Stanford Exploration Project, Report 94, November 11, 1997, pages 339–362

Stanford Exploration Project, Report 94, November 11, 1997, pages 339–362

Short Note

Empowering SEP's documents

Sergey Fomel, Matthias Schwab, and Joel Schroeder[1]

keywords: *LaTeXe, LaTeX, TeX, latex2html, publication, WWW, Web*

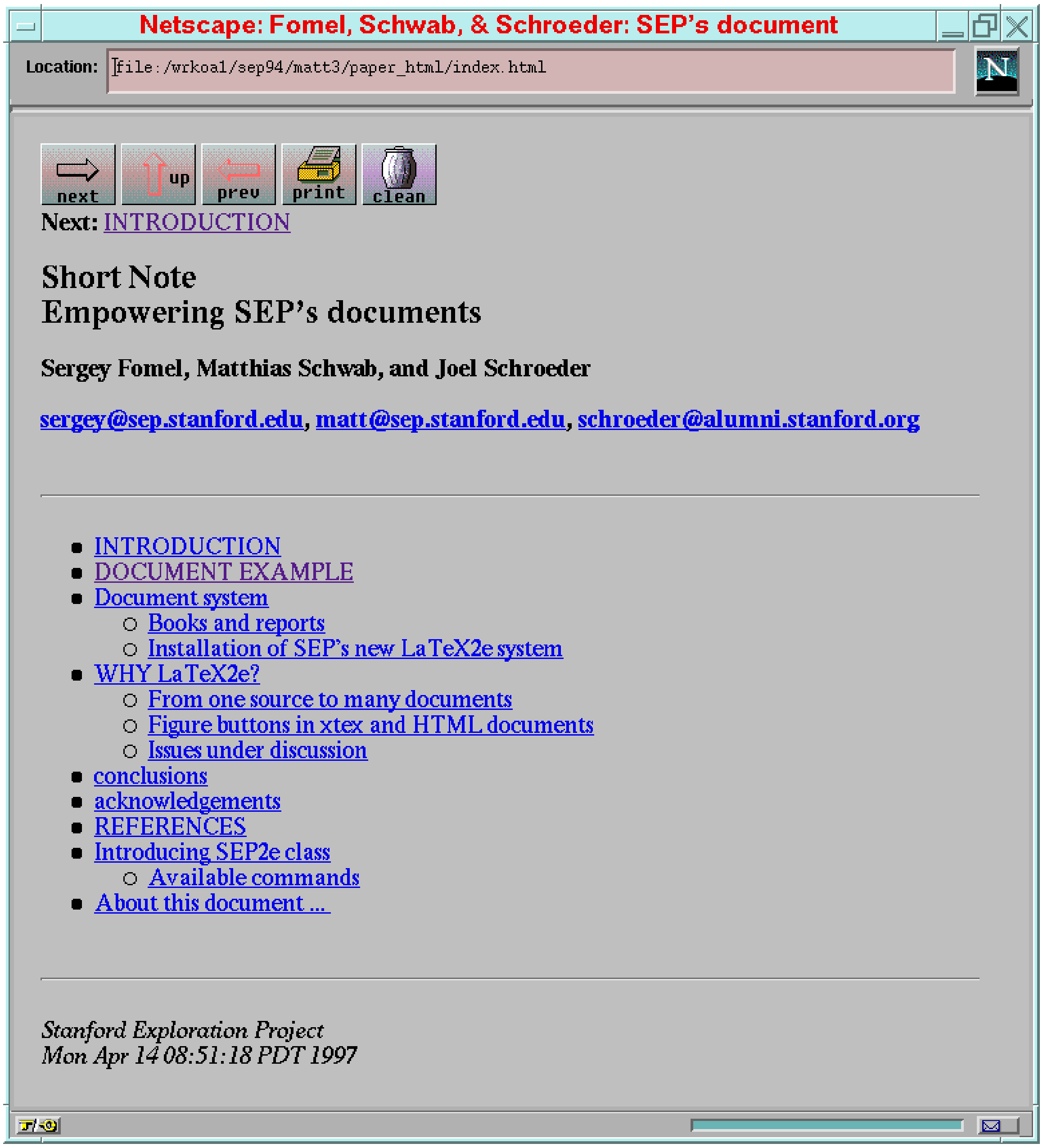
INTRODUCTION

The arrival of LaTeX 2_ε at SEP enhanced our LaTeX typesetting system and led us to overhaul SEP's customized macros. The revised system enables us to use the `latex2html` script (Drakos, 1996) to publish our documents routinely on the Internet. Additionally, we improved the communication between a document's makefile and its corresponding LaTeX file. Finally, we replaced a gigantic c-shell script (`texpr`) that governed SEP's entire document processing, by a set of small Perl scripts. These Perl scripts are easier to maintain because they deal more flexibly with a wide range of processing choices. We hope we can share our customized LaTeX 2_ε typesetting system with other communities such as SEG, TRIP (Rice University), or CWP (Colorado School of Mines.)

In this paper, we first present a short example of a reproducible SEP document. In the electronic version of the document, this article can be turned into an interactive or standard HTML version, several DVI versions, and, of course, a Postscript version. The second part of the article states the basic author and reader commands (make targets), discusses the most important new features, and lists unresolved issues and goals. In the appendix, we include an annotated table of all SEP macros.

