Title

markdoc —— a general-purpose literate programming package for Stata that produces dynamic analysis documents in various formats, such as **pdf, docx, odt, html, epub, markdown,** presentation slides in **pdf** or **html,** as well as dynamic Stata package help files **sthlp.**

To improve applications of the package for developing educational materials and encouraging university lecturers to ask students to practice literate programming for taking notes or doing their semester projects, MarkDoc was programmed to include unique features. For example, it includes a syntax highlighter, it recognizes markdown, html, and latex markup languages, it can render latex mathematical notations in **pdf, docx, html, odt,** and **tex** documents, automatically capture graphs from Stata and include them in the document, creates dynamic tables, and supports writing dynamic text for interpretting the analysis. Moreover, a user-friendly GUI interface was developed for the package (try **db markdoc**) to make using **MarkDoc** easier for newbies. These features make the package a complete tool for documenting data analysis, Stata packages, as well as a tool for producing educational materials within Stata Do-file editor.

The source code of the project is hosted on GitHub and also, the package documentation is hosted on GitHub wiki. all contributions to the package, including improving the documentation or providing further examples are welcome. further resources are available in the webpages below.

Homepage
Journal Article
MarkDoc Documentation Manual
Release Notes
Examples
Please ask your questions on statalist.org

Syntax

produce dynamic *documents* , *presentation slides* , or *help files* interactively

> **markdoc** filename [, **pandoc(***str***) printer(***str***) install test replace export(***name***)**
>   **markup(***name***) numbered style(***name***) template(***str***) toc title(***str***) author(***str***)**
>   **affiliation(***str***) address(***str***) summary(***str***) date master statax noisily**]

where filename can be:

---

smcl	converts *smcl* log file to any of the supported document formats. The *smcl* log is used for creating *dynamic document* as well as *dynamic slides* .
do	executes the *do-file* in a "cleared workspace" and produces a *dynamic document* or *dynamic slides* . A cleared workspace ensures the reproducibility of the analysis because it neglects the data that is already loaded in Stata and requires the user to load the data that is used for the analysis in the do-file.
ado │ **mata**	MarkDoc handles Stata programs differently. It creates **sthlp** help files or package vignettes (**pdf, html, docx,** etc) from Stata programming script files. This process merely extracts the documentation from the source. The documentation can be written with either smcl or Markdown or a combination of both.

MarkDoc package also includes a few more commands that can be used to facilitate writing the dynamic document or dynamic slides. These commands are briefly described below, while the complete documentation is available on GitHub wiki:

write dynamic text using any of the supported markup languages. You can also use this command to write text within a loop or a program.

> **txt** [**code**] [*display_directive* [*display_directive* [*...*]]]

include an image in the document. if *filename* is missing, **img** command captures, saves, and imports the current graph from Stata automatically.

> **img** [*filename*] [, **markup(***str***) title(***str***) width(***int***) height(***int***) markup(***str***) left**
>   **center**]

create a dynamic table in Markdown documents (Not supported in HTML and LaTeX). This
    command has several directives for styling the table, creating nested tables, and
    aligning the content of each column.

    tbl *(#[,#...] [\ #[,#...] [\ [...]]])*   [, **title(***str***)**]

execute **pandoc** commands directly from Stata

    pandoc *command*]

convert html files to pdf using **wkhtmltopdf** software

    wkhtmltopdf *command*]

