.	S011202	2	2002	.	SPAIN	Partner	U_S0112021

Note : dans le tableau ci-dessus on a seulement sélectionné les informations relatives aux partenaires (`q901=1`)

```
use "network.dta", clear
keep if q901==1
generate end_08=1 if q904f==2008
replace q904f=2007 if q904f==2008
replace q904f=2008 if q904f==.
expand (q904f-q904d+1)
bysort ident id_uni num_netmig: generate year=q904d+_n-1
drop if (q904f==year) & year!=2008 & end_08!=1
keep ident id_uni country year
sort id_uni year
save "net_conj.dta"
```

b) Les périodes d'union du répondant

variable name	storage type	display format	value label	variable label
ident	str7	%9s		Biographic individual identifier
num_union	byte	%9.0g		Number of the union
id_uni	str10	%10s		Union ID
q102d	int	%7.0g	Q102D	Start years of the relationship
q102f	int	%7.0g	Q102F	End year of the relationship
q103	byte	%10.0g	Q103	Type of dissolution
q105	int	%8.0g	Q105	In which year did you get married
q107	float	%15.0g	COUNTRY	In which country was she/he borned

Avant de régler le problème de l'overlapping, on transforme la base *union.dta* en format individuus-années, et on l'apparie à la base *network.dta*.

```
generate end_08=1 if q102f==2008
replace q102f=2007 if q102f==2008
replace q102f=2008 if q102f==.
expand (q102f-q102d+1)
bysort ident num_union: generate year=q102d+_n-1
drop if (q102f==year) & year!=2008 & end_08!=1
keep ident num_union id_uni q103 q105 q107 year

sort id_uni year
merge 1:1 ident id_uni year using "net_conj.dta"
sort id_uni year
drop _merge id_uni end_08
drop if num_union==.
```

A ce stade, le pays de résidence du partenaire (country) avant la première migration se trouve en valeur manquante si le conjoint n'est pas né dans un des pays européens où l'enquête a été réalisée³. On utilise la variable **q107** (pays de naissance du conjoint) de la base *union.dta* pour récupérer cette information.

```
generate loc_conj=country
replace loc_conj=q107 if loc_conj==.
label value loc_conj COUNTRY
drop country q107
sort ident year
```

Le problème de l'overlapping

L'**overlapping** s'observe lorsqu'au moins deux évènements de même type se déroulent simultanément. Une situation classique d'overlapping est, par exemple, lorsqu'un individu a plusieurs enfants vivants en même temps. Ici, il s'agit des unions polygames.

³ Par exemple, dans MAFE Sénégal, ceux qui ne sont pas nés en Espagne, France ou Italie (se reporter à la description du module network dans le document « presentation of the dataset »).

	ident	num_un~n	id_uni	q102d	q102f	q103	q105	q107	
1.	E1	1	U_E11	1990	.	.	1993	SENEGAL	
2.	E103	1	U_E1031	1992	1996	Separation	1992	SENEGAL	
3.	E103	2	U_E1032	2001	2008	Separation	2001	SPAIN	
4.	E147	1	U_E1471	1979	.	.	1979	SENEGAL	
5.	E147	2	U_E1472	1990	.	.	1990	SENEGAL	Overlapping à partir
6.	E147	3	U_E1473	2004	.	.	.	SENEGAL	de 1990
7.	F112	1	U_F1121	1967	.	.	1968	SENEGAL	
8.	F121	1	U_F1211	1998	.	.	1998	SENEGAL	
9.	I172	1	U_I1721	1985	1993	Deceased	1987	SENEGAL	
10.	I172	2	U_I1722	2002	.	.	2004	SENEGAL	
11.	S011202	1	U_S0112021	1999	.	.	1999	SENEGAL	
12.	S079601	1	U_S0796011	1965	.	.	1965	SENEGAL	
13.	S079601	2	U_S0796012	1977	.	.	1977	SENEGAL	Overlapping à partir
14.	S079601	3	U_S0796013	1984	1988	Deceased	1984	SENEGAL	de 1977

Dans l'exemple ci-dessous, après l'allongement de la base, les années renseignées par la variable **year** sont uniques jusqu'en 1989 (une union), doublées de 1990 à 2003 (deux unions simultanées), puis triplées de 2004 à l'année de l'enquête (trois unions simultanées).

	ident	num_un~n	q103	q105	year	loc_conj
30.	E147	1	.	1979	1979	SENEGAL
31.	E147	1	.	1979	1980	SENEGAL
32.	E147	1	.	1979	1981	SENEGAL
33.	E147	1	.	1979	1982	SENEGAL
34.	E147	1	.	1979	1983	SENEGAL
35.	E147	1	.	1979	1984	SENEGAL
36.	E147	1	.	1979	1985	SENEGAL
37.	E147	1	.	1979	1986	SENEGAL
38.	E147	1	.	1979	1987	SENEGAL
39.	E147	1	.	1979	1988	SENEGAL
40.	E147	1	.	1979	1989	SENEGAL
41.	E147	1	.	1979	1990	SENEGAL
42.	E147	2	.	1990	1990	SENEGAL
43.	E147	1	.	1979	1991	SENEGAL
44.	E147	2	.	1990	1991	SENEGAL
45.	E147	1	.	1979	1992	SENEGAL
46.	E147	2	.	1990	1992	SENEGAL
47.	E147	1	.	1979	1993	SENEGAL
48.	E147	2	.	1990	1993	SENEGAL
49.	E147	1	.	1979	1994	SENEGAL
50.	E147	2	.	1990	1994	SENEGAL
51.	E147	1	.	1979	1995	SENEGAL
52.	E147	2	.	1990	1995	SENEGAL
53.	E147	1	.	1979	1996	SENEGAL
54.	E147	2	.	1990	1996	SENEGAL
55.	E147	1	.	1979	1997	SENEGAL
56.	E147	2	.	1990	1997	SENEGAL
57.	E147	1	.	1979	1998	SENEGAL
58.	E147	2	.	1990	1998	SENEGAL
.
.	.	Idem de 1999 à 2003
.
69.	E147	1	.	1979	2004	SENEGAL
70.	E147	2	.	1990	2004	SENEGAL
71.	E147	3	.	.	2004	SENEGAL
72.	E147	1	.	1979	2005	SENEGAL
73.	E147	2	.	1990	2005	SENEGAL
74.	E147	3	.	.	2005	SENEGAL
75.	E147	1	.	1979	2006	SENEGAL

76.	E147	2	.	1990	2006	SENEGAL
77.	E147	3	.	.	2006	SENEGAL
78.	E147	1	.	1979	2007	SENEGAL
79.	E147	2	.	1990	2007	SENEGAL
80.	E147	3	.	.	2007	SENEGAL
81.	E147	1	.	1979	2008	SENEGAL
82.	E147	2	.	1990	2008	SENEGAL
83.	E147	3	.	.	2008	SENEGAL

Pour résoudre ce problème, on va effectuer un `reshape` en largeur afin d'obtenir un bloc de variables pour chaque union, de sorte que chaque année apparaissent une seule fois pour chaque individu.

Les tableaux ci-dessous illustrent la démarche avec une variable propre à chaque évènement (X) qui ne varie pas au cours du temps (exemple : le sexe d'un individu).

De (t1 à (t4) l'évènement est unique ; de (t5) à (t9), il s'observe 2 fois simultanément ; et en (t10) suite à la fin de l'évènement 1, il est à nouveau unique.

Avant reshape

t	num_event	X
t1	1	v1
t2	1	v1
t3	1	v1
t3	1	v1
t4	1	v1
t5	1	v1
t5	2	v2
t6	1	v1
t6	2	v2
t7	1	v1
t7	2	v2
t8	1	v1
t8	2	v2
t9	1	v1
t9	2	v2
t10	2	v2



Après reshape

t	num_event_1	X_1	num_event_2	X_2
t1	1	v1	.	.
t2	1	v1	.	.
t3	1	v1	.	.
t3	1	v1	.	.
t4	1	v1	.	.
t5	1	v1	2	v2
t6	1	v1	2	v2
t7	1	v1	2	v2
t8	1	v1	2	v2
t9	1	v1	2	v2
t10	.	.	2	v2

Pour réaliser cette transformation de la base avec Stata, on utilise l'instruction `reshape wide` (http://www.stata.com/bookstore/stata12/pdf/d_reshape.pdf).

Etape 1 : comme la base est de type individus-années, on va créer un identifiant (**id**) unique combinant l'identifiant du répondant (**ident**) et l'année de l'observation (**year**). Il s'agit de la variable d'identification de l'instruction `reshape`, soit une ligne par individu-année. On transforme tout d'abord la variable `year` en format caractère en créant la variable **y** avec l'instruction `tostring`.

```
tostring year, generate(y)
generate id=ident+y
drop y
```

Etape2:

Pour corriger ces situations d'overlapping on va privilégier ici la solution la plus simple, soit l'élargissement de la base sur chaque séquence d'union (**num_union**).

Cela aura pour conséquence de créer également des blocs de variables pour les individus dont les unions se sont succédées sans se chevaucher (période union1=> rupture union1 +période d'au moins un an sans union => période union2.....). C'est avec cette option que la suite du tutoriel sera développée.

En annexe une variante est proposée, avec un reshape effectué seulement sur les individus avec au moins une situation d'overlapping. Si on choisit cette option, le programme de création des variables de la seconde partie doit être modifié.

Au préalable, deux variables vont être créées pour la seconde partie du tutoriel : la variable **nbu** qui renvoie pour chaque année le nombre total d'union en cours, et **num_uni** qui est identique à num_union, cette dernière disparaissant avec l'instruction reshape).