Former SEP students, especially Martin Karrenbach, Joe Dellinger, and Kamal Al-Yahya, wrote custom made LaTeX (Lamport, 1994; Goosens et al., 1994) and TeX (Knuth, 1984) macros to enable *Geophysics*-style layout and reproducible figures. We have translated these SEP macros to LaTeX 2_ε, the modern LaTeX standard (Kopka and Daly, 1995). Authors at SEP can find a more elaborate document example in `/usr/local/src/our/sephelp/textour/paper.tex`.

[1]**email:** sergey@sep.stanford.edu, matt@sep.stanford.edu, schroeder@alumni.stanford.org


Netscape: Fomel, Schwab, & Schroeder: SEP's document
Location: file:/wrkoa1/sep94/matt3/paper_html/index.html
next
up
prev
print
clean
Next: INTRODUCTION
Short Note
Empowering SEP's documents
Sergey Fomel, Matthias Schwab, and Joel Schroeder
sergey@sep.stanford.edu, matt@sep.stanford.edu, schroeder@alumni.stanford.org
INTRODUCTION
DOCUMENT EXAMPLE
Document system
Books and reports
Installation of SEP's new LaTeX2e system
WHY LaTeX2e?
From one source to many documents
Figure buttons in xtex and HTML documents
Issues under discussion
conclusions
acknowledgements
REFERENCES
Introducing SEP2e class
Available commands
About this document ...
Stanford Exploration Project
Mon Apr 14 08:51:18 PDT 1997

DOCUMENT EXAMPLE

In this section, we imitate a miniature SEP research paper. At SEP, such a research paper not only includes an article with figures, equations, and references, but also a set of accompanying programs, input data, and makefiles. In its electronic form such a research document enables the reader to reproduce the results that the document presents (Schwab et al., 1996).

The research document that we embedded for documentation purposes implements Normal Moveout, a standard seismic processing step. The equation that governs the Normal Moveout is a mapping from time t to zero-offset time t_0

$$t^2 = t_0^2 + \frac{h^2}{V^2} \tag{1}$$

where h is the offset and V the subsurface velocity. We copied the underlying program and the following result from Jon Claerbout (1991).

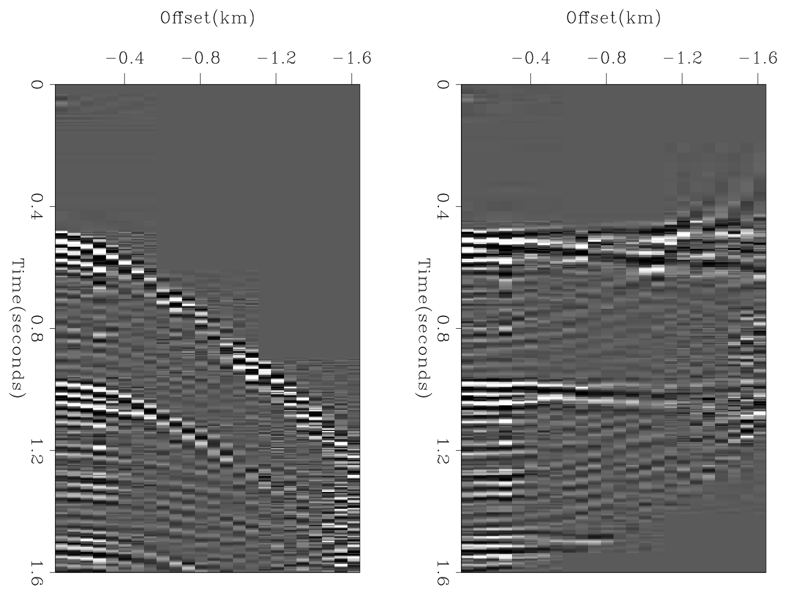
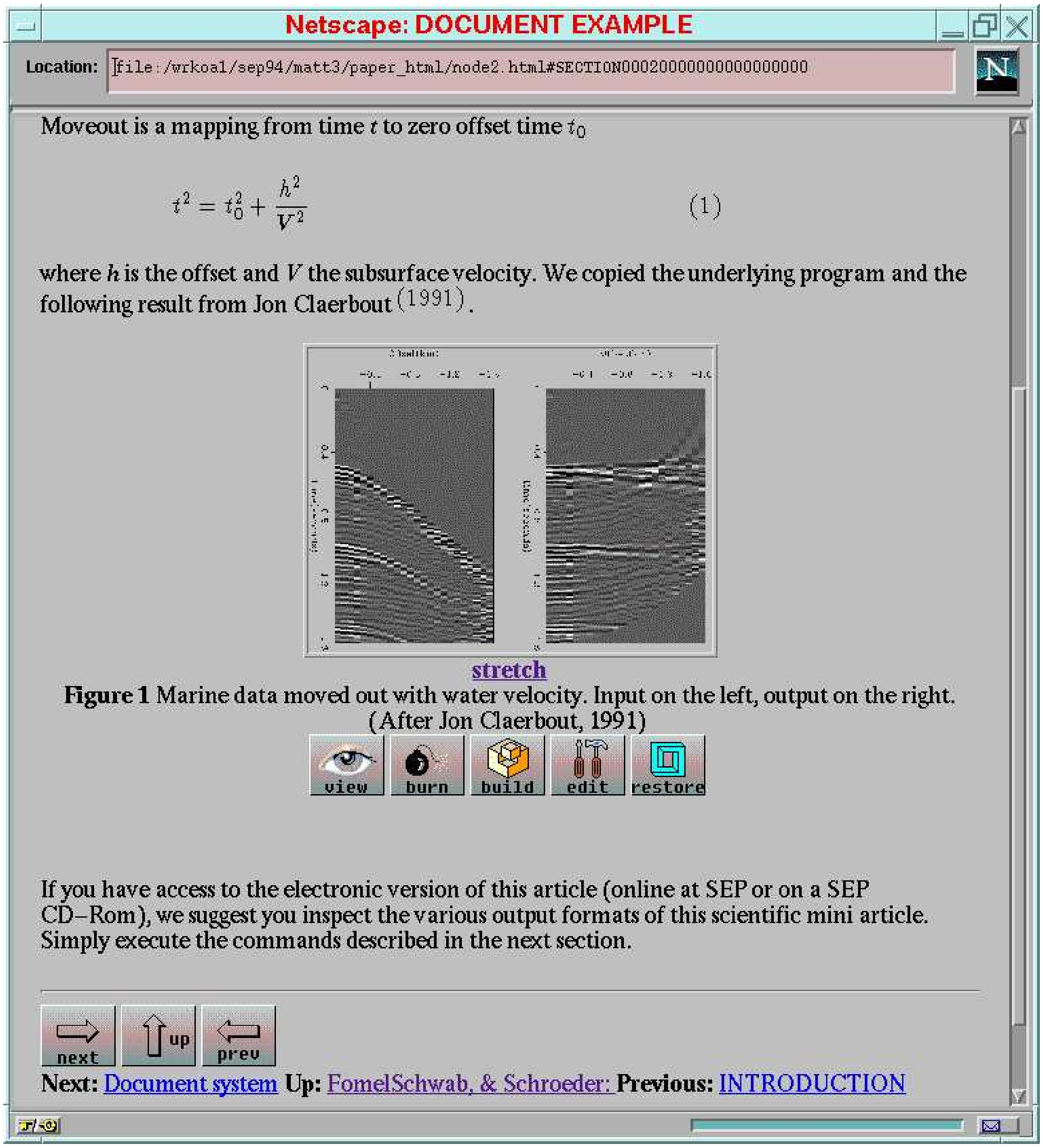


