

sysdir — Query and set system directories[Description](#)[Syntax](#)[Option](#)[Remarks and examples](#)[Also see](#)

Description

`sysdir` lists Stata's system directories.

`sysdir set` changes the path to Stata's system directories.

`personal` displays the path of the `PERSONAL` directory. `personal dir` gives a directory listing of the files contained in the `PERSONAL` directory.

`adopath` displays the ado-file path stored in the global macro `S_ADO`.

`adopath +` adds a new directory or moves an existing directory to the end of the search path stored in the global macro `S_ADO`.

`adopath ++` adds a new directory or moves an existing directory to the beginning of the search path stored in the global macro `S_ADO`.

`adopath -` removes a directory from the search path stored in the global macro `S_ADO`.

`set adosize` sets the maximum amount of memory in kilobytes that automatically loaded do-files may consume. The default is `set adosize 1000`. To view the current setting, type `display c(adosize)`.

These commands have to do with technical aspects of Stata's implementation. Except for `sysdir list`, you should never have to use them.

Syntax

List Stata's system directories

```
sysdir [list]
```

Reset Stata's system directories

```
sysdir set codeword [" ]path[" ]
```

Display path of PERSONAL directory and list files in it

```
personal [dir]
```

Display ado-file path

```
adopath
```

Add directory to end of ado-path

```
adopath + path_or_codeword
```

Add directory to beginning of `ado-path`

```
adopath ++ path_or_codeword
```

Remove directory from `ado-path`

```
adopath - {path_or_codeword | #}
```

Set maximum memory `ado`-files may consume

```
set adosize # [ , permanently ]           10 ≤ # ≤ 10000
```

where *path* must be enclosed in double quotes if it contains blanks or other special characters and *codeword* is { STATA | BASE | SITE | PLUS | PERSONAL | OLDPLACE }.

Option

`permanently` specifies that, in addition to making the change right now, the `adosize` setting be remembered and become the default setting when you invoke Stata.

Remarks and examples

[stata.com](https://www.stata.com)

Remarks are presented under the following headings:

- [Introduction](#)
- [sysdir](#)
- [adopath](#)
- [set adosize](#)

Introduction

In various parts of the Stata documentation, you will read that “Stata searches along the `ado-path`” for such-and-such. When we say that, what we really mean is “Stata searches along the path stored in the global macro `$$_ADO`”. Equivalently, we could say “searches along the path stored in `c(adopath)`” because `c(adopath) = $$_ADO`. These are just two different ways of saying the same thing. If you wanted to change the path, however, you would change the `$$_ADO` because there is no way to change `c(adopath)`.

Do not, however, directly change `$$_ADO`. Even if you have good reason to change it, you will find it easier to change it via the `adopath` command.

If you were to look inside `$$_ADO` (and we will), you would discover that it does not actually contain directory names—although it could—but contains codewords that stand for directory names. The `sysdir` command will show you the meaning of the codewords and allow you to change them.

sysdir

Stata expects to find various parts of itself in various directories (folders). Rather than describing these directories as `C:\Program Files\Stata14\ado\base` or `/usr/local/stata/ado`, these places are referred to by codewords. Here are the definitions of the codewords on a particular Windows computer:

```
. sysdir
  STATA: C:\Program Files\Stata14\
  BASE: C:\Program Files\Stata14\ado\base\
  SITE: C:\Program Files\Stata14\ado\site\
  PLUS: C:\ado\plus\
  PERSONAL: C:\ado\personal\
  OLDPLACE: C:\ado\
```

Even if you use Stata for Windows, when you type `sysdir`, you might see different directories listed.

The `sysdir` command allows you to obtain the correspondence between codeword and actual directory, and it allows you to change the mapping. Each directory serves a particular purpose:

STATA refers to the directory where the Stata executable is to be found.

BASE is where the original official ado-files that were shipped with Stata and any updated official ado-files that have been made available since then are installed.

SITE is relevant only on networked computers. It is where administrators may place ado-files for sitewide use on networked computers. No Stata command writes to this directory, but administrators may move files into the directory or obtain ado-files by using `net` and choose to install them into this directory; see [R] [net](#).

PLUS is relevant on all systems. It is where ado-files written by other people that you obtain using the `net` command are installed; by default, `net` installs files to this directory; see [R] [net](#).

PERSONAL is where you are to copy ado-files that you write and that you wish to use regardless of your current directory when you use Stata. (The alternative is to put ado-files in your current directory, and then they will be available only when you are in that directory.)

OLDPLACE is included for backward compatibility. Stata 5 users used to put ado-files here, both the personal ones and the ones written by others. Nowadays, they are supposed to put their personal files in **PERSONAL** and the ones written by others in **PLUS**.

