List of TEX commands

Martin Karrenbach [1]

/usr/local/src/our/word/tutorials/texcom/paper.tex

ABSTRACT

This is a list and brief description of the most commonly used TEX commands. The commands are given with arguments followed by a small description. By including "sepmacro.tex" you will get that functionality.

STYLISTIC COMMANDS

These commands govern some very basic behavior of your document; they do not contribute in any way to interactivity, but rather to the appearence of your document.

nopagenumbers	switch off pagenumbers
pageno{number}	set the page number to some arbitrary number
sepreport	set your paper style to a report conforms to Geophysics style guidelines
shortnote	set your paper style to a short note

[1] **email:** jon@sep.stanford.edu

STRUCTURING THE TEXT

For structuring a paper we have the following headlines prepared. Some print merely the headline, others do resetting of counters and internal TEX variables.

title{*title*}	set a title
author{*author name*}	set the author name
ABS{*text*}	put abstract in a box
INT	put heading: INTRODUCTION
DIS	put heading: DISCUSSION
CON	put heading: CONCLUSION
ACK	put heading: ACKNOWLEDGEMENT
REF	put heading: REFERENCES (better use **putbib**)
APP	put heading: APPENDIX A
APPENDIX{*name*}	put heading: APPENDIX name, also resets equation, figure and reference numbers and labels them correctly

INCLUDING OBJECTS

To make a document a living document you have to link the contents of your text with the actual object that you are describing in the text. We commonly include files such as programs and subroutines into the source code. A change in the outside source code is then directly reflected in the text itself. The paper and the source code live in symbiosis. Besides formatting and displaying the contents of the file in your text, the included object can be referenced.

listing{**filename**}	include a file in the text verbatim and teletype font very generic, no labels or captions, see **progblock**
keep{**filename**}	same as **listing**, but does not break across pages
prog{**label**} {**progname**} {**caption**}	include a program with a caption; label is local (**chaptername** is prepended) uses **PROGLABEL**
gprog{**label**} {**progname**} {**caption**}	include a program with a caption; label can be referenced globally, uses **GPROGLABEL**
progblock{**label**} {**progname**}	include a program with a caption; label is local (is prepended) uses **PROGLABEL**
gprogblock{**label**} {**progname**}	include a program with a caption; label can be referenced globally, uses **GPROGLABEL**
progref{**name**}	reference a page (**label**{**NAME**}**), in the following form:** `name()` **(page** **pagref**{**name**}**) , used for subroutines**
paper{**directory**}	include a paper in an SEP report (look at all.tex in the Adm directories of old reports); setting labels and counters.

LABELING AND REFERENCING

The previous commands insert objects. The following commands allow you to reference them uniquely in your document. The labels are unique by using different separators for different types. You can use the same name for equations, figures, programs or chapters. Which makes your text consistent from the authors and readers point of view. In case you need to have direct access to the unique labels, the default separators are:

- figures: "–"
- equations: "."
- programs: "+"
- chapters: "chapter="
- sections: "chapter=section="

Here is how you should reference the objects (instead of using **ref**).

\FIG{name}	reference a figure
\EQNLABEL{name}	set an equation label
\EQN{name}	reference an equation label
\PROGLABEL{name}	set a program label
\PROG{name}	reference a program label (teletype font)
\GPROGLABEL{name}	set a global program label
\GPROG{name}	reference a global program label (teletype font)
\CHAPLABEL{name}	set a chapter label
\CHAP{name}	reference a chapter label name within the same chapter + outside
\SECLABEL{name}	set a section label name
\SEC{name}	reference a section label name within the same chapter

OUTSIDE REFERENCING

For referencing these objects in a way that transcends chapters and sections of your document you have use these macros.

\CHAP{name}	reference chapter name
\CHAPSEC{chapter} {name}	reference section name in chapter chapter
\CHAPEQN{chapter} {name}	reference equation name in chapter chapter
\CHAPFIG{chapter} {name}	reference figure name in chapter chapter
\CHAPPROG{chapter} {name}	reference program name in chapter chapter

TEACHING USING THE ELECTRONIC BOOK

An electronic document can have exercises embedded in the text. These can be switched off, when printing the official version for a publisher, or can be turned on when teaching a class online.