Figure 1: Marine data moved out with water velocity. Input on the left, output on the right. (After Jon Claerbout, 1991) matt3-stretch [ER]

If you have access to the electronic version of this article (online at SEP or on a SEP CD-ROM), we suggest you inspect the various output formats of this scientific mini article. Simply execute the commands described in the next section.

DOCUMENT SYSTEM

SEP's make rules provide standard commands to create the various document versions. When within the SEP environment (at an SEP computer or on a SEP CD-



Netscape: DOCUMENT EXAMPLE

Location: file:/wrkoa1/sep94/matt3/paper_html/node2.html#SECTION00020000000000000000

Moveout is a mapping from time t to zero offset time t_0

$$t^2 = t_0^2 + \frac{h^2}{V^2} \qquad (1)$$

where h is the offset and V the subsurface velocity. We copied the underlying program and the following result from Jon Claerbout (1991).

stretch

Figure 1 Marine data moved out with water velocity. Input on the left, output on the right. (After Jon Claerbout, 1991)

view burn build edit restore

If you have access to the electronic version of this article (online at SEP or on a SEP CD–Rom), we suggest you inspect the various output formats of this scientific mini article. Simply execute the commands described in the next section.

next up prev

Next: Document system **Up:** FomelSchwab, & Schroeder: **Previous:** INTRODUCTION

ROM), go to any SEP document, type one of the following commands, and inspect the various results:

gmake paper.ps creates a Postscript file.

gmake paper.gs displays a Postscript file with `ghostview`.

gmake paper.dvi creates a standard DVI file.

gmake paper.read creates the DVI file and displays it with `xdvi`.

gmake paper.print creates the DVI file and prints it to the default printer.

gmake paper.ltx converts an SEP style TEX file into a proper LATEX file, paper.ltx. The paper.ltx file is helpful for debugging since the line numbers of TEX's log file refer to it.

gmake paper.slide creates a DVI file using an official slide macro set.

gmake paper.jslide creates a standard SEP DVI file but with an reduced page width of about 40 characters. Jon Claerbout likes to make slides from this DVI file by enlarging the text on a copy machine.

A reader or author may want to try some of these more experimental document commands:

gmake paper.html creates an HTML subdirectory (`paper_html`) that includes an HTML version of the LATEXpaper. The make target will invoke the `latex2html` script. The rule uses standard X11 programs to generate the necessary `.gif` figures for equations and figures. We still have to define a set of adequate "clean" rules.

gmake paper.browse updates the `paper_html` directory and starts-up a browser.

gmake paper.idvi creates a DVI version that can be rendered by IDVI, a Java renderer. IDVI is an experimental alternative to `latex2html`.

gmake paper.xtex displays a DVI file with `xtex`. This command uses an old set of SEP's LATEX macros. You may have to remove some new macros, such as keywords. Additionally, `xtex` does not find the activeplot figures correctly. In the near future, we probably have to remove the paper.xtex rule.

The targets above can be applied to text source files with any name, as long as the name ends with the suffix `.tex`. For example, to create a Postscript version from a text source file named `mytext.tex`, you execute `make mytext.ps`.

In general, the new $\LaTeX\,2_\varepsilon$ macros are backward compatible. You should be able to process any old SEP $\LaTeX$ file with the new SEP $\LaTeX\,2_\varepsilon$ macros. The only known backward incompatibility is that the new activeplot macro always requires a width or height statement.

To ease the maintenance of reproducible documents, we improved the communication between the document's makefile and its $\LaTeX$ file. Consequently, the author does not need to define the result directory in the $\LaTeX$ file (previously: `\def\figdir{name}`.) Instead, the new document processing system requests that information from your makefile. Try processing an old SEP $\LaTeX$ document after removing the figdir statement.