Do not change the definitions of **BASE**. You may want to change the definitions of **SITE**, **PERSONAL**, **PLUS**, or especially **OLDPLACE**. For instance, if you want to change the definition of **OLDPLACE** to `d:\ado`, type

```
. sysdir set OLDPLACE "d:\ado"
```

Resetting a system directory affects only the current session; the next time you enter Stata, the system directories will be set back to being as they originally were. If you want to reset a system directory permanently, place the `sysdir set` command in your `profile.do`; see [GSW] [B.3 Executing commands every time Stata is started](#), [GSM] [B.1 Executing commands every time Stata is started](#), or [GSU] [B.1 Executing commands every time Stata is started](#).

adopath

`adopath` displays and resets the contents of the global macro `$$_ADO`, the path over which Stata searches for ado-files. The default search path is

```
. adopath
[1] (BASE)      "C:\Program Files\Stata14\ado\base"
[2] (SITE)     "C:\Program Files\Stata14\ado\site"
[3]           "."
[4] (PERSONAL) "C:\ado\personal"
[5] (PLUS)     "C:\ado\plus"
[6] (OLDPLACE) "C:\ado"
```

Focus on the codewords on the left. `adopath` mentions the actual directories, but if you changed the meaning of a codeword by using `sysdir`, that change would affect `adopath`.

The above states that, when Stata looks for an ado-file, first it looks in `BASE`. If the ado-file is found, then that copy is used. If it is not found, then Stata next looks in `SITE`, and if it is found there, then that copy is used. And so the process continues. At the fourth step, Stata looks in the current directory (for which there is no codeword).

`adopath` merely presents the information in `$$_ADO` in a more readable form:

```
. display "$$_ADO"
BASE;SITE;. ;PERSONAL;PLUS;OLDPLACE
```

`adopath` can also change the contents of the path. In general, you should not do this unless you are sure of what you are doing because many features of Stata will stop working if you change the path incorrectly. At worst, however, you might have to exit and reenter Stata, so you cannot do any permanent damage. Moreover, it is safe to add to the end of the path.

The path may include actual directory names, such as `C:\myprogs`, or codewords, such as `PERSONAL`, `PLUS`, and `OLDPLACE`. To add `C:\myprogs` to the end of the path, type

```
. adopath + C:\myprogs
[1] (BASE)      "C:\Program Files\Stata14\ado\base"
[2] (SITE)     "C:\Program Files\Stata14\ado\site"
[3]           "."
[4] (PERSONAL) "C:\ado\personal"
[5] (PLUS)     "C:\ado\plus"
[6] (OLDPLACE) "C:\ado"
[7]           "C:\myprogs"
```

If later you want to remove `C:\myprogs` from the ado-path, you could type `adopath - C:\myprogs`, but easier is

```
. adopath - 8
[1] (BASE)      "C:\Program Files\Stata14\ado\base"
[2] (SITE)     "C:\Program Files\Stata14\ado\site"
[3]           "."
[4] (PERSONAL) "C:\ado\personal"
[5] (PLUS)     "C:\ado\plus"
[6] (OLDPLACE) "C:\ado"
```

When followed by a number, `'adopath -'` removes that element from the path. If you cannot remember what the numbers are, you can first type `adopath` without arguments.

□ Technical note

`adopath ++ path` works like `adopath + path`, except that it adds to the beginning rather than to the end of the path. Our recommendation is that you not do this. When looking for `name.ado`, Stata

loads the first file it encounters as it searches along the path. If you did not like our implementation of the command `ci`, for instance, even if you wrote your own and stored it in `ci.ado`, Stata would continue to use the one in the Stata directory because that is the directory listed earlier in the path. To force Stata to use yours rather than ours, you would have to put at the front of the path the name of the directory where your ado-file resides.

You should not, however, name any of your ado-files the same as we have named ours. If you add to the front of the path, you assume exclusive responsibility for the Stata commands working as documented in this manual.



set adosize

Stata keeps track of the ado-commands you use and discards from memory commands that have not been used recently. Stata discards old commands to keep the amount of memory consumed by such commands less than `adosize`. The default value of 1,000 means the total amount of memory consumed by ado-commands is not to exceed 1,000 KB. When an ado-command has been discarded, Stata will have to reload the command the next time you use it.

You can increase `adosize`. Typing `set adosize 1550` would allow up to 1,550 KB to be allocated to ado-commands. This would improve performance slightly if you happened to use one of the not-recently-used commands, but at the cost of some memory no longer being available for your dataset. In practice, there is little reason to increase `adosize`.

`adosize` must be between 10 and 10,000.

Also see

[R] [net](#) — Install and manage user-written additions from the Internet

[R] [query](#) — Display system parameters

[R] [update](#) — Check for official updates

[U] [17.5 Where does Stata look for ado-files?](#)