Pour créer des variables à partir de certaines fonctions, on utilise non pas l'instruction generate mais egen (« extension de generate »). La liste et les syntaxes des fonctions prises en charge par egen sont disponibles dans le document suivant: <http://www.stata.com/manuals13/degen.pdf>.

La variable **nbu** va être générée avec la fonction count.

```
bysort id: egen nbu=count(num_union)
label variable nbu "Number of unions"
generate num_uni=num_union
label variable num_uni "Number of the union"
```

L'ensemble des variables propres aux unions et donc susceptibles de varier d'une union à l'autre doivent entrer dans l'instruction reshape. Ce n'est pas le cas de la variable **nbu** qui compte seulement le nombre d'unions en cours pour chaque année et pour chaque individu.

```
sort id num_union
reshape wide q* loc_conj num_uni, i(id) j(num_union)
sort ident year
order ident year, first
drop id
sort ident year
save "loc_conj.dta"
```

En reprenant l'illustration précédente, le résultat obtenu après le reshape est le suivant (où **nbe** représente le nombre d'événements) :

ident	t	id	num_event_1	var_1	num_event_2	var_2	nbe
1	t1	1t1	1	v1	.	.	1
1	t2	1t2	1	v1	.	.	1
1	t3	1t3	1	v1	.	.	1
1	t3	1t4	1	v1	.	.	1
1	t4	1t4	1	v1	.	.	1
1	t5	1t5	1	v1	2	v2	2
1	t6	1t6	1	v1	2	v2	2
1	t7	1t7	1	v1	2	v2	2
1	t8	1t8	1	v1	2	v2	2
1	t9	1t9	1	v1	2	v2	2
1	t10	1t10	.	.	2	v2	1

Dans le cas de MAFE, Le résultat du reshape avec les variables de localisation des partenaires donne le résultat suivant:

	ident	year	loc_conj1	loc_conj2	loc_conj3	nbu
680.	E147	1979	SENEGAL	.	.	1
681.	E147	1980	SENEGAL	.	.	1
682.	E147	1981	SENEGAL	.	.	1
683.	E147	1982	SENEGAL	.	.	1
684.	E147	1983	SENEGAL	.	.	1
685.	E147	1984	SENEGAL	.	.	1
686.	E147	1985	SENEGAL	.	.	1
687.	E147	1986	SENEGAL	.	.	1
688.	E147	1987	SENEGAL	.	.	1
689.	E147	1988	SENEGAL	.	.	1
690.	E147	1989	SENEGAL	.	.	1
691.	E147	1990	SENEGAL	SENEGAL	.	2
692.	E147	1991	SENEGAL	SENEGAL	.	2
693.	E147	1992	SENEGAL	SENEGAL	.	2
694.	E147	1993	SENEGAL	SENEGAL	.	2
695.	E147	1994	SENEGAL	SENEGAL	.	2
696.	E147	1995	SENEGAL	SENEGAL	.	2
697.	E147	1996	SENEGAL	SENEGAL	.	2
698.	E147	1997	SENEGAL	SENEGAL	.	2
699.	E147	1998	SENEGAL	SENEGAL	.	2
700.	E147	1999	SENEGAL	SENEGAL	.	2
701.	E147	2000	SENEGAL	SENEGAL	.	2
702.	E147	2001	SENEGAL	SENEGAL	.	2
703.	E147	2002	SENEGAL	SENEGAL	.	2
704.	E147	2003	SENEGAL	SENEGAL	.	2
705.	E147	2004	SENEGAL	SENEGAL	SENEGAL	3
706.	E147	2005	SENEGAL	SENEGAL	SENEGAL	3
707.	E147	2006	SENEGAL	SENEGAL	SENEGAL	3
708.	E147	2007	SENEGAL	SENEGAL	SENEGAL	3
709.	E147	2008	SENEGAL	SENEGAL	SENEGAL	3

c) Appariement des bases répondants et conjoints

On peut enfin appairer les informations sur le(s) partenaire(s) à celles du répondant. Dans le cadre de ce tutoriel, on va limiter l'âge de l'individu à 14 ans qui correspond ici à l'âge le plus jeune pour la première union déclarée dans les observations sélectionnées.

```
use "loc_resp.dta", clear
sort ident year
merge 1:1 ident year using "loc_conj.dta"
drop _merge
drop if age<14
save "base_analyse.dta", replace
```

	ident	year	age	loc_resp	loc_conj1	loc_conj2	nbu
24.	E103	1992	14	SPAIN	SPAIN	.	1
25.	E103	1993	15	SPAIN	SPAIN	.	1
26.	E103	1994	16	SPAIN	SPAIN	.	1
27.	E103	1995	17	SENEGAL	SPAIN	.	1
28.	E103	1996	18	SENEGAL	.	.	.
29.	E103	1997	19	SPAIN	.	.	.
30.	E103	1998	20	SPAIN	.	.	.
31.	E103	1999	21	SPAIN	.	.	.
32.	E103	2000	22	SPAIN	.	.	.
33.	E103	2001	23	SPAIN	.	SPAIN	1
34.	E103	2002	24	SPAIN	.	SPAIN	1
35.	E103	2003	25	SPAIN	.	SPAIN	1

36.	E103	2004	26	SPAIN	.	SPAIN	1
37.	E103	2005	27	SPAIN	.	SPAIN	1
38.	E103	2006	28	SPAIN	.	SPAIN	1

39.	E103	2007	29	SPAIN	.	.	.
40.	E103	2008	30	SPAIN	.	.	.

Remarque : dans cet exemple il n'y a pas d'overlapping. Avec un reshape restreint (voir annexe), il n'y aurait pas d'élargissement et l'information sur les pays de résidence des partenaires serait seulement dans la variable loc_conj1. Les valeurs prises par la variable num_uni1 permettent alors de repérer le rang de chaque union.

PARTIE II. Exemples de création de variables d'analyse

L'objectif est de générer des variables de type TVC (Time Varying Covariate) sur le statut matrimonial du répondant et sur les localisations (pays de résidence) des différents partenaires en distinguant les unions « transnationale », « réunifiée » et « toujours unifiée ». L'intérêt des deux dernières modalités est que les valeurs dépendent de celles prises dans le passé.

1) Création des variables de statut matrimonial

Objectif

Reports des informations sur les bonnes lignes individu-année: motifs de séparations après la rupture d'une union, et années de mariage

a) Les motifs et les années de séparation

Suite à l'expand, l'information sur la rupture de l'union n'est reportée que sur les lignes correspondant aux années où l'union était en cours.

	ident	year	q1031	num_uni1	q1032	num_uni2
149.	F127	1978
150.	F127	1979
151.	F127	1980
152.	F127	1981
153.	F127	1982
154.	F127	1983
155.	F127	1984	Separation or divorce	1	.	.
156.	F127	1985	Separation or divorce	1	.	.
157.	F127	1986	Separation or divorce	1	.	.
158.	F127	1987	Separation or divorce	1	.	.
159.	F127	1988	Separation or divorce	1	.	.
160.	F127	1989	.	.	Separation or divorce	2
161.	F127	1990	.	.	Separation or divorce	2
162.	F127	1991	.	.	Separation or divorce	2
163.	F127	1992
164.	F127	1993
165.	F127	1994
166.	F127	1995
167.	F127	1996
168.	F127	1997
169.	F127	1998
170.	F127	1999
171.	F127	2000
172.	F127	2001
173.	F127	2002
174.	F127	2003
175.	F127	2004
176.	F127	2005
177.	F127	2006
178.	F127	2007
179.	F127	2008

Deux options sont envisageables :

- On peut renseigner toutes les lignes jusqu'au changement de répondant même si une nouvelle union apparaît.

- On peut se contenter d'avoir l'information sur les séparations lorsque le répondant ne s'était pas remis en couple. Mais avec cette dernière option, on perd l'information sur la rupture de l'union (séparation, décès du partenaire) en cas de trajectoire continue ou d'overlapping.

On a choisi de récupérer l'information sur la rupture de l'union jusqu'au changement de répondant (première ligne de codes), puis la supprimer les années où l'union était en cours (seconde ligne de codes).

Pour sa simplicité d'utilisation, la commande `for num n1/nk` : est particulièrement adaptée à la manipulation des blocs de variables générées après un `reshape wide`. L'instruction qui suit `for num n1/nk` : est effectuée k fois pour chaque valeur allant de n₁ à n_k. Dans cette instruction principale, la lettre X renvoie au numéro d'occurrence de chaque variable⁴.

```
for num 1/5: replace q103X=q103X[_n-1] if num_uniX==. & ident[_n]==ident[_n-1]
for num 1/5: replace q103X=. if num_uniX!=.
```

Dans la première ligne, `q103X[_n-1]` donne la valeur à récupérer, soit la valeur de q103X l'année précédente ([_n-1]).

Pour la première union de l'exemple, la première valeur remplacée est égale à q1031 en 1989 = q1031[en 1988], la seconde q1031 en 1990= q1031[en 1989] etc.....

La condition `ident[_n]==ident[_n-1]` limite la substitution des valeurs à un même répondant, car l'instruction `for num n1/nk` : ne peut être associée à `by sort`.