A SEP $\LaTeX\,2_\varepsilon$ document forces an author to supply a keyword list with the help of the macro `\keywords{}`. The keyword list is printed below the list of authors and will help us to index SEP documents in the future. We are planning to supply a set of keywords that an author can choose from (e.g., the SEG keywords for expanded abstracts.)

Finally, we added an enhanced clean rule, `jdistclean`, to our standard make rules. The distclean rule creates a minimal set of files: In addition to the intermediate files that are removed by the clean rule, it also removes all reproducible result files and any HTML directories.

Books and reports

The standard $\LaTeX\,2_\varepsilon$ book style simplifies the generation of books and reports. The book format solves the $\TeX$ memory limitation problem that we frequently experienced in the past when printing our reports. The chapters and their order are defined in a makefile. The table of contents is generated automatically by querying the titles of the included chapters. This organization will free us from storing all included chapters in sister directories.

The system for creating and maintaining books and reports consists of a small set of book-specific make rules (`Doc.rules.book`), empowered by set of simple perl scripts for automatic creation of the book attributes (table of contents, title page, author's biography.)

Installation of SEP's new $\LaTeX\,2_\varepsilon$ system

To download SEP's new $\LaTeX\,2_\varepsilon$ setup, follow instructions on our web page at `<http://sepwww.stanford.edu/softtex/>`. The page includes instructions as well as URLs to download the corresponding tar files. The web page also includes a Frequently Asked Question (FAQ) table, and we invite users to add additional helpful information. Additionally, we distribute our entire document system on our biannually released CD-ROMs.

WHY LaTeX 2_ε?

The LaTeX community has adapted the LaTeX 2_ε standard. New typesetting tools, such as the `latex2html` script, the java applet IDVI, or PDF distiller, expect LaTeX 2_ε. SEP needs these typesetting tools to publish on the web. Additionally, LaTeX 2_ε's clear package and class organization, as well as its ease of installation, should reduce SEP's TeX maintenance.

We had to translate the SEP-specific macros from LaTeX to LaTeX 2_ε dialect. In our translation, we aimed at two contradictory goals: to ease the portability of SEP documents and to be fully backward compatible.

To ease the portability of SEP documents, we developed a minimal set of SEP macros. SEP's minimal macro set contains only a third (26) of the traditional SEP macro family (78). This reduced macro set mostly deals with the inclusion of Postscript figures (activeplot), the inclusion of source code files (gprogblock), with author information (author, email, keywords), and SEP style definitions[2]. All other commands are standard LaTeX 2_ε commands. We hope that the minimal macro set will enable us to share our LaTeX 2_ε macros easily with other organizations that have similar needs (SEG, TRIP at Rice University, and CWP at Colorado School of Mines.)

We recommend this minimal set for future SEP documents. To restrict the processing of his document to the minimal set, an author needs to define `LATOPTS = option=minimal` in his document's makefile. However, the current default ensures backward compatibility since it uses the superset of traditional SEP macros. This backward compatibility will, we hope, simplify the transition to the new document system.

Why do we use LaTeX rather than any other text or word processing system such as Microsoft Word? We have used LaTeX in the past and all students at SEP can write LaTeX documents. We also have a legacy of LaTeX documents. LaTeX still seems to be the method of choice for mathematical typesetting, but we have not seriously tested some attractive-sounding alternatives, such as framemaker[3]. Additionally, since LaTeX 2_ε is freely distributed, we can include it on SEP's interactive CD-ROMs.

From one source to many documents

We wrote three relatively short Perl scripts and a few make rules to enable the author to translate his document file, paper.tex[4], to his favorite output format, such as Postscript, HTML, DVI, etc. The three Perl scripts – Actify, Latify, and latex2dvi

[2] These style definitions follow SEG's style definitions for publications in its *Geophysics* journal.
[3] http://www.adobe.com/
[4] At SEP we call this file `paper.tex`, despite the fact that the file is actually a slightly incomplete LaTeX file, not a TeX file.

– are extensible and easily maintained. These scripts replace SEP's former C-shell script, `texpr`.

Actify extracts relevant information, such as `RESDIR` (the result directory) and `ER, CR, NR` (reproducibility status), from the makefile and includes it in the document file. Until now, authors had to state the `RESDIR` in both the document file and the makefile. The reproducibility status was clumsily communicated via the creation of dummy files.

Latify turns the document source file into a complete LaTeX file. Latify adds a document style statement, a begin and end document statement, etc. The script's output is a standard LaTeX file that uses SEP macros.

latex2dvi translates standard LaTeX to a device independent format (DVI). Basically, it runs `latex, bibtex, makeindex` as many times as necessary.

This set of Perl scripts currently produces a family of output formats of the original document: The most common formats are DVI versions, that can be displayed with a DVI reader (xdvi) or converted to Postscript (dvips). For the time being, we additionally support an enhanced DVI format based on the xtex DVI reader. Xtex permits us to display DVI files that have buttons in the figure captions. When pressed, these buttons bring up a graphics user interface that enables the reader to interact with the figure. We hope to replace xtex by an alternative interactive renderer, since xtex is non-standard and hard to maintain.

Additionally, we use the freely available `latex2html` script, which translates standard LaTeX to HTML. We enhanced the latex2html Perl script to process certain non-trivial SEP macros into their HTML implementations. In particular, SEP's figure macro `activeplot` is translated by a customized Perl subroutine (`latex2html` explicitly endorses such enhancements.) In the next section, we discuss our attempt to create interactive HTML pages for SEP's reproducible documents.

Additionally, we are currently testing an alternative web solution based on the Java DVI renderer, IDVI (`<http://www.geom.umn.edu/java/idvi/>`). IDVI allows a reader to interact with a document through embedded Java applets.

Figure buttons in xtex and HTML documents

The button in the caption of Figure 1 (xtex or HTML version) is the entry point to the interactive reproduction of the article's result. The button in the traditional xtex version of this paper will issue a makefile command that in return invokes an xtpanel (Cole and Nichols, 1992) graphical user interface. The GUI offers the four standard commands of SEP's reproducible research: to view, burn, build, and clean the document.

In the HTML version of the paper, the reader can execute these standard commands by pressing buttons beneath the figure. To enable Netscape to issue certain commands (make invocations of the targets view, burn, build, and clean), we introduced a `sepshell` scripy and added an application mime type to the mailcap file. These `sepshell` programs issue the corresponding GNU-make commands on the local browser client. We code the absolute path of a document into a figure's HTML representation. The HTML figure buttons use this absolute pathname to connect to the document's corresponding makefile. In practice, this mechanism limits interactive HTML versions of reproducible documents to SEP's local machines, unless a reader downloads and installs the SEP environment on his own computer or runs it from a CD-ROM.

In addition to the traditional buttons that relate to the reproduction of a document's results, `latex2html` adds an edit and reset button. The edit button displays a document's makefile for editing. Other files can be opened and changed with the editor. If the document is maintained under RCS version control (you can find more about RCS at any GNU software site, e.g. `<http://www.delorie.com/gnu/>`), then the restore button will reset all files to the original state. This feature might be useful for teaching courses, but as a default we plan to disable the edit and restore button to avoid accidental changes of the underlying software.

A simple `nobutton` definition in the paper.tex file will result in a straightforward HTML or xtex document without any interactive buttons.

Issues under discussion

- James Rickett and Matthias Schwab want smaller buttons in the HTML document. What should the default buttons look like? A full set of buttons includes `up, next, previous, up-inactive, next-inactive, previous-inactive` (for navigation) `view, burn, build, edit, restore, print, clean, warning` (for `sepshell` interaction.)
- James Rickett points out that the figure reference takes you to the figure caption. Usually, a reader needs to scroll up his page to see the figure. Joel Schroeder tried to fix this but did not succeed.
- We hope we can soon demonstrate a reproducible web document implemented in Java.
- We wish there was a Java version of `make` (preferably called *jake*.)
- It would be nice if the HTML version of a document would contain all WWW-references as hyper-links. Do we need the Hyper-TeX package or just the syntax? Can we use the Hyper-TeX extensions in the bibliography files?
- Should we suppress TEX's very verbatim processing messages? Sergey prefers the interactive mode, since it enables him to ignore small errors while processing.

- If we gave all encapsulated Postscript figures the suffix *eps* instead of *ps*, then we could distinguish between figure.ps and paper.ps files in our clean rule (Tariq Alkhalifah suggested the *eps* suffix for a different reason.)

CONCLUSIONS

SEP has successfully switched from LaTeX to LaTeX 2_ε. The new system permits us to experiment with new Web representation and distribution of our research documents. Additionally, we hope the new system will be easier to maintain and to extend.

ACKNOWLEDGEMENTS

We especially want to thank Martin Karrenbach for his wealth of macros and his continuous effort to move SEP to LaTeX 2_ε. Martin also introduced us to IDVI. Thanks to James Rickett, Robert Clapp, and Biondo Biondi for their testing and suggestions.

REFERENCES

Claerbout, J. F., 1991, Earth Soundings Analysis: SEP–**71**, 1–304.

Cole, S., and Nichols, D., 1992, Xtpanel: An interactive panel builder: SEP–**75**, 497–520.

Drakos, N., 1996, All about LaTeX2HTML: http://cbl.leeds.ac.uk/ nikos/tex2html/doc/latex2html/latex2html.html.

Goosens, M., Mittelbach, F., and Samarin, A., 1994, The LaTeX companion: Addison-Wesley Publishing Company.

Knuth, D., 1984, TeX book: Addison-Wesley Publishing Company.

Kopka, H., and Daly, P. W., 1995, A guide to LaTeX 2_ε: Document preparation for beginners and advanced users: Addison-Wesley Publishing Company.

Lamport, L., 1994, LaTeX: A document preparation system 2/e: Addison-Wesley Publishing Company.

Schwab, M., Karrenbach, M., and Claerbout, J., 1996, Making scientific computations reproducible: SEP–**92**, 327–342.

APPENDIX A

INTRODUCING SEP2E CLASS

LaTeX 2_ε allows users to add their own classes to the standard classes (`article`, `letter`, `book`, `report`, etc.), supplied in the base LaTeX 2_ε distribution. We define the basic macro for SEP documents as a class (`.cls` file) rather than a package (`.sty` file) to avoid conflicts with other packages and because it has a predefined notion of the base class (explained below.)

In the design of the `sep2e` class, we attempted to

- preserve maximum back-compatibility with the older macros (`sepreport.sty`)
- keep it as small and simple as possible,
- use object-oriented tricks whenever possible,
- use standard LaTeX 2_ε packages whenever appropriate.

The `sep2e.cls` file is accompanied by a perl script `sep2e.pl`, which assists `latex2html` in translating class commands to their HTML equivalents.