MarkDoc options	Description
pandoc(*str***)**	specify the path to Pandoc software on the operating system
printer(*str***)**	specify the path to PDF driver on the operating system
install	Installs the required packages and software automatically on the user's machine, if they are not accessible
test	examines if MarkDoc is working properly by creating a test document
replace	replace the exported file if already exists
export(*name***)**	exports the *smclfile* to any of the supported document formats which are **pdf, slide** (i.e. pdf slides), **docx, odt, tex, html, epub**, and **md**
markup(*name***)**	specify the markup language used for writing the document and the default is Markdown
numbered	numbers Stata commands in the dynamic document.
unc	specify that markdoc is being accessed from a Windows server with UNC file paths
style(*name***)**	specify the style of the document for HTML, PDF, Docx, and LaTeX documents. The available styles are **simple**, **stata**, and **formal**. If the document is exported in LaTeX format, the **stata** option (also if used with **master** option) will produce a LaTeX article in the Stata Journal style, even if the document is written in Markdown. In other words, you can write your stata journal article using Markdown.
template(*str***)**	renders the document using an external style sheet file. When the document is written in Markdown or HTML and exported to HTML or PDF, a CSS filename can be specified to alter the appearance of the document. Similarly, when the document is written in Markdown and exported to Microsoft Word Docx or Open Office ODT, a reference document with the similar format can be used to alter the style of the exported document (i.e. create a reference document, change the styles and themes, and use it as a template file). If the document is written in LaTeX, this option can also be used to add the required packages to the dynamic document by providing a file that inludes the packages and the template set up.
toc	creates table of content in PDF, Microsoft Word Docx, and LaTeX documents
title(*str***)**	specify the title of the document
author(*str***)**	specify the author of the document
affiliation(*str***)**	specify the author affiliation in the document
address(*str***)**	specify the author's contact information in the document
summary(*str***)**	specify the summary of the document
date	specify the current date in the document

master	while creating a LaTeX or HTML document, MarkDoc only translates the content of the smcl file to tex or html respectively. Since the document does not include the required layout, it cannot be compiled (although it can be imported in a document). This option creates the layout in LaTeX and HTML to allow compiling the document. Many features of the HTML document (that are written with HTML markup) such as mathematical notations require this option. Otherwise, the user should build the layout from scratch.
build	when generating dynamic package documentation in sthlp format, the **build** option will also create the **stata.toc** and **pkgname.pkg** automatically, so that cou can host an installable version of the package on GitHub or your personal website.
statax	highlights the syntax of Stata codes in the HTML and PDF documents using Statax, which are a JavaScript syntax highlighter engine for Stata
<u>noisily</u>	enables extended log for debugging markdoc

<u>Installation</u>

The latest release as well as archived older versions of **markdoc** are hosted on GitHub website. MarkDoc receives weekly updates on GitHub so I recommend you to "watch" (subscribe) the repository's updates on GitHub to get the latest news about the package.

markdoc depends on several other Stata modules. If you have the github package} installed, you can install **markdoc** and all of its dependencies by typing:

    . github install haghish/markdoc

The **github** command is used for searching, installing, and uninstalling Stata packages with their dependencies from GitHub website. to install the **github** command, type:

    . net install github, from("https://haghish.github.io/github/")

<u>Description</u>

markdoc is a general-purpose literate programming package for Stata that produces *dynamic analysis documents* , *package vignette documentation* , *dynamic presentation slides* , as well as dynamic *Stata package help files* .

For creating a dynamic document or presentation slides, MarkDoc requires a smcl log-file as input and converts it to other formats. For generating dynamic Stata help files or package vignette documentation, MarkDoc requires a Stata script-file (do, ado, mata) as input and exports the documentations written within the source. Visit MarkDoc homepage for documentation about generating **dynamic Stata help files and documentations**.

MarkDoc supports three different markup languages which are Markdown, HTML, and LaTeX. All of these markup languages can also include an image in the document, support writing dynamic text, and creating dynamic table. MarkDoc can export documents in various file formats including **pdf** document and **slide**, Microsoft Office **docx**, Open Office and LibreOffice **odt**, LaTeX **tex**, **html**, **epub**, Markdown **md, slidy** HTML-based slide, and also **sthlp** and **smcl** for Stata documentation. If file format is not specified, MarkDoc creates a markdown **md** file. MarkDoc requires the Weaver package for making use of the txt, tbl, and img commands which are used for writing dynamic text, creating dynamic tables, and importing figures automatically in the document, respectively. MarkDoc also requires the Statax package which provides a JavaScript syntax highlighter for Stata and Mata code in HTML and PDF documents.