Effet de la première ligne:

	ident	year	q1031	num_uni1	q1032	num_uni2
149.	F127	1978
150.	F127	1979
151.	F127	1980
152.	F127	1981
153.	F127	1982
154.	F127	1983
155.	F127	1984	Separation or divorce	1	.	.
156.	F127	1985	Separation or divorce	1	.	.
157.	F127	1986	Separation or divorce	1	.	.
158.	F127	1987	Separation or divorce	1	.	.
159.	F127	1988	Separation or divorce	1	.	.
160.	F127	1989	Separation or divorce	.	Separation or divorce	2
161.	F127	1990	Separation or divorce	.	Separation or divorce	2
162.	F127	1991	Separation or divorce	.	Separation or divorce	2
163.	F127	1992	Separation or divorce	.	Separation or divorce	.
164.	F127	1993	Separation or divorce	.	Separation or divorce	.
165.	F127	1994	Separation or divorce	.	Separation or divorce	.
166.	F127	1995	Separation or divorce	.	Separation or divorce	.
167.	F127	1996	Separation or divorce	.	Separation or divorce	.
168.	F127	1997	Separation or divorce	.	Separation or divorce	.
169.	F127	1998	Separation or divorce	.	Separation or divorce	.
170.	F127	1999	Separation or divorce	.	Separation or divorce	.
171.	F127	2000	Separation or divorce	.	Separation or divorce	.
172.	F127	2001	Separation or divorce	.	Separation or divorce	.
173.	F127	2002	Separation or divorce	.	Separation or divorce	.

⁴ Si on utilisait la syntaxe des boucles de Stata, on aurait une expression très proche : `for num` serait remplacé par `foreach`, et dans l'instruction principale X par `i` :

```
forvalue i=1/5 {
  replace q103`i'=q103`i'[_n-1] if num_uni`i'==. & ident[_n]==ident[_n-1]
  replace q103`i'=. if num_uni`i'!=.
}
```

174.	F127	2003	Separation or divorce	.	Separation or divorce	.
175.	F127	2004	Separation or divorce	.	Separation or divorce	.
176.	F127	2005	Separation or divorce	.	Separation or divorce	.
177.	F127	2006	Separation or divorce	.	Separation or divorce	.
178.	F127	2007	Separation or divorce	.	Separation or divorce	.
179.	F127	2008	Separation or divorce	.	Separation or divorce	.

La seconde ligne supprime les informations sur les séparations les années de l'union. Si on ne les supprime pas, on devra ajouter une condition (avec num_uniX) pour générer les variables dichotomiques liées au statut matrimonial qui vont être créées dans la section suivante.

Effet de la seconde ligne :

	ident	year	q1031	num_uni1	q1032	num_uni2
149.	F127	1978
150.	F127	1979
151.	F127	1980
152.	F127	1981
153.	F127	1982
154.	F127	1983
155.	F127	1984	.	1	.	.
156.	F127	1985	.	1	.	.
157.	F127	1986	.	1	.	.
158.	F127	1987	.	1	.	.
159.	F127	1988	.	1	.	.
160.	F127	1989	Separation or divorce	.	.	2
161.	F127	1990	Separation or divorce	.	.	2
162.	F127	1991	Separation or divorce	.	.	2
163.	F127	1992	Separation or divorce	.	Separation or divorce	.
164.	F127	1993	Separation or divorce	.	Separation or divorce	.
165.	F127	1994	Separation or divorce	.	Separation or divorce	.
166.	F127	1995	Separation or divorce	.	Separation or divorce	.
167.	F127	1996	Separation or divorce	.	Separation or divorce	.
168.	F127	1997	Separation or divorce	.	Separation or divorce	.
169.	F127	1998	Separation or divorce	.	Separation or divorce	.
170.	F127	1999	Separation or divorce	.	Separation or divorce	.
171.	F127	2000	Separation or divorce	.	Separation or divorce	.
172.	F127	2001	Separation or divorce	.	Separation or divorce	.
173.	F127	2002	Separation or divorce	.	Separation or divorce	.
174.	F127	2003	Separation or divorce	.	Separation or divorce	.
175.	F127	2004	Separation or divorce	.	Separation or divorce	.
176.	F127	2005	Separation or divorce	.	Separation or divorce	.
177.	F127	2006	Separation or divorce	.	Separation or divorce	.
178.	F127	2007	Separation or divorce	.	Separation or divorce	.
179.	F127	2008	Separation or divorce	.	Separation or divorce	.

On aurait pu régler ce problème en amont dès la création de la base d'analyse en créant une base individus-événements liées aux séparations à partir de la base *union.dta*. Cette option, plus longue, est donnée en annexe.

b) Les années de mariages

Sur le même principe, l'information sur le mariage (**q105**) est aussi présente sur les lignes où l'union n'était encore qu'informelle. Comme pour les séparations, on choisit de la supprimer les années où l'union n'était qu'informelle, et pour la forme on remplace l'année par la modalité 1 associée au label

« mariage ». Il suffit de comparer l'année du mariage (q105X) à l'année de l'observation. Si la valeur de **q105** est inférieure à celle de **year**, cela signifie que l'union est informelle.

	ident	year	num_uni1	q1051
5.	E1	1990	1	1993
6.	E1	1991	1	1993
7.	E1	1992	1	1993
8.	E1	1993	1	1993
9.	E1	1994	1	1993
10.	E1	1995	1	1993
	.			
	.			
	.	idem		
	.			
20.	E1	2008	1	1993

```

for num 1/5: replace q105X=. if q105X>year & num_uniX!=. & ident[_n]==ident[_n+1]
for num 1/5: replace q105X=1 if q105X!=.

label define Q105 1 "Marriage", modify
for num 1/5: label value q105X Q105

```

	ident	year	num_un~1	q1051
5.	E1	1990	1	.
6.	E1	1991	1	.
7.	E1	1992	1	.
8.	E1	1993	1	Marriage
9.	E1	1994	1	Marriage
10.	E1	1995	1	Marriage
	.			
	.			
	.	idem		
	.			
20.	E1	2008	1	Marriage

c) Variables dichotomiques de statut matrimonial

Les variables **C**, **V**, **D**, **U**, **M** indiquent si l'individu est –pour chaque année donnée– célibataire, veuf, séparé ou divorcé, en union informelle, marié. Si C ne peut être égale à 1 qu'avant la première union, les autres situations peuvent se chevaucher : un individu peut être marié (M=1) et divorcé ou veuf d'une union précédente (D=1 ou V=1).

Sauf pour C qui est générée directement, on récupère dans un premier temps l'information des différents statuts pour chaque union (1 si oui). Ensuite, on additionne les valeurs obtenus avec la fonction `rsum` (row sum) qui, associée à l'instruction `egen`, permet de sommer pour chaque ligne les valeurs de plusieurs vecteurs colonnes. Si le résultat est supérieur à 0 alors l'individu est dans le statut en question.

```

generate C=1 if num_uni1==. & q1031==.
label variable C "Alone, never in partnership =1"

for num 1/5: generate V_X=1 if q103X==2
egen v=rsum(V_*)
generate V=1 if v>0 & v<.
drop V_* v
label variable V "Widowed =1"

for num 1/5: generate D_X=1 if q103X==1
egen d=rsum(V_*)
generate D=1 if d>0 & d<.
drop D_* d
label variable D "Divorced or separated =1"

for num 1/5: generate U_X=1 if num_uniX!=. & q105X==.
egen u=rsum(U_*)
generate U=1 if u>0 & u<.
drop U_* u
label variable U "Partnership, not married=1"

for num 1/5: generate M_X=1 if num_uniX!=. & q105X!=.
egen m=rsum(M_*)
generate M=1 if m>0 & m<.
drop M_* m
label variable M "Married=1"

```

	ident	year	q1031	q1051	num_uni1	q1032	q1052	num_uni2	C	V	D	U	M
180.	I172	1985	.	.	1	.	.	.	0	0	0	1	0
181.	I172	1986	.	.	1	.	.	.	0	0	0	1	0
182.	I172	1987	.	Marriage	1	.	.	.	0	0	0	0	1
183.	I172	1988	.	Marriage	1	.	.	.	0	0	0	0	1
184.	I172	1989	.	Marriage	1	.	.	.	0	0	0	0	1
185.	I172	1990	.	Marriage	1	.	.	.	0	0	0	0	1
186.	I172	1991	.	Marriage	1	.	.	.	0	0	0	0	1
187.	I172	1992	.	Marriage	1	.	.	.	0	0	0	0	1
188.	I172	1993	Partner deceased	0	1	0	0	0
189.	I172	1994	Partner deceased	0	1	0	0	0
190.	I172	1995	Partner deceased	0	1	0	0	0
191.	I172	1996	Partner deceased	0	1	0	0	0
192.	I172	1997	Partner deceased	0	1	0	0	0
193.	I172	1998	Partner deceased	0	1	0	0	0
194.	I172	1999	Partner deceased	0	1	0	0	0
195.	I172	2000	Partner deceased	0	1	0	0	0
196.	I172	2001	Partner deceased	0	1	0	0	0
197.	I172	2002	Partner deceased	2	0	1	0	1	0
198.	I172	2003	Partner deceased	2	0	1	0	1	0
199.	I172	2004	Partner deceased	.	.	Marriage	.	2	0	1	0	0	1
200.	I172	2005	Partner deceased	.	.	Marriage	.	2	0	1	0	0	1
201.	I172	2006	Partner deceased	.	.	Marriage	.	2	0	1	0	0	1
202.	I172	2007	Partner deceased	.	.	Marriage	.	2	0	1	0	0	1
203.	I172	2008	Partner deceased	.	.	Marriage	.	2	0	1	0	0	1

d) Variable synthétique de statut matrimonial

A partir de ces 5 nouvelles variables, on peut synthétiser l'information en créant la variable **marital** variable qui chaque année donne une information sur le statut matrimonial du répondant. L'ensemble des modalités créées ici n'est qu'une possibilité parmi d'autres.


```

generate marital=.
order marital, after(M)
replace marital=1 if C==1
replace marital=2 if D==1 & nbu==.
replace marital=3 if V==1 & nbu==.
replace marital=4 if U==1 & nbu==1
replace marital=5 if M==1 & nbu==1
replace marital=6 if U==1 & nbu>1 & nbu<. & M==0
replace marital=7 if M==1 & nbu>1 & nbu<.

label define mar_stat 1 "Alone, never in union", replace
label define mar_stat 2 "Alone, separated", modify
label define mar_stat 3 "Alone, widowed", modify
label define mar_stat 4 "Monogamy, informal union", modify
label define mar_stat 5 "Monogamy, married", modify
label define mar_stat 6 "Polygamy, only informal unions", modify
label define mar_stat 7 "Polygamy, at least one marriage", modify
label value marital mar_stat

```

	ident	year	num_uni1	q1031	q1051	nbu	C	V	D	U	M	marital
82.	F112	1965	1	0	0	0	0	Alone, never in union
83.	F112	1966	1	0	0	0	0	Alone, never in union
84.	F112	1967	1	.	.	1	0	0	0	1	0	Monogamy, informal union
85.	F112	1968	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
86.	F112	1969	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
87.	F112	1970	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
88.	F112	1971	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
89.	F112	1972	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
90.	F112	1973	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
91.	F112	1974	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
92.	F112	1975	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
93.	F112	1976	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
94.	F112	1977	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
95.	F112	1978	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
96.	F112	1979	1	.	Marriage	1	0	0	0	0	1	Monogamy, married
.
.
.	etc
.
.
125.	F112	2008	1	.	Marriage	1	0	0	0	0	1	Monogamy, married

marital	Freq.	Percent	Cum.
Alone, never in union	75	21.74	21.74
Alone, separated	6	1.74	23.48
Alone, widowed	9	2.61	26.09
Monogamy, informal union	24	6.96	33.04
Monogamy, married	132	38.26	71.30
Polygamy, only informal unions	9	2.61	73.91
Polygamy, at least one marriage	90	26.09	100.00
Total	345	100.00	

2) Variable de comparaison des pays de résidence des répondants et des partenaires

Objectifs

- Comparaisons des pays de résidence avec unions simultanées: transnationalité ou non des unions.
- Dépendance de la valeur d'une variable aux valeurs passées : « unions réunifiées » ou « toujours unifiées ».
- Stratification des comparaisons par structure d'union.

a) Comparaison des codes pays entre le répondant et le(les) partenaires

La nouvelle variable **uni_loc** prend la valeur 1 lorsque le répondant et se(s) partenaire(s) vivent dans le même pays, 0 sinon. Comme on ne peut pas ici intégrer `for num n1/nk`: à `generate` qui ne peut être appelée qu'une seule fois pour chaque nouvelle variable créée, on procède en deux étapes. Dans un premier temps, la variable est générée avec la première union seulement en comparant **loc_resp** à **loc_conj1**, et on associe ensuite `for num` à l'instruction `replace` pour les unions de rang supérieur.

```
generate uni_loc=1 if loc_resp==loc_conj1 & nbu>0 & nbu<.  
for num 2/5: replace uni_loc=1 if loc_resp==loc_conjX & nbu>0 & nbu<. & loc_conjX!=.  
for num 1/5: replace uni_loc=0 if loc_resp!=loc_conjX & nbu>0 & nbu<. & loc_conjX!=.  
  
label variable fam_loc "Respondant & partner(s) in the same country = 1"generate  
  
uni=1 if nbu!=.  
label variable uni "Respondant in union = 1"  
  
saveold "base_analyse.dta", replace
```

b) Création des variables synthétiques de localisation

b1. Exemple avec un répondant avec une seule union

Pour montrer la démarche sur un cas simple sans union simultanée, on va sélectionner un seul répondant

```
keep if ident=="E1"  
keep ident year age uni_loc loc_resp loc_conj1 uni  
save "E1.dta"
```

Comptage des années d'union et des années dans le même pays

Pour créer les modalités de localisation «always unified » et « reunified », qui dépendent des valeurs prises dans le passé par **uni_loc**, on va compter le nombre d'années passées dans l'union (**cuni**) et le nombre d'années où les partenaires vivaient dans le même pays (**cuni_loc**), et comparer les valeurs de ces deux variables. Lorsque la valeur de **cuni_loc** est inférieure à celle de **cuni**, cela indique que les partenaires n'ont pas toujours vécu dans le même pays (transnationalité). Pour faire ces comptages on associe la fonction `sum` (somme en cumul) à la commande `generate` (et non `egen`).

```
sort ident year
by ident: generate cuni=sum(uni)          if uni==1
by ident: generate cuni_loc=sum(uni_loc) if uni==1
sort ident year
```

Création de la TVC

Pour une année donnée, les différentes situations sont donc les suivantes:

- Si **uni = 0**, le répondant n'est pas en couple.
- Si **uni_loc = 0**, les partenaires de l'union ne vivent pas dans le même pays. Il y a transnationalité du couple.
- Si **uni_loc =1** et **cuni>cuni_loc**, les deux partenaires vivent tous les deux dans le même pays l'année t, mais cela n'a pas été toujours le cas (nombre d'années de l'union>nombre d'années où les deux personnes vivaient dans le même pays). Il y a réunification du couple.
- Si **uni_loc =1** et **cuni=cuni_loc**, les partenaires de l'union ont toujours vécu dans le même pays.

```
generate trans=0 if uni==0
replace trans=1 if uni_loc==0
replace trans=2 if uni_loc==1 & cuni>cuni_loc
replace trans=3 if uni_loc==1
& cuni==cuni_loc

label define TRANS 0 "No partner/spouse", modify
label define TRANS 1 "Transnational", modify
label define TRANS 2 "Reunified", modify
label define TRANS 3 "Always unified", modify
label value trans1 TRANS
```

Dans cet exemple, les deux partenaires ont vécu dans le même pays en 1990 et de 1992 à 2000. Les sommes cumulées de **cuni** et **cuni_loc** ne sont égales que pour la première année de l'union. Il s'en est suivi une année de transnationalité puis 9 années de réunification, le couple étant de nouveau transnational de 2001 jusqu'au moment de l'enquête en 2008.

	ident	year	age	uni_loc	uni	cuni	cuni_loc	
1.	E1	1990	18	1	1	1	1	Always unified
2.	E1	1991	19	0	1	2	1	Transnational
3.	E1	1992	20	1	1	3	2	Reunified
4.	E1	1993	21	1	1	4	3	Reunified
5.	E1	1994	22	1	1	5	4	Reunified
6.	E1	1995	23	1	1	6	5	Reunified
7.	E1	1996	24	1	1	7	6	Reunified
8.	E1	1997	25	1	1	8	7	Reunified

que le répondant et la partenaire 2 (num_union=2) ont toujours vécu dans le même pays, la transnationalité de la structure [23] en t11 et t12 vient du fait que la partenaire 3 résidait dans un autre pays que le répondant. Après la rupture de cette dernière union, on retrouve la structure [2] qui est ici de nouveau considérée comme « always unified ».

ident	t	num_uni_1	num_uni_2	num_uni_3	suni_id	cuni	cuni_loc	trans
1	t1	alone
1	t2	1	.	.	1	1	1	always unified
1	t3	1	.	.	1	2	1	transnational
1	t3	1	.	.	1	3	2	reunified
1	t4	1	2	.	12	1	1	always unified
1	t5	1	2	.	12	2	2	always unified
1	t6	1	2	.	12	3	3	always unified
1	t7	1	2	.	12	4	4	always unified
1	t8	1	2	.	12	5	5	always unified
1	t9	1	2	.	12	6	6	always unified
1	t10	.	2	.	2	1	1	always unified
1	t11	.	2	3	23	1	0	transnational
1	t12	.	2	3	23	2	0	transnational
1	t13	.	2	.	2	2	2	always unified

Le format de la variable d'identification de la structure d'union doit être de type caractère sans quoi l'on risquerait d'avoir la même valeur de **suni_id** pour deux structures différentes. Par exemple, avec les unions simultanées 1 et 2, on obtiendrait le même id que l'union unique 3 avec un format numérique. Pour créer cette variable, on utilise alors la fonction `concat` associée à l'instruction `egen`.