The user invokes `sep2e.cls` with the command `\documentclass[options]{sep2e}`[5] The class options are:

alone This option is the default. It means *process the file as a stand-alone article.*

report This option is used to typeset SEP's bi-annual reports.

book This option is used to typeset books in the style of Jon Claerbout's trilogy.

minimal This option helps the user to restrain himself from superfluous SEP's extensions to LaTeX (otherwise, fully supported for backward compatibility.) It is required by `latex2html`.

Option `alone` loads the LaTeX `article` class into `sep2e`. It means that all commands and definitions of the official `article` class are available to the document, processed with the `alone` option unless they were redefined in the `sep2e` class. Likewise, the other two option load the `book` class into `sep2e`[6]. All other options of `\documentclass` are passed directly to the corresponding base class (article or book.) For example, to process a book with 12-point fonts, you can use `\documentclass[book,12pt]{sep2e}`.[7]

[5]Usually, this line is inserted for the author by SEP's `Latify` script.

[6]We are going to add a `thesis` option eventually.

[7]The `Latify` script inserts the `12pt` option by default.

Available commands

The tables below list the SEP LaTeX2e macros.

- *No* in the *minimal* column means that the command is supported, but not recommended (it remains for backward-compatibility only.) You are encouraged to use the alternative implementation.
- #1, #2, etc. refer to the command arguments in the order of their appearance.

General formatting commands (headings,sections, etc.)			
Command	*minimal*	*What it does*	`latex2html` *translation*
`\sepreport`	Yes	Sets the normal article layout. This command is invoked by default.	Analogous.
`\shortnote`	Yes	Sets the short-note article layout.	Analogous.
`\morespace`	No	Increases the inter-line spacing to 1.5.	N/A
			`\renewcommand{\baselinestretch}{1.5}`
`\doublespace`	No	Increases the inter-line spacing to 2.	N/A
			`\renewcommand{\baselinestretch}{2.0}`
`\nopagenumbers`	No	Switches off the page numbering.	N/A
			`\renewcommand{\thepage}{}`
`\bold{}`	Maybe	In the math mode, sets the argument in the bold face.	Emulates the same behavior by inclosing the argument between `<B>` and `</B>`.
			`\ensuremath{\mathbf{#1}}`
`\ital{}`	No	Sets the argument in italics.	N/A
			`\textit{#1}`
`\righthead{}`	Yes	Sets the right heading (normally a short name of the article.) Ignored for the `book` option.	Includes it in `<TITLE>`.
`\lefthead{}`	Yes	Sets the left heading (normally author's last name.) Ignored for the `book` option.	Includes it in `<TITLE>`.

`\footer{}`	Yes	Sets the footer. Ignored for the `book` option.	Ignored.
`\email{}`	Yes	Sets the email string. The default string is *Author has no known email address.* Should be applied before `\author`.	Analogous. Additionally, the email string is broken into individual e-mail addresses (for multiple authors.)
`\noemailaddress`	Yes	Switches off displaying of the e-mail string.	Analogous.
`\keywords{}`	Yes	Sets the keyword string. The default string is *not available.* Should be applied before `\author`.	Ignored (for now.)
`\nokeywords`	Yes	Switches off displaying of the keyword string.	Ignored.
`\title{}`	Yes	Under `alone` and `report` options, this LaTeX command is redefined to display the paper's title. If `\shortnote`, the title is left-justified (instead of centering) and prepended with "Short Note".	Analogous. The title string is placed between `<H1>` and `</H1>`.
`\author{}`	Yes	Under `alone` and `report` options, this LaTeX command is redefined to display the author(s) name. The keyword string is displayed under the author (unless `\nokeyword`.) The email string is displayed as a footnote (unless `\noemailaddress`)	Analogous. The keyword string is ignored (for now), the email string is displayed under the author(s) name as hyperlink(s) to `mailto:` (James Rickett's idea.)
`\chapter{}`	Maybe	Under `alone` options, this LaTeX command is equivalent to `title`.	Analogous.
`\mhead{}`	No	Equivalent to `\section{}`. Under `alone` and `report` options, the `\section` command has been redefined to reflect *Geophysics* style (capital letters, centered headings.)	N/A

`\section{#1}`			
`\shead{}`	No	Equivalent to (redefined) `\subsection{}`.	N/A
`\subsection{#1}`			
`\sshead{}`	No	Equivalent to (redefined) `\subsubsection{}`.	N/A
`\subsubsection{#1}`			
`\ssshead{}`	Maybe	Supersmall heading (rarely used.)	N/A
`\smallskip\begin{flushleft}\footnotesize\textbf{#1}\end{flushleft}`			
`\sssshead{}`	Maybe	Extrasmall heading.	N/A
`\smallskip\begin{flushleft}\scriptsize\textbf{#1}\end{flushleft}`			
`\boxit{}`	Maybe	Puts the text in the argument in a frame.	Analogous (uses a bordered table.)
`\ABS{}`	Yes	Puts abstract in the argument in a frame (*Geophysics* style.) Ignored if `\shortnote`.	Analogous (uses a bordered table.)
`\parab`	No	Obsolete. It was used previously to define paragraphs in the abstract, but is no longer needed.	N/A
`\INT`	No	Starts section INTRODUCTION.	N/A
`\section{Introduction}`			
`\DIS`	No	Starts section DISCUSSION.	N/A
`\section{Discussion}`			
`\CON`	No	Starts section CONCLUSIONS.	N/A
`\section{Conclusions}`			
`\ACK`	No	Starts section ACKNOWLEDGMENTS.	N/A
`\section{Acknowledgements}`			
`\REF`	No	Starts section REFERENCES.	N/A
`\section{References}` or `\section*{REFERENCES}` (for books)			