MarkDoc creates the dynamic documents by converting *smcl* log-file to other file formats mentioned above or parsing the documentation written in Stata script files. The documentation should be written within the source files using a special notations that are seperated from regular comments. Weaving the dynamic document or dynamic slides can take place at any point without requiring closing the log-file, which provides live-preview of the document. This is the biggest advantage of MarkDoc and Weaver packages to other classic literate programming packages which cannot provide live-preview of the document in an interactive analysis session.

MarkDoc supports both Stata and Mata languages. Therefore, advanced users who work or program in Mata, can use Markdoc - with the same syntax and markup notation - to produce a dynamic document or slides, or document their programs. The same source that is used for generating dynamic Stata **sthlp** help files, can be used to produce Microsoft Word **docx**, **pdf**, etc.

For a more detailed documentation and examples, visit **MarkDoc Homepage**.

Writing mathematical notation

markdoc can render LaTeX mathematical notations not only when the document is exported to LaTeX **tex**, but also when the document is exported to **pdf** document or **slide**, Microsoft Office **docx**, OpenOffice and LibreOffice **odt**, and **html**.

Mathematical notations can be inline a text paragraph or on a separate line. For writing inline notations, place the notation between single dollar signs (e.g. 10^2 + b^2 = c^2$). For including notation on a separate line, place the notations between double dollar signs (e.g. $10^2 + b^2 = c^2$$). The example below demonstrates how to export a PDF presentation slides with notations:

. qui log using example, replace

    /***
    Mathematical notations can be inline a text paragraph e.g. $a^2 + b^2 = c^2$
    or on a separate line such as:

    $$a^2 + b^2 = c^2$$
        ***/

. qui log c
. markdoc example, export(slide) printer("/usr/texbin/pdflatex")

Inserting an image or figure in the document

Any of the supported markup languages can be used to insert a figure in the document. In general, there are two ways for inserting an image in the document. First, you can use Markdown, HTML, or LaTeX syntax for inserting an image —— that is already saved in your hard drive —— in the document. The other solution is using img command. **img** command can take the filename of exsisting image on the hard drive and print the markup code (Markdown, HTML, or LaTeX. the default is Markdown) int he document. **img** command can also auto-export the current graph and import it in the document. For more information in this regard see the img help file and also Examples and explanations on MarkDoc homepage

Writing dynamic text

the **txt** command is borrowed from weaver package to print dynamic text in the the exported dynamic document. It can be used for interpreting the analysis results or dynamically referring to values of scalars or macros in the dynamic document. Writing dynamic text allows the content of the text to change by altering analysis codes and thus is the desirable way for explaining the analysis results. The text and macros can be styled using any of the supported markup languages in MarkDoc which are *markdown* , *LaTeX*, and *HTML*. This command is fully documented in the txt help file.

Creating dynamic tables with tbl command

the **tbl** command also belongs to weaver package. The syntax of this command is similar to matrix input, however, it can include *String*, digits, scalars, and macros to create a dynamic table. This command is fully documented in the tbl help file.

Markers

In addition to 3 markup languages, MarkDoc also introduces a few handy markers for annotating the smcl log-file. These markers can be used to specify what parts of the log-file should not appear in the dynamic document. The table below provides a brief summary of these annotating markers. In general, comments - unless they appear after a command - will be ignored in the dynamic document. However, the markers mentioned below are "special comments" that will influence the MarkDoc process.

Marker	Description
Creating text block	
/* ... *****/**	creates a block of comments in the smcl file that will be interpreted in the dynamic document
Hiding command or output	
/**/ *command* **/***/** *command*	only include the **output** in the dynamic document only include the **command** in the dynamic document
Hiding a section	

//OFF
...   Anything placed after **"//OFF"** until **"//ON** markers will be
                                      ignored in the dynamic document
//ON

Appending external files

//IMPORT <u>filename</u>   Include an external text file (Markdown, HTML, LaTeX) to
                                      the dynamic document

_____

Apart from the writing markers " **/***** " and " *****/** ", which are used for writing text, the
other markers **are not supported within loops**. Simply because smcl log-file does not print
the output after each command in the loop. Nonetheless, writing markup text within the
loop is not recommended either because it only gets printed once. For active writing
within the loop or a program, see the <u>txt</u> command or <u>weaver</u> package.