Identifiant de la structure d'union

```
use "loc_resp.dta", clear
for num 1/5: tostring num_uniX, generate (x_X)
egen suni_id=concat(x_*)
replace suni_id="" if suni_id == "....."
drop x_*
order suni_id, after(nbu)
```

Ensuite, la TVC est construite comme en b1), à ceci près que les variables **cuni** et **cuni_loc** sont stratifiées par l'identifiant de la structure d'union **suni_id**. On ajoute également une modalité à la variable **trans**, celle où l'individu n'est pas en union (nbu=).

```
sort ident suni_id year
by ident suni_id: generate cuni=sum(uni) if uni==1
by ident suni_id: generate cuni_loc=sum(uni_loc) if uni==1
sort ident year

generate trans =0 if uni==.
replace trans =1 if uni_loc==0
replace trans =2 if uni_loc==1 & cuni>cuni_loc
replace trans =3 if uni_loc==1 & cuni==cuni_loc

label define TRANS 0 "No partner/spouse", modify
label define TRANS 1 "Transnational", modify
label define TRANS 2 "Reunified", modify
label define TRANS 3 "Always unified", modify
label value trans TRANS
drop cuni cuni_loc
```

```
tabulate trans
```

Marital & Geographic situation each year	Freq.	Percent	Cum.
No partner/spouse	90	26.09	26.09
Transnational	54	15.65	41.74
Reunified	57	16.52	58.26
Always unified	144	41.74	100.00
Total	345	100.00	

Pour avoir plus de détails au niveau du statut matrimonial, on peut utiliser la variable **marital**. Ici on a juste distingué les épisodes « jamais en union », des « séparations et veuvage ».

```
generate trans2=0 if trans==0 & C==1
replace trans2=1 if trans==0 & (D==1 | V==1)
replace trans2=2 if trans==1
replace trans2=3 if trans==2
replace trans2=4 if trans==3

label define TRANS2 0 "Never in union", modify
label define TRANS2 1 "Separated or widowed", modify
label define TRANS2 2 "Transnational", modify
label define TRANS2 3 "Reunified", modify
label define TRANS2 4 "Always unified ", modify
label value trans2 TRANS2
```

Marital & Geographic situation each year	Freq.	Percent	Cum.
2			
Never in union	75	21.74	21.74
Separated or widowed	15	4.35	26.09
Transnational	54	15.65	41.74
Reunified	57	16.52	58.26
Always unified	144	41.74	100.00
Total	345	100.00	

Extrait de la base trans entre 1950 et 2008 (tous les 5 ans)

	ident	year	age	id_hh	uni	uni_loc	cuni	cuni_loc	trans2
5.	E1	1990	18	1....	1	1	1	1	Always unified
10.	E1	1995	23	1....	1	1	6	5	Reunified
15.	E1	2000	28	1....	1	1	11	10	Reunified
20.	E1	2005	33	1....	1	0	16	10	Transnational
23.	E1	2008	36	1....	1	0	19	10	Transnational
27.	E103	1995	17	1....	1	0	4	3	Transnational
32.	E103	2000	22	Separated or widowed
37.	E103	2005	27	.2...	1	0	5	0	Transnational
40.	E103	2008	30	Separated or widowed
43.	E147	1970	16	Never in union
48.	E147	1975	21	Never in union
53.	E147	1980	26	1....	1	1	2	2	Always unified
58.	E147	1985	31	1....	1	1	7	7	Always unified
63.	E147	1990	36	12...	1	1	1	1	Always unified
68.	E147	1995	41	12...	1	0	6	4	Transnational
73.	E147	2000	46	12...	1	0	11	4	Transnational
78.	E147	2005	51	123..	1	0	2	0	Transnational
81.	E147	2008	54	123..	1	0	5	0	Transnational
82.	F112	1965	14	Never in union
87.	F112	1970	19	1....	1	1	4	3	Reunified
92.	F112	1975	24	1....	1	1	9	8	Reunified
97.	F112	1980	29	1....	1	1	14	13	Reunified
102.	F112	1985	34	1....	1	1	19	18	Reunified
107.	F112	1990	39	1....	1	1	24	23	Reunified
112.	F112	1995	44	1....	1	1	29	28	Reunified
117.	F112	2000	49	1....	1	1	34	33	Reunified
122.	F112	2005	54	1....	1	1	39	38	Reunified
125.	F112	2008	57	1....	1	1	42	41	Reunified
130.	F121	1990	18	Never in union
135.	F121	1995	23	Never in union
140.	F121	2000	28	1....	1	0	3	0	Transnational
145.	F121	2005	33	1....	1	0	8	0	Transnational
148.	F121	2008	36	1....	1	0	11	1	Transnational
151.	F127	1980	16	Never in union
156.	F127	1985	21	1....	1	1	2	2	Always unified
161.	F127	1990	26	.2...	1	0	2	0	Transnational
166.	F127	1995	31	..3..	1	0	2	0	Transnational
171.	F127	2000	36	..3..	1	0	7	0	Transnational
176.	F127	2005	41	..3.5	1	1	4	4	Always unified
179.	F127	2008	44	..3.5	1	1	7	7	Always unified
180.	I172	1985	14	1....	1	1	1	1	Always unified
185.	I172	1990	19	1....	1	1	6	6	Always unified
190.	I172	1995	24	Separated or widowed
195.	I172	2000	29	Separated or widowed
200.	I172	2005	34	.2...	1	0	4	0	Transnational
203.	I172	2008	37	.2...	1	0	7	0	Transnational
205.	S011202	1990	15	Never in union
210.	S011202	1995	20	Never in union
215.	S011202	2000	25	1....	1	0	2	1	Transnational
220.	S011202	2005	30	1....	1	1	7	2	Reunified
223.	S011202	2008	33	1....	1	0	10	3	Transnational
224.	S079601	1950	14	Never in union
229.	S079601	1955	19	Never in union
234.	S079601	1960	24	Never in union
239.	S079601	1965	29	1....	1	1	1	1	Always unified
244.	S079601	1970	34	1....	1	1	6	6	Always unified
249.	S079601	1975	39	1....	1	1	11	11	Always unified
254.	S079601	1980	44	12...	1	1	4	4	Always unified
259.	S079601	1985	49	123..	1	1	2	2	Always unified
264.	S079601	1990	54	12...	1	1	10	10	Always unified
269.	S079601	1995	59	12...	1	1	15	15	Always unified
274.	S079601	2000	64	12...	1	1	20	20	Always unified
279.	S079601	2005	69	12...	1	1	25	25	Always unified
282.	S079601	2008	72	12...	1	1	28	28	Always unified
287.	S093701	1950	18	Never in union
292.	S093701	1955	23	Never in union
297.	S093701	1960	28	Never in union
302.	S093701	1965	33	1....	1	1	5	5	Always unified
307.	S093701	1970	38	.2...	1	0	1	0	Transnational
312.	S093701	1975	43	.234.	1	1	1	1	Always unified

317.	S093701	1980	48	.2345	1	1	3	3	Always unified
322.	S093701	1985	53	.2345	1	1	8	8	Always unified
327.	S093701	1990	58	.2345	1	1	13	13	Always unified
332.	S093701	1995	63	.2345	1	1	18	18	Always unified
337.	S093701	2000	68	.2345	1	1	23	23	Always unified
342.	S093701	2005	73	.2345	1	0	28	27	Transnational
345.	S093701	2008	76	.2345	1	0	31	27	Transnational

-----+

Extrait de la base trans entre 1985 et 1995

	ident	year	age	id_hh	uni	uni_loc	cuni	cuni_loc	trans2
1.	E1	1986	14	Never in union
2.	E1	1987	15	Never in union
3.	E1	1988	16	Never in union
4.	E1	1989	17	Never in union
5.	E1	1990	18	1....	1	1	1	1	Always unified
6.	E1	1991	19	1....	1	0	2	1	Transnational
7.	E1	1992	20	1....	1	1	3	2	Reunified
8.	E1	1993	21	1....	1	1	4	3	Reunified
9.	E1	1994	22	1....	1	1	5	4	Reunified
10.	E1	1995	23	1....	1	1	6	5	Reunified
24.	E103	1992	14	1....	1	1	1	1	Always unified
25.	E103	1993	15	1....	1	1	2	2	Always unified
26.	E103	1994	16	1....	1	1	3	3	Always unified
27.	E103	1995	17	1....	1	0	4	3	Transnational
58.	E147	1985	31	1....	1	1	7	7	Always unified
59.	E147	1986	32	1....	1	1	8	8	Always unified
60.	E147	1987	33	1....	1	1	9	9	Always unified
61.	E147	1988	34	1....	1	1	10	10	Always unified
62.	E147	1989	35	1....	1	1	11	11	Always unified
63.	E147	1990	36	12...	1	1	1	1	Always unified
64.	E147	1991	37	12...	1	1	2	2	Always unified
65.	E147	1992	38	12...	1	1	3	3	Always unified
66.	E147	1993	39	12...	1	1	4	4	Always unified
67.	E147	1994	40	12...	1	0	5	4	Transnational
68.	E147	1995	41	12...	1	0	6	4	Transnational
102.	F112	1985	34	1....	1	1	19	18	Reunified
103.	F112	1986	35	1....	1	1	20	19	Reunified
104.	F112	1987	36	1....	1	1	21	20	Reunified
105.	F112	1988	37	1....	1	1	22	21	Reunified
106.	F112	1989	38	1....	1	1	23	22	Reunified
107.	F112	1990	39	1....	1	1	24	23	Reunified
108.	F112	1991	40	1....	1	1	25	24	Reunified
109.	F112	1992	41	1....	1	1	26	25	Reunified
110.	F112	1993	42	1....	1	1	27	26	Reunified
111.	F112	1994	43	1....	1	1	28	27	Reunified
112.	F112	1995	44	1....	1	1	29	28	Reunified
126.	F121	1986	14	Never in union
127.	F121	1987	15	Never in union
128.	F121	1988	16	Never in union
129.	F121	1989	17	Never in union
130.	F121	1990	18	Never in union
131.	F121	1991	19	Never in union
132.	F121	1992	20	Never in union
133.	F121	1993	21	Never in union
134.	F121	1994	22	Never in union
135.	F121	1995	23	Never in union
156.	F127	1985	21	1....	1	1	2	2	Always unified
157.	F127	1986	22	1....	1	1	3	3	Always unified
158.	F127	1987	23	1....	1	1	4	4	Always unified
159.	F127	1988	24	1....	1	1	5	5	Always unified
160.	F127	1989	25	.2...	1	0	1	0	Transnational
161.	F127	1990	26	.2...	1	0	2	0	Transnational
162.	F127	1991	27	.2...	1	0	3	0	Transnational
163.	F127	1992	28	..34.	1	1	1	1	Always unified
164.	F127	1993	29	..34.	1	1	2	2	Always unified
165.	F127	1994	30	..3..	1	0	1	0	Transnational
166.	F127	1995	31	..3..	1	0	2	0	Transnational
180.	I172	1985	14	1....	1	1	1	1	Always unified
181.	I172	1986	15	1....	1	1	2	2	Always unified
182.	I172	1987	16	1....	1	1	3	3	Always unified
183.	I172	1988	17	1....	1	1	4	4	Always unified
184.	I172	1989	18	1....	1	1	5	5	Always unified
185.	I172	1990	19	1....	1	1	6	6	Always unified
186.	I172	1991	20	1....	1	1	7	7	Always unified
187.	I172	1992	21	1....	1	1	8	8	Always unified
188.	I172	1993	22	Separated or widowed
189.	I172	1994	23	Separated or widowed
190.	I172	1995	24	Separated or widowed
204.	S011202	1989	14	Never in union