`\APPENDIX{}`	Yes	Starts an appendix. The argument is normally a capitalized letter. Equations and figures in an appendix have different numeration and are prepended with the appendix name.	Analogous. The translation is a little buggy right now, should be fixed eventually.
`\APP`	No	Starts appendix A.	N/A
			`\APPENDIX{A}`
`\booktoreport`	No	Obsolete. Now chapters in books and articles in reports can have the same source, interpreted differently according to the `sep2e` options.	N/A
`\reporttobook`	No	Obsolete. Now chapters in books and articles in reports can have the same source, interpreted differently according to the `sep2e` options.	N/A

Labeling and referencing commands

Command	*minimal*	*What it does*
`\oldlabel{}`	No	Same as `\label{matt3/}`.
		`\label{#1}`
`\oldref{}`	No	Same as `\ref{}`.
		`\ref{#1}`
`\oldpageref{}`	No	Same as `\pageref{}`.
		`\pageref{#1}`
`\FIGLABEL{}`	No	Labels a figure. Equivalent of this command is used internally by `\activeplot` and `\activesideplot`.
		`\label{fig:#1}`
`\FIG{}`	No	References a figure, labeled by `\FIGLABEL`.
		`\ref{fig:#1}`
`\CHAPFIG{}{}`	No	References a figure in a different chapter. The first argument is the directory name of the referenced chapter.
		`\ref{#1/fig:#2}`

`\EQNLABEL{}`	No	Labels an equation.
		`\label{eqn:#1}`
`\EQN{}`	No	References an equation, labeled by `\EQNLABEL`.
		`(\ref{eqn:#1})`
`\CHAPEQN{}{}`	No	References an equation in a different chapter. The first argument is the directory name of the referenced chapter.
		`(\ref{#1/eqn:#2})`
`\PROGLABEL`	No	Labels a program. This command is used internally by `\prog` and `\progblock`.
		`\label{prog:#1}`
`\PROG{}`	No	References the page of a program, labeled by `\PROGLABEL`, and prints the name of the program with `()`.
		`\texttt{#1()} \vpageref{prog:#1}`
`\GPROGLABEL{}`	No	Labels a program. This command is used internally by `\gprog` and `\gprogblock`.
		`\label{/prog:#1}`
`\GPROG{}`	No	References the page of a program, labeled by `\GPROGLABEL`, and prints the name of the program with `()`.
		`\texttt{#1()} \vpageref{/prog:#1}`
`\LPROG{}`	No	Prints the name of a program with `()` without referencing its page.
		`\texttt{#1()}`
`\CHAPLABEL{}`	No	Labels a chapter.
		`\label{paper:#1}`
`\CHAP{}`	No	References a chapter.
		`\ref{#1/paper:#1}`
`\chapterlabel{}`	No	The same as `\CHAPLABEL`.
		`\label{paper:#1}`
`\chapterref{}`	No	The same as `\CHAP`.
		`\ref{#1/paper:#1}`

File input commands

Command	*minimal*	*What it does*	`latex2html` *translation*
`\inputdir{}`	Maybe	Defines the directory name for the files being included. The default is the current directory. One of the "low-level" commands.	Analogous.
`\sepinput{}`	No	Inputs a file from the directory set by `\inputdir`.	N/A
			`\input{#1}`
`\FloatListing`	Yes	Sets a flag so that included programs will behave as floating figures.	Ignored.
`\NoFloatListing`	Yes	Sets a flag so that included programs will NOT behave as floating figures. This command is invoked by default.	Ignored.
`\BreakListing`	No	The same as `\NoFloatListing`.	N/A
`\NoBreakListing`	No	The same as `\FloatListing`.	N/A
`\listing{}`	Yes	Inputs a program in a small font and verbatim style.	Analogous (uses <PRE>.)
`\gprog{}{}{}`	Yes	The first argument is a label, which can be referenced with `\GPROG`. The second argument is the program name. The third argument is the caption text. The letter "g" stands for *global*. It means that the program is supposed to be referenced in other chapters of a book or other papers in a report.	Analogous.
`\prog{}{}{}`	Yes	The first argument is a label, which can be referenced with `\PROG`. The second argument is the program name. The third argument is the caption text.	Analogous.
`\gprogblock{}{}`	Yes	The first argument is a label, which can be referenced with `\GPROG`. The second argument is the program name. Observes the flag set by `\FloatListing` or `\NoFloatListing`. The letter "g" stands for *global*. It means that the program is supposed to be referenced in other chapters of a book or other papers in a report.	Analogous.

`\progblock{}{}`	Yes	The first argument is a label, which can be referenced with `\PROG`. The second argument is the program name. Observes the flag set by `\FloatListing` or `\NoFloatListing`.	Analogous.

Bibliography commands			
Command	*minimal*	*What it does*	`latex2html` *translation*
`\cite{}`	Yes	Puts a bibliography reference in the form `(author, year)`.	Translates into a transparent GIF image.
`\shortcite{}`	Yes	Puts a bibliography reference in the form `(year)`.	Translates into a transparent GIF image.
`\longcite{}`	Yes	Puts a bibliography reference in the form `author (year)`.	Translates into a transparent GIF image.
`\putbib[]`	No	Inserts a bibliography list.	N/A
			`\bibliographystyle{sep}\bibliography{#1}`
`\reference{}`	Maybe	Emulates an entry in a bibliography list.	Unimplemented yet. Ignored for now.