Writing text in the do-file

As noted, MarkDoc package allows writing and styling text as a comment in the do file.
Text can be placed between **"/***"** and **"***/"** signs, where these signs are placed on
separate lines individually. Here is an example:

   /***

   Text heading
   _____

   subheading
   _____

   When you write a dynamic document in MarkDoc, place text between
   the "/***" and "***/" signs. But they should be placed on separate lines,
   as shown in this example.

   ***/

Hiding commands in dynamic document

Use **"/**/"** sign before each Stata command to hide it from the document. However, this sign
does not hide the command outputs, but only the command itself.

Here is an example:

  /**/ _sysuse auto_
  /**/ _regress price mpg_

Hiding output in dynamic document

Use **"/***/"** sign before each Stata command to hide its output in the dynamic document.
However, this sign does not hide the command itself, but only the output. In contrast to
<u>quietly</u> command which hides the output in the smcl log, the **"/***/"** sign only eliminates
the output in the dynamic document and the output will be registered in the smcl log file.
Using this marker is very similar to the example above.

Hiding a large part of the smcl file

If you want to register several commands and outputs in the smcl log, but you wish to
remove them from the dynamic document, begin the section you wish to hide with **"//OFF"** on
a separate line as shown below. MarkDoc will ignore anything that comes after this marker
until you specify **"//ON"** marker which will continue interpreting the smcl log. In contrast
to turning the <u>log off</u> and <u>log on</u> for ignoring part of the code in the document, these
markers allow you to save everything in the log file and yet, exclude them in the dynamic
document. Note that these markers cannot be written in Stata interactively and should be
placed or run from the do file.

   //OFF

   command
   command
   ...

   //ON

Supported markup languages

MarkDoc supports three markup languages which are [Markdown](#), [HTML](#), and [LaTeX](#). Whereas writing with Markdown allows exporting the document in any format (including HTML and LaTeX), writing with HTML or LaTeX requires exporting the document in **PDF** or **html** and **tex** format respectively. Markup languages hould not be used together in one document because MarkDoc process each markup language differently.

Markdown syntax

Writing with Markdown can make your script files appealing. It is a fairly minimalistic language which makes your text distinguishable from the computer code, in contrast to HTML and LaTeX, which add to the complexity of the code. To learn about using Markdown syntax for styling text and importing graphs, see [Writing with Markdown in Stata](#).

Software Installation

The MarkDoc package requires additional software which can be installed manually or automatically. The required software are [Pandoc](#) and [wkhtmltopdf](#) driver. They are both opensource freeware, supported for any common operating system such as Microsoft Windows, Macintosh, and Linux. Naturally, users who wish to use LaTeX markup for writing the documentation, will need a [TeX distribution with pdfLaTeX](#).

After a manual installation, the path to executable Pandoc should be specified in **pandoc(***str***)** option. Similarly, the path to executable wkhtmltopdf or pdfLaTeX should be given to **printer(***str***)** option. Note that the **printer(***str***)** option is only needed for compiling **pdf** document.

With automatic installation (i.e. using the **install** option), Pandoc and Wkhtmltopdf are downloaded and placed in Weaver directory which is located in /ado/plus/Weaver/ on your machine. To find the location of ado/plus/ directory on your machine use the [sysdir](#) command which returns the system directories. The usual complete paths to the Weaver directory are shown below. Note that username refers to your machine's username.

   Windows: *C:\ado\plus\Weaver*

   Macintosh: */Users/username/Library/Application Support/Stata/ado/plus/Weaver*

   Unix: */home/username/ado/plus/Weaver*

Set file paths permanently

After manual installation, the paths to the executable Pandoc, wkhtmltopdf, and pdfLaTeX can be permanently set using **weave setup** command. This command will open *weaversetup.ado* document, where you can define the files paths as global

   weave setup

Software troubleshoot

As mentioned, the required software can be installed manually or automatically. The optional automatic installation is expected to work properly in Microsoft Windows **XP,** Windows **7,** and Windows **8.1,** Macintosh **OSX 10.9.5,** Linux **Mint 17 Cinnamon** (32bit & 64bit), Ubuntu **14** (64bit), and **CentOS 7** (64bit). Other operating systems may require manual software installation.