\showiex	show interactive exercises in a book or paper
\noshowiex	switch off interactive exercises
\iex{directory} {file}	include file located in directory as an interactive exercise

FALLBACK COMMANDS

A few TeX macros have been redefined in the course of developing the living document. If you really need to use the original versions, don't despair, here are the synonyms.

oldlabel	saved copy of original latex labeling
oldref	saved copy of original latex reference
oldpageref	saved copy of original latex page reference

UTILITIES FOR INPUTTING FILES

The following commands are used by the higher level commands that include objects and lalbel them. You can use these also directly.

sepinput{file}	input file from the right place in a multi-authoring environment use **pathname** variable and/or **inputdir** variable
inputdir{directory}	sets input directory from which **sepinput** will take files if **inputdir** is not defined it defaults to **pathname**, if **pathname** is not defined default is ".”
includepf{directory} {file}	**include** file from directory, sets **pathname** to be the specified directory.

SETTING ATTRIBUTES AND VARIABLES

A few variables and attrributes can be set, that govern the interactive behaviour of the document.

interactive	sets interactive mode
notinteractive	no labels are shown, button clicks in text are disabled
def**figdir**{**directory**}	defines a figure directory (defaults to **pathname/Fig**)
pathname	is defined by **includepf**{**directory**} {**file**} **pathname is set to directory.**
nobutton	switch off buttons
withbutton	switch on buttons (default)

INTERACTION MECHANISM

We impose a structure on the interaction, in that we unify the mechanism in which each figure activates itself. The interaction mechanism for the included objects can be customized. By default the `cake` mechanism is used. Interaction behaviour can be modified by setting the approriate flag in yor document.

Authors have the choice to use, `cake`, `make` or `gmake`. (Note that **syscall** or **SYSCALL** command, executes the raw command without going through an interaction mechanism, besides the UNIX system call.)

UseCake	use cake mechanism (set flags by **def****cakeflag**{**...**})
UseMake	use make mechanism (set flags by **def****makeflag**{**...**})
UseGmake	use gmake mechanism (set flags by **def****gmakeflag**{**...**})

LOWEST LEVEL INTERACTION

You can place buttons in your text wherever you like. For that you have to use the following system call commands. To be safe the actual command should go to a specific directory and execute some commands.

(e.g. **\syscall{some text} {(cd \pathname ; echo hello world)}**) In this case it is assured that the commands is executed in the directory **\pathname**; a global variable that indicates where the document's directory is located. That is really important to a multi-paper/multi-author environment.

\syscall{text} {command}	shows text in small font and executes command
\SYSCALL{text} {command}	shows text in normal font and executes command

INCLUDING FIGURES

With Interaction

The most often used plot macros. Each take the same argument list.

activeplot{name} {height=...,width=...} {dir} {caption}	include postscript plot with file name "name"; give it optionally a certain height and width. Put up a button with the figurename "name"; when clicked on the button the command " cd dir ; cake name.idoc " is executed as a system call.
activeplothere{...}	(plot here)
activeplottop{...}	(plot at top of page)
activeplotbottom{...}	(plot at bottom of page)
activeplotpage{...}	(plot on a single page)
activesideplot{...}	(plot figure and caption side by side)
activebioplot{...}	(plot and label it as biography picture)

Without Interaction:

\passiveplot{name} {height=...,width=...} {dir} {caption}	
\passiveplothere{...}	(plot here)
\passiveplottop{...}	(plot at top of page)
\passiveplotbottom{...}	(plot at bottom of page)
\passiveplotpage{...}	(plot on a single page)
\passivesideplot{...}	(plot figure and caption side by side)
\passivebioplot{...}	(plot and label it as biography picture)

MISCELLANEOUS

\plotbox {name} {height=...,width=...}	(plot a figure on that spot without caption, good for filler pages or special plots)
\activeplotbox {name} {height=...,width=...}	(plot a figure on that spot without caption, good for filler pages or special plots); with interaction capability

Joe's Thesis plot macros

There is a fifth argument in the following macros, which is the caption that is added to the list of figures. Otherwise each argument is identical to the ones above.

Tactiveplot{...}	???
Tpassiveplot{...}	???