

DAS ANSITERM PAKET

VO.2.1 2026-08-26

Terminalfenster mit Aufzeichnung von Shellausführungen und ANSI-Interpretation

Matthias WERNER¹

<https://github.com/werner-matthias/osglecture>

Das Paket stellt zwei Umgebungen zur Verfügung, die Text in einer Box mit Terminal-Anmutung darstellen.

Dabei ermöglicht `ansitermexec` (bei aktiviertem `--shell-escape`) die Ausführung und Protokollierung von Shellbefehlen und deren Ergebnissen, wobei ANSI-Steuerzeichen korrekt umgesetzt werden. Wenn `\DocumentMetadata{tagging=on}` gesetzt ist, wird die Ausgabe getaggt.

Das Paket ist Teil des `osglecture`-Bundles, aber von den anderen Paketen weitgehend unabhängig.

Inhaltsverzeichnis

1 Nutzung	1 2 Implementation	4
1.1 Beispiele	2	
1.2 Optionen	3 Index	5

1 Nutzung

Achtung, Sicherheitsrisiko!

Die Umgebung `ansitermexec` benötigt zur Ausführung die Übersetzung mit `--shell-escape`. Dies stellt ein Sicherheitsrisiko dar, da damit Lua^AT_EX beliebigen Code auf ihrem System ausführen kann. Stellen Sie sicher, dass Sie nur unkritischen Code in den Body der Umgebung schreiben.

Laden Sie das Paket auf dem üblichen Weg mit `\usepackage{ansiterm}`. Es gibt keine Paketoptionen. Das Paket stellt zwei Umgebungen zur Verfügung und benötigt Lua^AT_EX.

1. matthias.werner@informatik.tu-chemnitz.de

`\begin{ansiterm}[\langle Optionen \rangle]`

Gibt den Inhalt innerhalb eines Terminalfensters aus.

`\begin{ansitermexec}[\langle Optionen \rangle]`

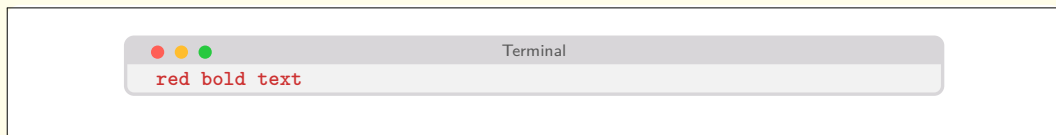
Gibt den Inhalt zeilenweise an eine zur Ausführung Shell weiter und setzt gibt die Ein- und Ausgabe in einem Terminalfenster aus.

1.1 Beispiele

```

1 \documentclass{article}
2 \usepackage{ansiterm}
3 \begin{document}
4   \begin{ansiterm}[escape-char=$]
5     ${1;31mred bold text${0m
6   \end{ansiterm}
7 \end{document}

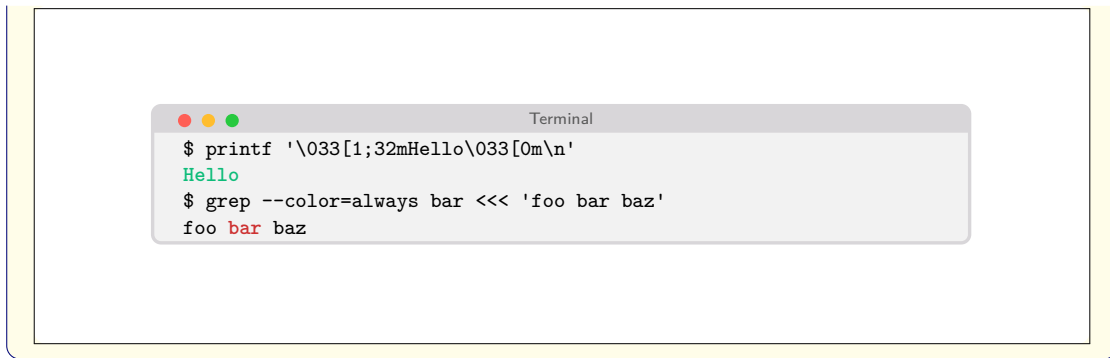
```



```

1 \documentclass{article}
2 \usepackage{ansiterm}
3 \begin{document}
4   \begin{ansitermexec}[
5     shell=/bin/sh,
6     cache=true,
7     sandbox=true,
8     show-input=true,
9   ]
10   printf '\033[1;32mHello\033[0m\n'
11   grep --color=always bar <<< 'foo bar baz'
12 \end{ansitermexec}
13 \end{document}

```



1.2 Optionen

`title = {<Überschrift des Terminals>}` Voreinstellung: `terminal`

`fontsize = {<Schriftgrößenbefehl>}` Voreinstellung: `\small`

`ansitermexec shell = {<Shell>}` Voreinstellung: `/bin/sh`
Shell zur Ausführung

`ansitermexec cwd = {<Verzeichnis>}` Voreinstellung: `.`
Arbeitsverzeichnis für die Ausführung

`ansitermexec cache = true|false|refresh` Voreinstellung: `true`
Legt fest, ob Ergebnis bei erneuter Übersetzung wiederverwendet wird. `refresh` erneuert den Cache.

`ansitermexec sandbox = true|false` Voreinstellung: `true`
Wenn der Schlüssel gesetzt (`true`) ist, wird der in einem neu angelegten Verzeichnis ausgeführt, das anschließend wieder gelöscht wird. Die Ergebnisse werden trotzdem ggf. im Cache behalten.

`ansitermexec show-input = true|false` Voreinstellung: `true`
Bei `false` wird nur die Ausgabe gezeigt.

`ansitermexec prompt = {<Prompt>}` Voreinstellung: `$~`

`escape-char = {<Zeichen für ESC>}` Voreinstellung: `$`

`lines = {<Zeilenbereich>}` Voreinstellung: `-`
Auswahl der Zeilen, die dargestellt werden.

`ellipsis = true|false` Voreinstellung: `true`
Zeigt Auslassungszeichen „:“ für übersprungene Zeilen

`ansitermexec fail-on-error = true|false` Voreinstellung: `true`
Erzeugt Fehlermeldung, wenn die Ausführung einen Fehlerstatus zurück gibt.

2 Implementation

Um die Implementationsbeschreibung zu erhalten, kommentieren Sie bitte `\OnlyDescription` in `ansiterm.dtx` aus.

Index

A

ansiterm (Umgebung) 2
ansitermexec (Umgebung) 1 ff.

C

cache 3
cwd 3

D

\DocumentMetadata 1

E

ellipsis 3
escape-char 3

F

fail-on-error 3
fontsize 3

L

lines 3

O

\OnlyDescription 4

P

prompt 3

S

sandbox 3
shell 3
show-input 3
\small 3

T

title 3

U

\usepackage 1

ansitermexec

ansitermexec