However, if for some technical or permission reasons MarkDoc fails to download, access, or run Pandoc, install it manually and provide the file path to Pandoc using **pandoc(***str***)** option. visit [Installing Pandoc for Stata packages](#) for more information regarding manual installation of Pandoc.

Calling Pandoc

[Pandoc commands can also be executed from Stata](#). This command takes the path to the executable Pandoc from MarkDoc and allows you to use Pandoc seamlessly for converting files within Stata

   pandoc ./example.tex -o ./example.html

Remarks

If the log-file is closed exactly using **"qui log c"** command, Markdoc automatically removes this command from the end of the file.

Similarly, MarkDoc removes **"qui log off"** from the logfile. Therefore **"qui log off"** and **"qui log on"** can be used to separate codes in the dofile that are not wanted in the dynamic document, but still required in for the analysis. Nonetheless, this is not a proper practice and can harm the transparency of the analytic session. The log-file should include as much information about the history of the analysis as possible. Use the "Markers" for hiding sections of the log-file in the dynamic document.

Dynamic Document Examples

```
set linesize 90

    qui log using example, replace

            /***
            Introduction to MarkDoc (heading 1)
            ─────────────────────────────────────

            Using Markdown (heading 2)
            ────────────────────────────

            Writing with __markdown__ syntax allows you to add text and graphs to
            _smcl_ logfile and export it to a editable document format. I will demonstrate
            the process by using the __Auto.dta__ dataset.

            ###Get started with MarkDoc (heading 3)
            I will open the dataset, list a few observations, and export a graph.
            Then I will export the logfile to Microsoft Office docx format.
            ***/

    /***/ sysuse auto, clear
    /**/  list in 1/5
    histogram price
    graph export graph.png,  width(400) replace

            /***
            Adding a graph or image in the report
            ──────────────────────────────────────────

            Adding a graph using Markdown
            ───────────────────────────────────

            In order to add a graph using Markdown, I export the graph in PNG format.
            You can explain the graph in the "brackets" and define the file path in parentheses

            ![explain the graph](./graph.png)
            ***/

    qui log c

    markdoc example, replace export(html) install
    markdoc example, replace export(docx)
    markdoc example, replace export(tex) master
    markdoc example, replace export(pdf)
    markdoc example, replace export(epub)
```

Dynamic Slide Examples

```
    qui log using example, replace

            /***
            ───
            title:MarkDoc Dynamic Slides
            author: E. F. Haghish
            ───

            Slide 1
            ─────────

            - Writing with __markdown__ syntax allows you to add text and graphs
            to _smcl_ logfile and export it to a editable document format. I will demonstrate
            the process by using the __Auto.dta__ dataset.

            - I will open the dataset, list a few observations, and export a graph.
            Then I will export the logfile to Microsoft Office docx format.

            Adding commands and output
            ────────────────────────────────

            ***/
```

```
sysuse auto, clear
histogram price   graph export graph.png, width(400) replace

        /***
        Adding image in a slide
        ─────────────────────────────


        ![Histogram of the price variable](./graph.png)
        ***/

qui log c markdoc example, replace export(slide) install printer("/usr/texbin/pdflatex")
```

Author

E. F. Haghish
Center for Medical Biometry and Medical Informatics
University of Freiburg, Germany
and
Department of Mathematics and Computer Science
University of Southern Denmark
haghish@imbi.uni-freiburg.de

MarkDoc Homepage
Package Updates on Twitter

Also see

Weaver: HTML & PDF Dynamic Report producer

Statax: JavaScript syntax highlighter for Stata

This help file was dynamically produced by MarkDoc Literate Programming package