

WOLFGANG KÜHN

iTe 1.1 User's Guide



the interactive T_EX editor

User's Guide for Version 1.1

Wolfgang Kühn

January 13, 1999

1 Description

iTeX is an easy to use, interactive authoring tool for L^AT_EX and T_EX documents. With **iTeX**, it is possible to select arbitrary display fragments such as figures, equations, labels, captions, etc. These fragments can then be freely moved, rotated and scaled. Existing tools are either awkward to use (for example cryptic L^AT_EX extensions), or they produce results of poor quality (font of captions does not match L^AT_EX font, for example).

Because **iTeX** is written in *Emacs-lisp*, it integrates nicely in the editing process. It is not a WYSIWYG front end for L^AT_EX, and **iTeX** does not change the way you use L^AT_EX to typeset 95% of your text. But it will help you to manage the remaining 5% fast with a high quality result.

To the user, **iTeX** is an easy to use and effective tool which is a lot of fun.

2 Copyright

All files in this distribution are copyrighted. You are allowed to freely copy and use these files for non-commercial use only. Do not redistribute a version which is incomplete or is changed in any way. The author makes no representations about the suitability of this software for any purpose. It is provided *as is* without expressed or implied warranty.

3 Installation

You must run ‘configure’ to install iTe. See Section 11 for customization.

4 Distribution

The **iTe** distribution is very small and presently contains the following files:

README	better read the manual
ite.el	lisp interface
ite_init.ps	PostScript interface
ite.tex	TeX interface
itedvi.tex	ditto
itegraphics.tex	ditto
manual.ps	User Manual in postscript format
example.tex	example file
demo	well, a demo

The only essential files to run **iTe** are ‘ite.el’ and ‘ite.tex’, which together claim less than 30kb. The newest version can always be down-loaded from the **iTe** home page at <http://www.zib.de/Visual/software/ite>. Also available from the **iTe** home page is an online version of the manual.

5 System Requirements

Because **iTe** is written in the three interpreted languages *Emacs-lisp*, *postscript* and \TeX , it will run on a large variety of platforms without the need to configure. The following tools are required.

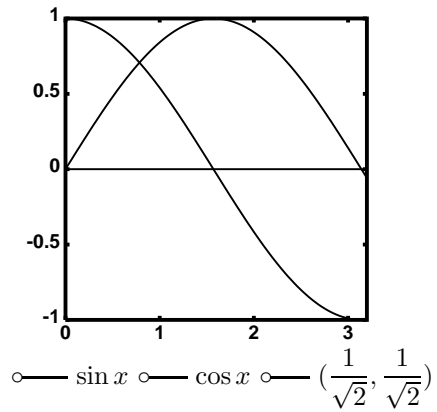
- *Emacs* or any variant of the famous editor;
- *dvips*, the dvi to ps translator by Radical Eye Software;
- *gs* (ghostscript), the postscript interpreter by Aladdin Enterprises.

iTe was successfully tested on the following platforms:

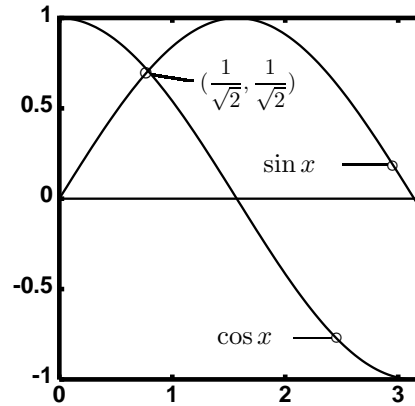
- SGI O2 with XEmacs(19.14), *dvips*(5.58f) and *gs*(4.03, 5.01, 5.10);
- SUN Sparc Ultra 1 with Emacs(19.34.1), *dvips*(5.58f) and *gs*(4.03);
- PC with Linux 2.0, XEmacs(20.4), *dvips*(5.58f) and *gs*(2.6.2).

