

Description

`st_tsrevar(s)` is the equivalent of Stata's [\[TS\] **tsrevar**](#) programming command: it generates temporary variables containing the evaluation of any *op.varname* combinations appearing in *s*.

`_st_tsrevar(s)` does the same thing as `st_tsrevar()`. The two functions differ in how they respond to invalid elements of *s*. `st_tsrevar()` aborts with error, and `_st_tsrevar()` places missing in the appropriate element of the returned result.

Syntax

real rowvector `st_tsrevar(string rowvector s)`

real rowvector `_st_tsrevar(string rowvector s)`

Remarks and examples

Both of these functions help achieve efficiency when using views and time-series variables. Assume that in *vars*, you have a list of Stata variable names, some of which might contain time-series *op.varname* combinations such as `l.gnp`. For example, *vars* might contain

```
vars = "gnp r l.gnp"
```

If you wanted to create in *V* a view on the data, you would usually code

```
st_view(V, ., vars)
```

We are not going to do that, however, because we plan to do many calculations with *V* and, to speed execution, we want any *op.varname* combinations evaluated just once, as *V* is created. Of course, if efficiency were our only concern, we would code

```
V = st_data(., vars)
```

Assume, however, that we have lots of data, so memory is an issue, and yet we still want as much efficiency as possible given the constraint of not copying the data. The solution is to code

```
st_view(V, ., st_tsrevar(tokens(vars)))
```

`st_tsrevar()` will create temporary variables for each *op.varname* combination (`l.gnp` in our example), and then return the Stata variable indices of each of the variables, whether newly created or already existing. If `gnp` was the second variable in the dataset, `r` was the 23rd, and in total there were 54 variables, then returned by `st_tsrevar()` would be `(2, 23, 55)`. Variable 55 is new, created by `st_tsrevar()`, and it contains the values of `l.gnp`. The new variable is temporary and will be dropped automatically at the appropriate time.

Conformability

```
st_tsrevar(s), _st_tsrevar(s):
    s:      1 × c
    result: 1 × c
```

Diagnostics

`st_tsrevar()` aborts with error if any variable name is not found or any *op.varname* combination is invalid.

`_st_tsrevar()` puts missing in the appropriate element of the returned result for any variable name that is not found or any *op.varname* combination that is invalid.

Also see

[M-5] [st_varindex\(\)](#) — Obtain variable indices from variable names

[M-5] [st_varname\(\)](#) — Obtain variable names from variable indices

[M-4] [Stata](#) — Stata interface functions

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on [citing Stata documentation](#).

