



The World's Largest Open Access Agricultural & Applied Economics Digital Library

This document is discoverable and free to researchers across the globe due to the work of AgEcon Search.

Help ensure our sustainability.

Give to AgEcon Search

AgEcon Search

<http://ageconsearch.umn.edu>

aesearch@umn.edu

*Papers downloaded from **AgEcon Search** may be used for non-commercial purposes and personal study only. No other use, including posting to another Internet site, is permitted without permission from the copyright owner (not AgEcon Search), or as allowed under the provisions of Fair Use, U.S. Copyright Act, Title 17 U.S.C.*

No endorsement of AgEcon Search or its fundraising activities by the author(s) of the following work or their employer(s) is intended or implied.

Stata tip 83: Merging multilingual datasets

Devra L. Golbe
 Hunter College, CUNY
 New York
 dgolbe@hunter.cuny.edu

`merge` is one of Stata’s most important commands. Without specific instructions to the contrary, `merge` holds the master data file inviolate. Its variables are neither replaced nor updated. Variable and value labels are retained. All these properties and more are well documented. What is not so well documented is how `merge` interacts with Stata’s multiple language support (`label language`), added in Stata 8.1 and described in Weesie (2005). In essence, Stata users must pay careful attention to which languages are defined and current when merging files.

The language feature is useful not only for multiple “real” languages (e.g., English and French) but also for using different sets of labels for different purposes, such as short labels (“Mines”) and long (“Non-Metallic and Industrial Metal Mining”) in one data file. `merge` may generate unexpected results if attention is not paid to the language definitions and, in particular, the current language in each file. Multilingual datasets to be merged should be defined with common languages and each should have the same language set as the current language.

I illustrate using `auto.dta`. Starting with `autotech.dta`, create a new dichotomous variable, `guzzler`, defined as `mpg < 25`, label the variable and its values in English (en) and French (fr), and save to a file called `tech.dta`. The tabulations below display variable and value labels:

```
. label language en
. tabulate guzzler
```

Gas Guzzler	Freq.	Percent	Cum.
No	19	25.68	25.68
Yes	55	74.32	100.00
Total	74	100.00	

```
. label language fr
. tabulate guzzler
```

Verte	Freq.	Percent	Cum.
Oui	19	25.68	25.68
Non	55	74.32	100.00
Total	74	100.00	

From `auto.dta`, create an English-labeled file, `origin.dta`. Now merge in `tech.dta` and tabulate `guzzler` and `foreign`:

```
. tabulate guzzler foreign
```

Verte	Car type		Total
	Domestic	Foreign	
Oui	8	11	19
Non	44	11	55
Total	52	22	74

`foreign` is labeled in English, but `guzzler` is labeled in French. How did that happen? The label `language` and `labelbook` commands can clarify:

```
. label language
```

Language for variable and value labels

In this dataset, value and variable labels have been defined in only one language: en

(output omitted)

```
. labelbook
```

value label origin

(output omitted)

definition

0 Domestic

1 Foreign

variables: foreign

value label yesno_en

(output omitted)

definition

0 No

1 Yes

variables:

value label yesno_fr

(output omitted)

definition

0 Oui

1 Non

variables: guzzler

(output omitted)

On reflection, this is what we might have expected. The master dataset is inviolate in the sense that its language—English (and only English)—is preserved. The using dataset has two languages, but only the labels from the current language (French) are attached to the single language (English) in the master dataset.

We could, of course, redefine our languages and reattach the appropriate labels. However, if we plan to merge our master file with a multilingual dataset, a better strategy is to prepare the master file by defining the same two languages and then merge. It is crucial that the current languages are the same in both files. To illustrate, add French labels to `origin.dta`. Then consider what happens if the current languages differ. Suppose that the current language in the master file (`origin.dta`) is French, but the current language in the using file (`tech.dta`) is English. Then the labels from the current language in the using file (English) are attached in the current language of the master file (French), and the French labels (noncurrent) from the using file are not attached at all:

```
. label language fr
(fr already current language)
. tabulate guzzler foreign
```

Gas Guzzler	Origine USA Autre		Total
No	8	11	19
Yes	44	11	55
Total	52	22	74

```
. label language en
. tabulate guzzler foreign
```

guzzler	Car type Domestic Foreign		Total
0	8	11	19
1	44	11	55
Total	52	22	74

Although we could of course correct the labels afterward, it is easier to make sure that the files are consistent before the merge. If we set the current language to English (or French) in both files before merging, we see that the labels are properly attached:

```
. label language en
(en already current language)
. tabulate guzzler foreign
```

Gas Guzzler	Car type Domestic Foreign		Total
No	8	11	19
Yes	44	11	55
Total	52	22	74

```
. label language fr
```

```
. tabulate guzzler foreign
```

Verte	Origine		Total
	USA	Autre	
Oui	8	11	19
Non	44	11	55
Total	52	22	74

Whether or not the labels are properly attached, `merge` does preserve them. But a little planning in advance will ensure that they are attached in the way you expect.

In passing, it is worth noting that the situation is more complex if there are no languages defined in the master file. In that case, issuing a `label language` statement changes the behavior of `merge`. First, let's see what happens if we ignore the language of the master file. Thus we start with the original `autotech.dta` and merge in the multilingual `origin.dta`. We see that two languages have been defined, and the labels are properly attached:

```
. webuse autotech, clear
(1978 Automobile Data)
. merge 1:1 make using origin, nogenerate
(output omitted)
```

```
. tabulate foreign
```

Car type	Freq.	Percent	Cum.
Domestic	52	70.27	70.27
Foreign	22	29.73	100.00
Total	74	100.00	

```
. label language fr
```

```
. tabulate foreign
```

Origine	Freq.	Percent	Cum.
USA	52	70.27	70.27
Autre	22	29.73	100.00
Total	74	100.00	

If before merging, however, we check that the master file's only language is the default, we get different results:

```
. webuse autotech, clear
(1978 Automobile Data)
```

```
. label language
```

Language for variable and value labels

```
    In this dataset, value and variable labels have been defined in only one
    language:  default
```

```
(output omitted)
```

```
. merge 1:1 make using origin, nogenerate
(output omitted)
. label language
Language for variable and value labels
    In this dataset, value and variable labels have been defined in only one
    language: default
(output omitted)
```

In this case, only the default language is defined. Tabulation of **foreign** indicates that labels are attached from only the current language in the using file. The reason is that issuing the **label language** command sets the characteristics that define the current language and all available languages. **merge** then declines to overwrite those characteristics with characteristics from the using dataset. If they are as yet undefined, however, those characteristics are taken from the using language.

Reference

Weesie, J. 2005. Multilingual datasets. *Stata Journal* 5: 162–187.