6 Gallery of examples

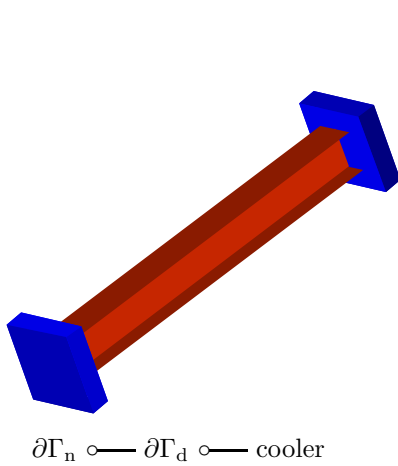
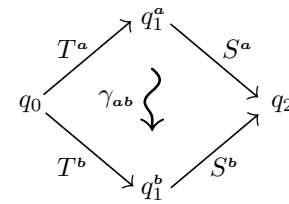
before iTe



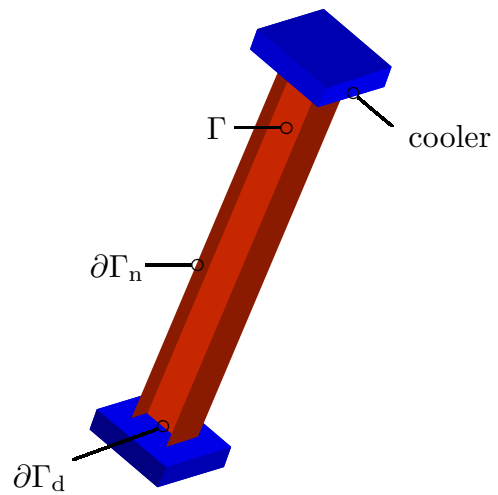
after iTe



$$\begin{matrix} \uparrow \\ \uparrow \\ \uparrow \\ \uparrow \end{matrix} \rightsquigarrow q_0 q_2 \quad q_1^b q_1^a \quad T^a T^b \quad S^a S^b \quad \gamma_{ab}$$



$$\circ - \Gamma - \circ$$



7 Invocation and Demo

The demo is started by typing

```
> demo
```

at the shell prompt. A recorded **iTe** session is then played back. No user input is required.

The following gives a quick-start to edit the example file using **iTe**. It is not the most efficient way to use **iTe**. See Section 11 on how to autoload and launch **iTe** from within *Emacs*.

1. Tex the example file 'example.tex' with the shell command

```
> latex example.tex
```

2. Start **iTe** with the shell command

```
> emacs -load ite.el
```

At the prompt, enter 'example.tex'.

3. Use the **iTe** mode key bindings to edit.

Note that the keys are only active if the *Emacs* window showing the file receives keyboard input. It is possible that the commands for launching *dvips* or *gs* were not found and an error is indicated. In this case also see Section 11.

8 iTe Mode Key Bindings

KEY	BINDING
Navigational keys	
space	select next page with objects
b	select previous page with objects
n	select next object
p	select previous object
Object manipulation keys first binding	
up-arrow	move selected object up
down-arrow	move selected object down
left-arrow	move selected object left
right-arrow	move selected object right
Object manipulation keys second binding	
up-arrow	magnify selected object
down-arrow	shrink selected object
left-arrow	rotate counterclockwise
right-arrow	rotate clockwise
Toggle key	
t	toggle between first and second binding
Modifiers for object manipulation keys	
Ctrl	manipulate in smaller steps
Alt	manipulate in tiny steps
Zooming	
Z	no zoom, not centered
z	no zoom but center
Ctrl z	zoom in on current object
Alt z	zoom in even more
Miscellaneous	
r	redisplay page (refresh)
q	quit iTe mode
s	suspend iTe mode
h	toggle help

The suspend key 's' needs further explanation. Use it to temporarily suspend iTe. To return to iTe mode, type 'ESCxite-mode'. The suspend key can be used to make *minor* changes in the manuscript during an iTe session. These changes **will not be** in effect until after the manuscript is recompiled and iTe restarted. Be aware not to delete any iTe objects, as this will utterly confuse iTe.

The zoom key 'z' may also take a numerical prefix. This number divided by 10 is then the zoom factor. For example the key sequence 'ESCu30z' will zoom in on the current object by a factor of 3.