205.	S011202	1990	15	Never in union	
206.	S011202	1991	16	Never in union	
207.	S011202	1992	17	Never in union	
208.	S011202	1993	18	Never in union	
209.	S011202	1994	19	Never in union	
210.	S011202	1995	20	Never in union	
259.	S079601	1985	49	123..	1	1	2	2	Always unified
260.	S079601	1986	50	123..	1	1	3	3	Always unified
261.	S079601	1987	51	123..	1	1	4	4	Always unified
262.	S079601	1988	52	12...	1	1	8	8	Always unified
263.	S079601	1989	53	12...	1	1	9	9	Always unified
264.	S079601	1990	54	12...	1	1	10	10	Always unified
265.	S079601	1991	55	12...	1	1	11	11	Always unified
266.	S079601	1992	56	12...	1	1	12	12	Always unified
267.	S079601	1993	57	12...	1	1	13	13	Always unified
268.	S079601	1994	58	12...	1	1	14	14	Always unified
269.	S079601	1995	59	12...	1	1	15	15	Always unified
322.	S093701	1985	53	.2345	1	1	8	8	Always unified
323.	S093701	1986	54	.2345	1	1	9	9	Always unified
324.	S093701	1987	55	.2345	1	1	10	10	Always unified
325.	S093701	1988	56	.2345	1	1	11	11	Always unified

326.	S093701	1989	57	.2345	1	1	12	12	Always unified
327.	S093701	1990	58	.2345	1	1	13	13	Always unified
328.	S093701	1991	59	.2345	1	1	14	14	Always unified
329.	S093701	1992	60	.2345	1	1	15	15	Always unified
330.	S093701	1993	61	.2345	1	1	16	16	Always unified

331.	S093701	1994	62	.2345	1	1	17	17	Always unified
332.	S093701	1995	63	.2345	1	1	18	18	Always unified

Extrait de la base trans entre 2000 et 2008

	ident	year	id_hh	uni	uni_loc	cuni	cuni_loc	trans2
15.	E1	2000	1....	1	1	11	10	Reunified
16.	E1	2001	1....	1	0	12	10	Transnational
17.	E1	2002	1....	1	0	13	10	Transnational
18.	E1	2003	1....	1	0	14	10	Transnational
19.	E1	2004	1....	1	0	15	10	Transnational
20.	E1	2005	1....	1	0	16	10	Transnational
21.	E1	2006	1....	1	0	17	10	Transnational
22.	E1	2007	1....	1	0	18	10	Transnational
23.	E1	2008	1....	1	0	19	10	Transnational
32.	E103	2000	Separated or widowed
33.	E103	2001	.2...	1	0	1	0	Transnational
34.	E103	2002	.2...	1	0	2	0	Transnational
35.	E103	2003	.2...	1	0	3	0	Transnational
36.	E103	2004	.2...	1	0	4	0	Transnational
37.	E103	2005	.2...	1	0	5	0	Transnational
38.	E103	2006	.2...	1	0	6	0	Transnational
39.	E103	2007	Separated or widowed
40.	E103	2008	Separated or widowed
73.	E147	2000	12...	1	0	11	4	Transnational
74.	E147	2001	12...	1	0	12	4	Transnational
75.	E147	2002	12...	1	0	13	4	Transnational
76.	E147	2003	12...	1	0	14	4	Transnational
77.	E147	2004	123..	1	0	1	0	Transnational
78.	E147	2005	123..	1	0	2	0	Transnational
79.	E147	2006	123..	1	0	3	0	Transnational
80.	E147	2007	123..	1	0	4	0	Transnational
81.	E147	2008	123..	1	0	5	0	Transnational
117.	F112	2000	1....	1	1	34	33	Reunified
118.	F112	2001	1....	1	1	35	34	Reunified
119.	F112	2002	1....	1	1	36	35	Reunified
120.	F112	2003	1....	1	1	37	36	Reunified
121.	F112	2004	1....	1	1	38	37	Reunified
122.	F112	2005	1....	1	1	39	38	Reunified
123.	F112	2006	1....	1	1	40	39	Reunified
124.	F112	2007	1....	1	1	41	40	Reunified
125.	F112	2008	1....	1	1	42	41	Reunified
140.	F121	2000	1....	1	0	3	0	Transnational
141.	F121	2001	1....	1	0	4	0	Transnational
142.	F121	2002	1....	1	0	5	0	Transnational
143.	F121	2003	1....	1	0	6	0	Transnational
144.	F121	2004	1....	1	0	7	0	Transnational
145.	F121	2005	1....	1	0	8	0	Transnational
146.	F121	2006	1....	1	0	9	0	Transnational
147.	F121	2007	1....	1	1	10	1	Reunified
148.	F121	2008	1....	1	0	11	1	Transnational
171.	F127	2000	..3..	1	0	7	0	Transnational
172.	F127	2001	..3..	1	0	8	0	Transnational
173.	F127	2002	..3.5	1	1	1	1	Always unified
174.	F127	2003	..3.5	1	1	2	2	Always unified
175.	F127	2004	..3.5	1	1	3	3	Always unified
176.	F127	2005	..3.5	1	1	4	4	Always unified
177.	F127	2006	..3.5	1	1	5	5	Always unified
178.	F127	2007	..3.5	1	1	6	6	Always unified
179.	F127	2008	..3.5	1	1	7	7	Always unified
195.	I172	2000	Separated or widowed
196.	I172	2001	Separated or widowed
197.	I172	2002	.2...	1	0	1	0	Transnational
198.	I172	2003	.2...	1	0	2	0	Transnational
199.	I172	2004	.2...	1	0	3	0	Transnational
201.	I172	2006	.2...	1	0	5	0	Transnational
202.	I172	2007	.2...	1	0	6	0	Transnational
203.	I172	2008	.2...	1	0	7	0	Transnational
215.	S011202	2000	1....	1	0	2	1	Transnational
216.	S011202	2001	1....	1	0	3	1	Transnational
217.	S011202	2002	1....	1	0	4	1	Transnational
218.	S011202	2003	1....	1	0	5	1	Transnational
219.	S011202	2004	1....	1	0	6	1	Transnational
220.	S011202	2005	1....	1	1	7	2	Reunified
221.	S011202	2006	1....	1	1	8	3	Reunified
222.	S011202	2007	1....	1	0	9	3	Transnational

223.	S011202	2008	1....	1	0	10	3	Transnational
274.	S079601	2000	12...	1	1	20	20	Always unified
275.	S079601	2001	12...	1	1	21	21	Always unified
276.	S079601	2002	12...	1	1	22	22	Always unified
277.	S079601	2003	12...	1	1	23	23	Always unified
278.	S079601	2004	12...	1	1	24	24	Always unified
279.	S079601	2005	12...	1	1	25	25	Always unified
280.	S079601	2006	12...	1	1	26	26	Always unified
281.	S079601	2007	12...	1	1	27	27	Always unified
282.	S079601	2008	12...	1	1	28	28	Always unified
337.	S093701	2000	.2345	1	1	23	23	Always unified
338.	S093701	2001	.2345	1	1	24	24	Always unified
339.	S093701	2002	.2345	1	1	25	25	Always unified
340.	S093701	2003	.2345	1	1	26	26	Always unified
341.	S093701	2004	.2345	1	1	27	27	Always unified
342.	S093701	2005	.2345	1	0	28	27	Transnational
343.	S093701	2006	.2345	1	0	29	27	Transnational
344.	S093701	2007	.2345	1	0	30	27	Transnational
345.	S093701	2008	.2345	1	0	31	27	Transnational

ANNEXES

Création de la base d'analyse

1) Création de la base individus années *year.dta* à partir d'un module biographique

a) Modules dont les trajectoires ne sont pas strictement continues

Chaque module biographique contient des variables communes (le sexe, l'année de naissance, niveau d'éducation....) issues de la base *general.dta*. On va utiliser de nouveau ici la variable **q1a**.

Par exemple dans la base « union ».

```
describe ident num_union q1a
      storage  display  value
variable name  type    format  label    variable label
-----
ident          str7    %9s          Biographic individual identifier
num_union      byte    %9.0g       Number of the union
q1a            int     %17.0g      Q1A       Year of birth
```

La seule différence avec la première solution est qu'on doit avoir une seule observation (ligne) par individu. Pour les personnes qui ont déclaré plus d'une union, on ne retient que la première (`num_union=1`), en n'oubliant pas de conserver également ceux qui n'en n'ont déclaré aucune (`num_union=.`).

```
use "union.dta", clear
keep if num_union==1 | num_union==.
```

Les instructions sont ensuite strictement identiques à celles du début du tutoriel.

```
generate dur=2008-q1a+1
expand (dur)
bysort ident: generate year=q1a+(_n-1)
bysort ident: generate age= _n-1
label variable year "Year"
label variable age "Age"
drop num_union q1a a2008
save "year.dta"
```

b) A partir d'un module strictement continu, *house.dta* ou *activity.dta*

On va prendre la base *house.dta*, sans utiliser la variable `q1a` (année de naissance) présente dans chaque module de l'enquête.