Plotting commands			
Command	*minimal*	*What it does*	`latex2html` *translation*
`\withbutton`	No	Sets a flag to include figure buttons.	N/A
`\nobutton`	No	Sets a flag NOT to include figure buttons. The same effect can be achieved with `\plot` and `\sideplot`.	N/A
`\syscall{}{}`	No	Obsolete. Used previously in conjunction with `xtex`.	Ignored.
`\SYSCALL{}{}`	No	Obsolete. Used previously in conjunction with `xtex`.	Ignored.

`\plotbox[]{}{}`	Yes	The first (optional) argument is the relative directory name, defaulted to `\figdir`. The second argument is the figure name. The third argument contains plot options (typically `width=`, `height=`). Use it for figures, which are not defined in RESULTS lists of the makefile.	Analogous. Puts a corresponding GIF figure in a bordered table. The size of the figure is computed from the plot options, assuming the resolution of 75 dpi.
`\plot[]{}{}{}`	Yes	The first (optional) argument specifies the position of the figure and is defaulted to `htbp` (here, top, bottom, page). The second argument is the figure name. The third argument contains plot options (typically `width=`, `height=`). The fourth argument contains the caption text.	The corresponding GIF figure is placed in a bordered table. The size of the figure is computed from the plot options, assuming the resolution of 75 dpi. A hyperlink to the full-sized GIF image is placed under the figure. *Problem:* the hyperreference to the figure goes under it. Sometimes it is inconvenient, and we would prefer the reference to be on top of the figure (pointed out by James Rickett.)

`\activeplot[]{}{}{}{}`	Yes	The first (optional) argument specifies the position of the figure and is defaulted to `htbp` (here, top, bottom, page). The second argument is the figure name. The third argument contains plot options (typically `width=`, `height=`). The fourth argument indicates the reproducibility status and is inserted by `Actify`. The fifth argument contains the caption text. The figure name is placed in a frame, and the reproducibility flag is placed in `[]`.	The corresponding GIF figure is placed in a bordered table. The size of the figure is computed from the plot options, assuming the resolution of 75 dpi. A hyperlink to the full-sized GIF image is placed under the figure. The figure is supplied with interactive `sepshell` buttons.
`\sideplot[]{}{}{}`	Yes	Analogous to `\plot`, but the figure has a fixed maximum width, and the caption is placed on the left side of the plot. The first (optional) argument specifies the position of the figure and is defaulted to `htbp` (here, top, bottom, page). The second argument is the figure name. The third argument contains plot options (typically `width=`, `height=`). The fourth argument contains the caption text.	The corresponding GIF figure is placed in a table with the caption on the left side. The size of the figure is computed from the plot options, assuming the resolution of 75 dpi. A hyperlink to the full-sized GIF image is placed under the figure. *Problem:* the hyperreference to the figure goes under it. Sometimes it is inconvenient, and we would prefer the reference to be on top of the figure (pointed out by James Rickett.)

`\activesideplot[]{}{}{}{}`	Yes	Analogous to `\activeplot`, but the figure has a fixed maximum width, and the caption is placed on the left side of the plot. The first (optional) argument specifies the position of the figure and is defaulted to `htbp` (here, top, bottom, page). The second argument is the figure name. The third argument contains plot options (typically `width=`, `height=`). The fourth argument indicates the reproducibility status and is inserted by `Actify`. The fifth argument contains the caption text. The figure name is placed in a frame, and the reproducibility flag is placed in `[]`. **Warning:** Previously we were able to use something like `\activesideplot{name}{2in}{}{caption}`, where the length `2in` was interpreted as the figure height. Now you have to spell out `height=2in`. This is the only serious back-compatability problem that we have encountered.	The corresponding GIF figure is placed in a table with the caption on the left side. The size of the figure is computed from the plot options, assuming the resolution of 75 dpi. A hyperlink to the full-sized GIF image is placed under the figure. The figure is supplied with interactive `sepshell` buttons. *Problem:* the hyperreference to the figure goes under it. Sometimes it is inconvenient, and we would prefer the reference to be on top of the figure (pointed out by James Rickett.)
`\passiveplot{}{}{}{}`	No	Obsolete. Equivalent to `\activeplot`.	N/A
`\passivesideplot{}{}{}{}`	No	Obsolete. Equivalent to `\activesideplot`.	N/A

`\activeplothere{}{}{}{}`	No	Equivalent to `\activeplot` with the first (optional) argument set to `h` (here).	N/A
			`\activeplot[h]{#1}{#2}{#3}{#4}`
`\activeplotpage{}{}{}{}`	No	Equivalent to `\activeplot` with the first (optional) argument set to `p` (page).	N/A
			`\activeplot[p]{#1}{#2}{#3}{#4}`
`\activeplottop{}{}{}{}`	No	Equivalent to `\activeplot` with the first (optional) argument set to `t` (top).	N/A
			`\activeplot[t]{#1}{#2}{#3}{#4}`
`\activeplotbottom{}{}{}{}`	No	Equivalent to `\activeplot` with the first (optional) argument set to `b` (bottom).	Equivalent to `\activeplot`.
			`\activeplot[b]{#1}{#2}{#3}{#4}`
`\notinteractive`	No	Obsolete. Ignored.	Ignored.
`\interactive`	No	Obsolete. Ignored.	Ignored.
`\iex{}{}`	No	Obsolete. Ignored.	Ignored.
`\noshowiex`	No	Obsolete. Ignored.	Ignored.
`\showiex`	No	Obsolete. Ignored.	Ignored.
`\noIdocMenu`	No	Obsolete. Ignored.	Ignored.