Sometimes the window manager is too attentive and catches key events before they get to Emacs. You have to edit the respective resource files to suppress this behavior. For example, if you use the *fvwm* window manager, then add (or edit) the following lines to your '.fvwmrc' file:

```
# press arrow + control anywhere, and scroll by 1 page
Key Left      R      C      Scroll -100 0
```

Key Right	R	C	Scroll +100 +0
Key Up	R	C	Scroll +0 -100
Key Down	R	C	Scroll +0 +100

press arrow + meta key, and scroll by 1/10 of a page

Key Left	R	M	Scroll -10 +0
Key Right	R	M	Scroll +10 +0
Key Up	R	M	Scroll +0 -10
Key Down	R	M	Scroll +0 +10

9 A complete Example

This is the file 'example.tex', which contains the complete $\text{\LaTeX} 2_{\epsilon}$ code for a working example. The resulting figure is shown to the right.

```

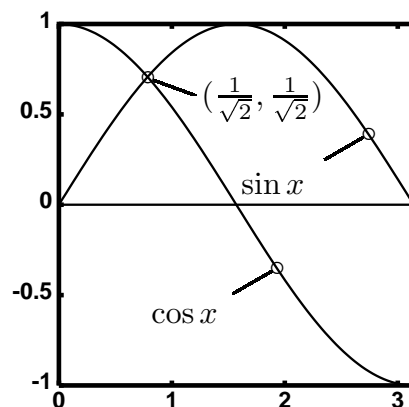
\documentclass{article}
\usepackage{graphics}
\input ite

\begin{document}

\begin{iteblock}(5cm,5cm)
\ite\includegraphics{sc.eps}
\ite$\grabber{5}$
\ite$\sin x$
\ite$\grabber{5}$
\ite$\cos x$
\ite$\grabber{5}$
\ite$(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$
\end{iteblock}

\end{document}

```



10 The \LaTeX Interface

The file 'ite.tex' has to be included at the beginning of the document with

```
\input ite
```

It will often be necessary to load the package *graphics* (or *graphicx*) with

```
\usepackage{graphics}
```

for $\text{\LaTeX} 2_{\epsilon}$ or

```
\documentstyle[graphics]{article}
```

for $\text{\LaTeX} 2.09$.

10.1 The *iteblock* Environment

```

\begin{iteblock}(x_dimen,y_dimen)
:

```

```
\ite<iTe object>
:
\end{iteblock}
```

The **iTe** block creates a box with the specified dimensions. While using **iTe**, this box is framed for convenience. The box will not appear in the printed version. The dimensions `x_dimen` and `y_dimen` are physical dimensions, each expressed either as a floating point number plus unit of measure or as a macro which expands to such a measure. Examples are `1in`, `2.54cm` or `\textwidth`.

You may specify as many `\ite` directives as you wish. The preferred way is to put each `\ite` statement at the beginning of a line. Each `\ite` is followed by an `<iTe object>`, which is any L^AT_EX code than can be placed inside an `\mbox`. This implies that it is possible for an `<iTe object>` to be a `\parbox` or a `\minipage`.

Nested **iTe** blocks are only allowed if the inner blocks are encapsulated by an *itebind* environment, see Section 10.3.

iTe will expand each `\ite` statement to `\ITE(x y a s)`, where x and y is the translation in units of `\ITEunit`, a is the angle of rotation, and s is the scaling factor of the object. You should not change these numbers manually.

10.2 The iTe unit

```
\ITEunit=<dimension>
```

Specify the unit which **iTe** uses. The default is 0.5pt or approximately 0.35mm. You should change the value of `\ITEunit` only if a higher resolution is required. With the default, the two 'X' characters in the example

```
\ITE(0 0 0 1)X
\ITE(200 0 0 1)X
```

are 100pt apart. With `"\ITEunit=1mm"`, the distance is 200mm.

10.3 The itebind environment

The *iteblock* environment is not allowed to appear in certain places in the T_EX file. These are

- inside another *iteblock*;
- in included files;
- in macros.

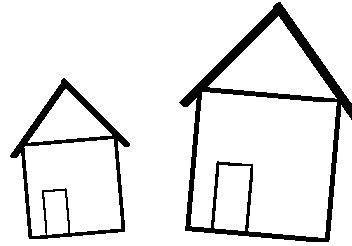
However, these restrictions do not apply if the *iteblock* environment is enclosed by an *itebind* environment. The effect of the *itebind* environment is that the *iteblock* environment is no longer recognized in an interactive **iTe** session. The use of the *itebind* environment is therefore always a two staged process. First an *iteblock* environment is interactively edited using **iTe**. Then it is enclosed (or frozen) by an *itebind* environment. The following example shows a typical use.


```

\newcommand{\house}{\begin{itebind}
\begin{iteblock}(0cm,0cm)
\ITE(0 0 0 1)\framebox(20,20){}
\ITE(7 0 0 0.482)\framebox(10,20){}
\ITE(19 66 -50 1)\rule{20pt}{1pt}
\ITE(-4 35 50 1)\rule{20pt}{1pt}
\end{iteblock}
\end{itebind}}

\begin{iteblock}(3cm,3cm)
\ITE(87 95 5 1.728)\house
\ITE(230 103 -5 2.589)\house
\end{iteblock}