Toutes les années de vie du répondant jusqu'à l'année de l'enquête sont couvertes, et la valeur de `q301d` pour `num_log=1` correspond à l'année de naissance de l'individu. On procède comme dans les autres modules (*migration.dta*, *union.dta* et *network.dta*) en n'oubliant pas de supprimer les

doublons générés par la procédure `expand`, les trajectoires étant strictement continues ($q301d(t+1)=q301f(t)$).

```
use "house.dta", clear
generate end_08=1 if q301f==2008
replace q301f=2007 if q301f==2008
replace q301f=2008 if q301f==.
expand (q301f-q301d+1)
bysort ident num_log: generate year=q301d+_n-1
drop if (q301f==year) & year != 2008 & end_08 != 1
bysort ident: generate age=_n-1
keep ident year age
save "year.dta"
```

2) Pays de résidence du répondant à partir de la base « house »

La démarche est strictement identique à la précédente, sachant qu'on dispose du pays de résidence avec la variable `q302`. C'est en fait la solution la plus rapide car on génère directement la base `loc_resp.dta`.

```
use "house.dta", clear
generate end_08=1 if q301f==2008
replace q301f=2007 if q301f==2008
replace q301f=2008 if q301f==.
expand (q301f-q301d+1)
bysort ident num_log: generate year=q301d+_n-1
drop if (q301f==year) & year != 2008 & end_08!=1
drop end_08
save "loc_resp.dta"
```

3) Correction de l'overlapping : réduction de la largeur du reshape

Si l'on souhaite restreindre l'élargissement de la base aux seuls individus qui ont vécu au moins une année dans une union polygame, on doit réaliser le `reshape` avec une autre variable que `num_uni` (ici la variable `overlap`). Cette variable prend toujours la valeur 1 pour les individus sans overlapping, et prend les valeurs de `num_union` dans le cas contraire. Pour repérer la présence d'overlapping (ici la polygamie), on peut utiliser l'instruction de gestion des doublons de Stata `duplicates`.

(<http://www.stata.com/manuals13/dduplicates.pdf>)

```
generate overlap=1
duplicates tag id, generate(dup)
bysort ident: egen tdup=total(dup)
replace overlap=num_union if tdup>0 & tdup<.
drop dup tdup
sort ident overlap
reshape wide q* loc_conj num_union, i(id) j(overlap)
order ident year, first
drop id
sort ident year
save "loc_conj.dta"
```

Difference entre les reshapes générés par les variables num_event et overlap lorsque les évènements se succèdent sans overlapping

On suppose ici que le premier évènement dure de (t1) à (t2), le second démarre en (t4)

ident	t	id	num_event	overlap	var	nbe
1	t1	1t1	1	1	v1	1
1	t2	1t2	1	1	v1	1
1	t4	1t4	2	1	v2	1
1	t5	1t5	2	1	v2	1
1	t6	1t6	2	1	v2	1
1	t7	1t7	2	1	v1	1

Reshape avec la variable overlap : reshape wide var, i(id) j(overlap).

Pour la deuxième union la valeur d'overlap reste toujours à 1. La base n'est donc pas modifiée, et la variable overlap disparaît.

ident	t	id	num_event	var	nbe
1	t1	1t1	1	v1	1
1	t2	1t2	1	v1	1
1	t4	1t4	2	v2	1
1	t5	1t5	2	v2	1
1	t6	1t6	2	v2	1
1	t7	1t7	2	v2	1

Reshape avec num_event : reshape wide var, i(id) j(num_event)

La variable num_event disparaît mais la base est élargie et contient une variable supplémentaire (var_2).

ident	t	id	var_1	var_2	overlap	nbe
1	t1	1t1	v1	.	1	1
1	t2	1t2	v1	.	1	1
1	t4	1t4	.	v2	1	1
1	t5	1t5	.	v2	1	1
1	t6	1t6	.	v2	1	1
1	t7	1t7	.	v2	1	1

Création des variables d'analyse

1) Report des informations sur les séparations

a) Réduire le report aux années sans union

Pour avoir l'information sur les séparations seulement lorsque le répondant n'est pas en couple, il suffit de rajouter la condition `nbu==.` à la première ligne

```
for num 1/5: replace q103X=q103X[_n-1] if num_uniX==. & nbu==. & ident[_n]==ident[_n-1]
for num 1/5: replace q103X=. if num_uniX!=.
```

Contrairement à ce qui a été fait dans le tutoriel, l'information sur la première séparation n'est reportée que pour les années 1996 à 2000, avant la seconde union qui démarre en 2001.

	ident	year	age	num_uni1	q1031	num_uni2	q1032
24.	E103	1992	14	1	.	.	.
25.	E103	1993	15	1	.	.	.
26.	E103	1994	16	1	.	.	.
27.	E103	1995	17	1	.	.	.
28.	E103	1996	18	.	Separation or divorce	.	.
29.	E103	1997	19	.	Separation or divorce	.	.
30.	E103	1998	20	.	Separation or divorce	.	.
31.	E103	1999	21	.	Separation or divorce	.	.
32.	E103	2000	22	.	Separation or divorce	.	.
33.	E103	2001	23	.	.	2	.
34.	E103	2002	24	.	.	2	.
35.	E103	2003	25	.	.	2	.
36.	E103	2004	26	.	.	2	.
37.	E103	2005	27	.	.	2	.
38.	E103	2006	28	.	.	2	.
39.	E103	2007	29	.	.	Separation or divorce	.
40.	E103	2008	30	.	.	Separation or divorce	.

b) Reporter les informations à partir de la base `union.dta`

Avec STATA cette solution est plus longue⁵. Il s'agit de créer dans la première partie du programme une base individus-événements pour les épisodes de séparation. A partir de la base `union.dta`, on sélectionne les unions qui ont été rompues avec `q103` et on calcule la durée entre 2008 et `q102f`. On utilise l'instruction `expand` et on crée la variable `year` pour cette nouvelle base (`separation.dta`). Après avoir créé la variable `year` dans la base `union.dta`, on apparie les deux bases. On peut ensuite

⁵ C'est cette approche qui est utilisée pour le tutoriel des logiciels SAS et R.

appairer la base *union.dta* avec la base *network.dta* (en format individus-années) et réaliser le reshape.

```
use "union.dta" , clear
keep if q103!=.
keep ident id_uni num_union q102f q103
gen dur=2008-q102f+1
expand dur
bysort id_uni: generate year= q102f+_n-1
keep ident num_union year q103 id_uni
sort id_uni year
saveold "separation.dta"

use "union.dta", clear
drop q103
generate end_08=1 if q102f==2008
replace q102f=2007 if q102f==2008
replace q102f=2008 if q102f==.
expand (q102f-q102d+1)
bysort ident num_uni: generate year=q102d+_n-1
drop if (q102f==year) & year!=2008 & end_08!=1
keep ident num_union id_uni q102d q102f q105 q107 year
sort id_uni year
/*merge*/
merge id_uni year using "separation.dta"
drop _merge
.
.
* => merge avec base network.dta puis reshape wide
.
.
```

Si l'on souhaite limiter le report aux années sans union, la base *separations.dta* peut être créée de la manière suivante :

```
use "union.dta"
keep if q103!=.
keep ident id_uni num_union q102f q103
bysort ident: generate no_uni=q102d[_n+1]
keep if no_uni!=.
generate dur=no_uni-q102f+1
expand dur
keep if dur>0
bysort id_uni: generate year= no_uni+_n-1
```

2) Comparaison des pays de résidence des co-épouses

Pour comparer les pays de résidence des différentes co-épouses, plusieurs façons de procéder sont possibles. On choisit ici de comparer chaque le cods pays, **loc_conjX**, avec la moyenne de ces codes par nombre d'union simultanées ($n_{bu}>1$): si, une année donnée, cette moyenne n'est pas strictement égale aux codes pays des coépouses, cela indique qu'au moins deux coépouses ne vivent pas dans le même pays.

Création du code moyen des pays :

```
egen t_code=rsum(loc_conj1 loc_conj2 loc_conj3) if nbu>1 & nbu<.
generate m_code=t_code/nbu if nbu>1 & nbu<.
drop t_code
```

Comme on ne peut pas seulement comparer ce code moyen à la première union seulement (num_uni1), car celle-ci peut être rompue, on effectue cette comparaison sur toutes les unions.

```
generate loc_poly=1 if loc_conj1==m_code
for num 2/5: replace loc_poly=1 if loc_conjX==m_code
replace loc_poly=0 if loc_poly==. & nbu>1 & nbu<.
label variable loc_poly "Partners in the same country =1"
```

	year	loc_resp	loc_co~1	loc_co~2	loc_co~3	loc_co~4	loc_co~5	nbu	m_code	loc_poly
298.	1961	99341	99341	1	.	.
299.	1962	99341	99341	1	.	.
300.	1963	99341	99341	1	.	.
301.	1964	99341	99341	1	.	.
302.	1965	99341	99341	1	.	.
303.	1966	99341	99341	1	.	.
304.	1967	99341	99341	1	.	.
305.	1968	99341	99341	99341	.	.	.	2	99341	1
306.	1969	99341	99341	99341	.	.	.	2	99341	1
307.	1970	99341	.	99341	.	.	.	1	.	.
308.	1971	99341	.	99341	.	.	.	1	.	.
309.	1972	99341	.	99341	99341	.	.	2	99341	1
310.	1973	99341	.	99341	99341	.	.	2	99341	1
311.	1974	99341	.	99341	99341	.	.	2	99341	1
312.	1975	99341	.	99341	99341	99341	.	3	99341	1
313.	1976	99341	.	99341	99341	99341	.	3	99341	1
314.	1977	99341	.	99341	99341	99341	.	3	99341	1
315.	1978	99341	.	99341	99341	99341	99341	4	99341	1
316.	1979	99341	.	99341	99341	99341	99341	4	99341	1
317.	1980	99341	.	99341	99341	99341	99341	4	99341	1
.
.	Idem
.
.
339.	2002	99341	.	99341	99341	99341	99341	4	99341	1
340.	2003	99341	.	99341	99341	99341	99341	4	99341	1
341.	2004	99341	.	99341	99341	99341	99341	4	99341	1
342.	2005	99341	.	99341	99341	96000	99341	4	98505.75	0
343.	2006	99341	.	99341	99341	96000	99341	4	98505.75	0
344.	2007	99341	.	99341	99341	96000	99134	4	98454	0
345.	2008	99341	.	99341	99341	96000	99134	4	98454	0

Programme alternatif

Le tutoriel n'a proposé qu'une manière de faire parmi d'autres. Dans la logique de celui réalisé avec R (diffusion prévue en 2015) on peut procéder autrement, en générant la grande majorité des variables avant le reshape:

- Modifier les informations de la variable **q103** en créant la base séparation (voir ci-dessus dans l'annexe).
- Modifier les informations de la variable **q105** après l'expand de la base *union.dta*.
- Création des variables dichotomique **V, D, U, M** après l'appariement de la base *union.dta* et *separation.dta*.
- Création de la variable **uni_loc** après avoir apparié la base *loc_conj.dta* et *loc_resp.dta*.

Pour ces deux derniers types de variables, on procède par comptage avec la fonction `total` associé à l'instruction `egen`.

```
*****
* ANALYSIS DATASET *
*****

/* person year database for each year of life with general.dta*/

use "general.dta", clear
generate a2008=2008
generate dur=(a2008-qla)+1
expand dur
bysort ident: generate year=qla+_n-1
bysort ident: generate age=_n-1
label variable year "Year"
label variable age "Age"
save "year.dta"

/* person year database for each year of migration */
use "migration.dta", clear
drop if num_mig==.
generate end_08=1 if q601f==2008
replace q601f=2007 if q601f==2008
recode q601f .=2008
expand (q601f-q601d+1)
bysort ident num_mig: generate year=q601d+_n-1
drop if (q601f==year) & year!=2008 & end_08!=1
drop end_08
sort ident year
save "loc_resp.dta", replace

/* merge loc_resp.dta & year.dta */
use "year.dta"
sort ident year
merge 1:1 ident year using "loc_resp.dta"
drop _merge
rename q602 loc_resp
recode loc_resp (.=99341) /*99341 = code for Senegal*/
keep ident year age loc_resp
save "loc_resp.dta", replace

/* person year dataset for migration (partners)*/
use "network.dta", clear
keep if q901==1
generate end_08=1 if q904f==2008
replace q904f=2007 if q904f==2008
replace q904f=2008 if q904f==.
```

```

expand (q904f-q904d+1)
bysort id_uni num_netmig: generate year=q904d+_n-1
drop if (q904f==year) & year!=2008 & end_08!=1
keep ident id_uni country q904d q904f year
sort id_uni year
save "net_conj.dta", replace

/* separation dataset => q103 */
use "union.dta" , clear
keep if q103!=.
keep ident id_uni num_union q102f q103
gen dur=2008-q102f+1
expand dur
bysort id_uni: generate year= q102f+_n-1
keep ident num_union year q103 id_uni
sort id_uni year
saveold "separation.dta"

/*union dataset*/
use "union.dta", clear
drop q103

generate uni=1

generate end_08=1 if q102f==2008
replace q102f=2007 if q102f==2008
replace q102f=2008 if q102f==.
expand (q102f-q102d+1)
bysort ident num_uni: generate year=q102d+_n-1
drop if (q102f==year) & year!=2008 & end_08!=1
keep ident num_union id_uni q102d q102f q105 q107 year uni
sort id_uni year

/*q105*/
replace q105=. if q105>year
replace q105=1 if q105!=.

drop if num_union==.

/*merge union + separation + net_conj*/
merge id_uni year using "separation" "net_conj.dta"
drop _merge
sort ident year
keep ident year id_uni num_union q103 q107 q105 id_uni country uni

replace country=q107 if country==.
drop if num_union==.

save "loc_conj.dta", replace

tostring year, generate(y)
generate id=ident+y
drop y

replace uni=0 if uni==.

/* Dummies V,D,U,M for marital status */
replace q105=0 if q105==.

/* V=widowed */
generate x=1 if q103==2
replace x=0 if x==.
bysort id: egen cumx=total(x)
generate V=1 if cumx>0
replace V=0 if V==.
drop x cumx

```

```

/* D=divorced/separated */
generate x=1 if q103==1
replace x=0 if x==.
bysort id: egen cumx=total(x)
generate D=1 if cumx>0
replace D=0 if D==.
drop x cumx

/* U=free union */
generate x=1 if q105==0 & uni==1
replace x=0 if x==.
bysort id: egen cumx=total(x)
generate U=1 if cumx>0
replace U=0 if U==.
drop x cumx

/* M=married */
generate x=1 if q105==1
replace x=0 if x==.
bysort id: egen cumx=sum(x)
generate M=1 if cumx>0
replace M=0 if M==.
drop x cumx

/* num_uni for part II */
generate num_uni=num_union if uni==1

/* variable nbu for part II */
bysort id: egen nbu=total(uni)

sort id
save "loc_conj", replace

/* merge loc_cong loc_resp */

use "loc_resp.dta", clear

tostring year, generate(y)
generate id=ident+y
drop y
sort id

merge id using "loc_conj.dta"
drop _merge

replace nbu=0 if nbu==.
drop if age<14

/* variable uni_loc */
generate loc=1 if loc_resp==country & uni==1
replace loc=0 if loc==. & uni==1
bysort id: egen tloc=total(loc)
sort id num_uni
generate uni_loc=1 if tloc==nbu
replace uni_loc=0 if tloc<nbu | nbu==0
drop loc tloc q107 uni id_uni q105 q103 country loc_resp

/* reshape wide */
sort id
drop if num_union==. /* error if num_union==. */
reshape wide num_uni, i(id) j(num_union)
sort ident age
drop id
save "analys.dta", replace

```

```

/* merge loc_resp => year before the first union*/

use "loc_resp.dta", clear
keep ident age
sort ident age
merge ident age using "analys.dta"
drop _merge
drop if age<14

replace D=0 if D==.
replace V=0 if V==.
replace U=0 if U==.
replace M=0 if M==.
replace nbu=0 if nbu==.

replace uni_loc=. if nbu==0

*****
* B.ANALYSIS VARIABLES *
*****

/** TVC MARITAL STATUS **/

/* variable C=never in union */

generate C=1 if nbu==0 & D==0 & V==0
order C, before(V)

/* tvc marital */
order marital, after(M)
replace marital=1 if C==1
replace marital=2 if D==1 & nbu==0
replace marital=3 if V==1 & nbu==0
replace marital=4 if U==1 & nbu==1
replace marital=5 if M==1 & nbu==1
replace marital=6 if U==1 & nbu>1 & M==0
replace marital=7 if M==1 & nbu>1

label define mar_stat 1 "Alone, never in union", replace
label define mar_stat 2 "Alone, separated", modify
label define mar_stat 3 "Alone, widowed", modify
label define mar_stat 4 "Monogamy, informal union", modify
label define mar_stat 5 "Monogamy, married", modify
label define mar_stat 6 "Polygamy, only informal unions", modify
label define mar_stat 7 "Polygamy, at least one marriage", modify
label value marital mar_stat

tabulate marital

/** TVC MARITAL STATUS & LOCALISATION*/

/* suni_id: id for each union's structure*/
for num 1/5: tostring num_uniX, generate (x_X)
egen suni_id=concat(x_*)
replace suni_id="" if suni_id=="....."
drop x_*
order suni_id, after(marital)

/* tvc trans1 trans2*/
sort ident suni_id year
generate uni=1 if nbu>0
by ident suni_id: generate cuni=sum(uni) if uni==1
by ident suni_id: generate cuni_loc=sum(uni_loc) if uni==1
sort ident year

generate trans=0 if nbu==0
replace trans=1 if uni_loc==0

```

```

replace trans=2 if uni_loc==1 & cuni>cuni_loc
replace trans=3 if uni_loc==1 & cuni==cuni_loc
label variable trans "Marital/Geographical situation each year"

label define TRANS 0 "No partner/spouse", modify
label define TRANS 1 "Transnational", modify
label define TRANS 2 "Reunified", modify
label define TRANS 3 "Always unified", modify
label value trans TRANS

tabulate trans

/* Same variable with some details for marital status */
generate trans2=0 if trans==0 & C==1
  replace trans2=1 if trans==0 & (D==1 | V==1) & nbu==0
  replace trans2=2 if trans==1
  replace trans2=3 if trans==2
  replace trans2=4 if trans==3

label define TRANS2 0 "Never in union", modify
label define TRANS2 1 "Separated or widowed", modify
label define TRANS2 2 "Transnational", modify
label define TRANS2 3 "Reunified", modify
label define TRANS2 4 "Always unified", modify
label value trans2 TRANS2

tabulate trans2

```