```



10.4 Plain T_EX

iTe can also be used with plain T_EX. The example file 'example.tex' in equivalent plain T_EX format is

```

\input epsf
\input ite

\beginiteblock(\hsize,7cm)
\ite\epsffile{sc.eps}
\ite$\grabber{5}$
\ite$\sin x$
\ite$\grabber{5}$
\ite$\cos x$
\ite$\grabber{5}$
\ite$(1\over{\sqrt{2}},1\over{\sqrt{2}})$
\enditeblock

\bye

```

Likewise, the *itebind* environment has to be replaced by a matching `\beginitebind ... \enditebind` pair. A restriction for plain TeX is that it cannot use the *graphics* or *graphicx* package, see Section 13.

11 The *Emacs* Interface

There are two *Emacs* functions and two variables that are important to the user.

11.1 Functions

In *Emacs*, functions are called by typing 'ESCx' followed by the function name. Here 'ESC' stands for the 'ESCAPE' key. To call function `ite`, you have to type 'ESCxite'.

Function: `ite`

Call this function to start **iTe**. Only one **iTe** session is allowed at a time.

Function: `ite-mode`

Call this function to resume a suspended **iTe** session. See the warnings in Section 8.

11.2 Variables

In the header of `'ite.el'` are defined several variables useful to the user. Depending on system configuration, the following have to be set:

Variable: `its-dvips-command`

Change this variables so that it invokes *dvips* on your system. The default is `"dvips"`.

Variable: `ite-gs-command`

Change this variables so that it invokes *gs* on your system. The default is `"gs"`.

11.3 Autoloading iTe

If the following line

```
(autoload 'ite "~/ite/ite.el" "Interactive TeX Editor" t)
```

is added to the appropriate init file (for example to `'~/ .emacs'`), then there is no need to explicitly load **iTe** with the `'-load ite.el'` command line option whenever *Emacs* is launched. The sample line above assumes that the file `'ite.el'` resides in the directory `'/ite'` and has to be changed if necessary.

12 The L^AT_EX-iTe Cycle

While editing a T_EX file, you can cycle through as many **iTe** sessions as you wish.

1. In L^AT_EX-mode, edit and compile your text;
2. Start **iTe** with `'ESCxite'`;
3. Edit using the key bindings of Section 8;
4. Quit **iTe** with `'q'`;
5. Repeat.

You can also temporarily suspend **iTe** as described in Section 8.

13 Device Independence

The inclusion of the file `'ite.tex'` in the header does not produce device independent *dvi* code. There are two auxiliary files, both available from the **iTe** home page, which produce increasingly device independent code.

itegraphics.tex This will use the rotation and scaling operations of the *graphics* or *graphicx* package. One of these packages has to be included. The *graphics* and *graphicx* packages produce code which is more device independent than the code produced by `'ite.tex'`. Use `'itegraphics.tex'` when sending your L^AT_EX file to a publisher.

itedvi.tex Does not produce any device dependent code. No rotation and no scaling takes place. Use 'itegraphics.tex' when sending your \LaTeX file to a publisher who does not permit the inclusion of the *graphics* or *graphicsx* package.

It is important to understand that an interactive **iTe** session is only possible after the inclusion of 'ite.tex'.

14 Limitations

Presently, **iTe** does not allow the unrestricted (see Section 10.3) use of `\begin{iteblock}` ... `\end{iteblock}` groups

- in a nested fashion;
- in included files.

Both limitations will disappear in the next major release of **iTe**.

15 Feedback

Please report suggestions and bugs to me at kuehn@zib.de. Before you report a bug, please download the newest version of **iTe** and check if the bug persists. There is also a **bug list** at the **iTe** home page. If you do report a bug, please provide the following, if possible:

- Version of **iTe**:
- Version of *Emacs* or *XEmacs*:
- Version of *dvips*:
- Version of *gs*:
- Concise description of bug:
- Any idea how the bug may be fixed: