The atbegshi package

Heiko Oberdiek
<brheiko.oberdiek at googlemail.com>

2011/10/05 v1.16

Abstract

This package is a modern reimplementation of package everyshi without
the burden of compatibility. It makes use of \(\varepsilon\)-\TeX{}'s if available. Both \(\LaTeX\) and plain \TeX{} are supported.

Contents

1 Documentation 2
   1.1 Examples ................................................. 4
   1.1.1 Example: circle in background ...................... 4
   1.1.2 Example: adding TrimBox for dvipdfmx ............ 5

2 Method of \texttt{\textbackslash shipout} overloading 5
   2.1 \texttt{\textbackslash shipout} .......................... 5
   2.2 \texttt{\textbackslash afterassignment} ................ 6
   2.3 Test for direct or indirect boxes ..................... 6
      2.3.1 With \(\varepsilon\)-\TeX{} .......................... 7
      2.3.2 Without \(\varepsilon\)-\TeX{} ........................ 7
      2.3.3 \texttt{\textbackslash lastkern} method ................ 8
   2.4 Output ................................................. 8
   2.5 Separate box register ................................ 9
   2.6 Summary ................................................. 9
      2.6.1 With \(\varepsilon\)-\TeX{} .......................... 9
      2.6.2 Without \(\varepsilon\)-\TeX{}, traditional way ........ 10
      2.6.3 \texttt{\textbackslash lastkern} method ................ 10

3 Implementation 11
   3.1 Reload check and package identification ............. 11
   3.2 Catcodes .............................................. 12
   3.3 Preparations .......................................... 13
   3.4 Additions to the shipout box .......................... 17
   3.5 Positioning .......................................... 19
   3.6 Patches ................................................. 20
      3.6.1 Package \texttt{\textbackslash crop} ................. 20
      3.6.2 Package \texttt{\textbackslash everyshi} ............. 22
      3.6.3 Class \texttt{\textbackslash memoir} .................. 23

4 Test 25
   4.1 Catcode checks for loading ........................... 25

5 Installation 30
   5.1 Download .............................................. 30
   5.2 Bundle installation ................................... 30
   5.3 Package installation .................................. 30
   5.4 Refresh file name databases ........................... 30
   5.5 Some details for the interested ...................... 31
1 Documentation

Package \texttt{atbegshi} redefines \texttt{\shipout} to insert hooks for user code that is executed before the page is shipped out. The code may modify or even discard the output page. Three hooks are implemented:

1. A hook that is executed for every page, see \texttt{\AtBeginShipout}\par

2. A hook that is executed for the next page only, see \texttt{\AtBeginShipoutNext}\par

3. A hook that is only executed for the first page, see \texttt{\AtBeginShipoutFirst}\par

The hooks are executed in this order. The following three macros provide the user interface for adding code to these hooks:

\begin{verbatim}
\AtBeginShipout \{\langle code\rangle\}
\AtBeginShipoutBox
\end{verbatim}

Execute the \texttt{(code)} for every page. The page contents is held in box register \texttt{\AtBeginShipoutBox} and may be modified. Use \texttt{\AtBeginShipoutDiscard} if you want to discard the page.

\textit{Note:} Package \texttt{everyshi} uses box register 255. With package \texttt{atbegshi} you must use \texttt{\AtBeginShipoutBox} instead.

If \LaTeX{} calls \texttt{\shipout} in \texttt{\@outputpage} (part of its output routine), the meaning of \texttt{\protect} is \texttt{\noexpand}. \LaTeX{} sets \texttt{\protect} to the appropriate \texttt{\@typeset@protect} in the box that is shipped out. This is too late for the hooks, they are called earlier in the redefined \texttt{\shipout}. Therefore package \texttt{atbegshi} sets \texttt{\protect} to \texttt{\@typeset@protect} before it calls the hooks. (In \texttt{\EveryShipout} of package \texttt{everyshi} the user is responsible for the correct setting of \texttt{\protect}.)
This reimplements package everyshi’s `AtNextShipout`. The `<code>` is executed at shipout time of the next page only. It is just a convenience macro, it can be easily replaced by something like:

```latex
\newcommand{\MyShipoutHook}{}
\AtBeginShipout{\MyShipoutHook}
\gdef\MyShipoutHook{...
\gdef{\MyShipoutHook}{
```

(This can be necessary, if hook order does matter).

This reimplements LaTeX’s `AtBeginDvi`. This hook is usually used for `special` commands that include PostScript header files. The `code` is directly executed in a `vbox` that is put at the beginning of the output page. Dealing with the output box `AtBeginShipoutBox` is not necessary and not permitted here.

This macro notifies package atbegshi that the output page is discarded. The remaining hook code and the remaining hooks are not executed and the page is thrown away. Also `deadcycles` is cleared to zero like an ordinary `shipout` would do.

Usually the redefinition of `shipout` is delayed by `AtBeginDocument` (if this macro exists). This can be too late, if other packages also redefines `shipout` and the order does matter. `AtBeginShipoutInit` forces the immediate redefinition of `shipout`.

A quite common use case is the addition of `special` or other whatsis to the page output box. Macro `AtBeginShipoutAddToBox` puts `<stuff>` in a box with zeroed dimensions. The box with the `<stuff>` is put in the upper left corner of the shipout box `AtBeginShipoutBox`. Macro `AtBeginShipoutAddToBox` puts the `<stuff>` in the background, the other macro `AtBeginShipoutAddToBoxForeground` in the foreground after the original shipout box contents is set.

A void shipout box (that means a discarded page) remains void that means `<stuff>` is ignored in this case. The box type of `AtBeginShipoutBox` is preserved. Also the box nesting level for the original contents of `AtBeginShipoutBox` remains, for example, to avoid trouble with links across pages in case of pdfTeX.

This is a macro that puts material in the background of box `AtBeginShipoutBox`. The `<background material>` is set in an `hbox`, the reference point is the upper left corner of the output page. In case of pdfTeX in PDF mode, the settings of `pdfhorigin` and `pdfvorigin` are respected.
The macro \AtBeginShipoutUpperLeft is intended to be used in one of the hook setting macros, such as \AtBeginShipout, \AtBeginShipoutFirst, or \AtBeginShipoutNext.

For \LaTeX users the (background material) is set inside a picture environment:

\begin{picture}(0,0)
  \setlength{\unitlength}{1pt}\
  \put(0,0){\begin{picture}(0,0)
  \put(0,0){(background material)}
  \end{picture}}
\end{picture}

\AtBeginShipoutUpperLeftForeground {\begin{picture}(0,0)
  \setlength{\unitlength}{1pt}\
  \put(0,0){\begin{picture}(0,0)
  \put(0,0){(foreground material)}
  \end{picture}}
\end{picture}}

See \AtBeginShipoutUpperLeft. The difference is that the material is put in the foreground.

\AtBeginShipoutOriginalShipout {box}

It stores the meaning of \shipout at the time this package is loaded.

\AtBeginShipoutBoxWidth
\AtBeginShipoutBoxHeight
\AtBeginShipoutBoxDepth

These macros store the dimensions of the output box \AtBeginShipoutBox before the original shipout is called. If \shipout is not redefined before the package loading or the box dimensions are not changed by the redefined \shipout, these macros contain the dimensions of the shipout box. These values can be remembered by \label and \ref. For example, this is done by the package module zref-pagelayout of project zref. The dimensions of the shipout page can be used in some \TeX engines (pdf\TeX in PDF mode, \Xe\TeX) to calculate the media size of the shipout page if \pdfpagewidth and \pdfpageheight are not set.

### 1.1 Examples

#### 1.1.1 Example: circle in background

In this example we put a circle in the background in the middle of the paper.

1 (*example1*)
2 \documentclass[a4paper]{article}
3 \usepackage{color}
4 \usepackage{atbegshi}
5 \usepackage{picture}

Package picture makes life a little easier, because we can now also use length specifications in picture’s commands.

5 \usepackage{picture}

Now we draw the circle in the middle of the paper. \put moves downwards, because the origin is at the top of the page, not at its bottom.

6 \AtBeginShipout{\AtBeginShipoutUpperLeft{\AtBeginShipoutNext{%
7  \put(0.5\paperwidth,-0.5\paperheight){\circle{10}}%
8  }%}
9 }%
10 }
11 \begin{document}
12 \section{Hello World}
13 \newpage
14 \AtBeginShipoutNext{%
Only on this page we add a red cross.
\newpage
This page has the circle only.
\par\vspace{\fill}
The next page will be discarded.
\newpage
\AtBeginShipoutNext{%\AtBeginShipoutDiscard}
This page is discarded.
\newpage
The last page.
\end{document}

1.1.2 Example: adding TrimBox for dvipdfmx

Now an example from “real life” follows. Someone from the mailing list for dvipdfmx wants to put a TrimBox on every page. If we use \AtBeginShipout, we have to put the \special inside the box \AtBeginShipoutBox that gets shipped out.

\documentclass{minimal}
\usepackage{atbegshi}
\usepackage[dvipdfm, paperwidth=630bp, paperheight=810bp]{geometry}
\AtBeginShipout{%\setbox\AtBeginShipoutBox=\hbox{% \special{pdf: put @thispage <</TrimBox[9 9 621 801]>>}% \box\AtBeginShipoutBox}
}\begin{document}
First page
\newpage
Second page
\end{document}

Remember, in \AtBeginShipoutBoxFirst the \setbox wrapper code is implicitly given and the \special is used directly.

2 Method of shipout overloading

2.1 shipout

The \TeX\ primitive command \texttt{shipout} takes a box specification and puts the box as a new page in the output file. There are two kinds of box specifications:

Direct boxes: They are given by \texttt{hbox}, \texttt{vbox}, or \texttt{vtop},
e.g. \texttt{shipout|hbox(Hello World).}
Indirect boxes: \box or \copy references a box register by number. The box register contains the contents of the box.

Note: \box also clears the box register globally.

Then we have to differentiate between void and empty boxes:

Void: Initially or after \box there is no box in the box register. In this case the box register is not empty, but void.

Empty: A box with empty contents, such as \hbox{} (= \null) or \vbox{} is an empty \hbox or empty \vbox. If a box register holds such a box, the box still exists, therefore the box register is not void.

2.2 \afterassignment

We want to overload \shipout to do something with the box. It is quite impossible to do this reliably by catching the box using macro arguments. The variety of box specifications is too large, Examples:

\shipout\null
\shipout\vbox{...}
\shipout\vtop\bgroup...
\shipout\box255

Even worse, the braces don’t need to be balanced:

\shipout\hbox\bgroup}
\shipout\vbox\egroup

Happily TeX provides a reliable way via \afterassignment. It takes a macro name and executes it just after the assignment.

Now we can redefine \shipout. The box specification that follows \shipout is caught by \setbox. This is an assignment to a box register. \afterassignment notifies T\TeX, that we want to call \@test right after the assignment:

\shipout :=
\afterassignment\@test
\setbox\mybox=

We have seen different box specifications. Indirect boxes are easy to understand:

\shipout\box0 ⇒ \setbox\mybox=\box0 \@test

However direct boxes can have arbitrary contents with lots of other assignments. It would be quite unpredictable if T\TeX would put \@test after the first of such an assignment or after the box specification if the box lacks of assignments. Therefore T\TeX puts \@test right at the beginning of the box specification, e.g:

\shipout\hbox{Hello World}
⇒ \setbox\mybox=\hbox{\@test Hello World}

2.3 Test for direct or indirect boxes

Now we want to execute \@test, but where are we? We can be after the completed box assignment, if \shipout was called with an indirect box. Or we are right at the beginning of a direct box.
2.3.1 With \texttt{-\LaTeX}

With the \texttt{-\LaTeX}'s extensions the answer is very easy: Being inside the direct box means that we are inside a new group. The new primitive command \texttt{\currentgrouplevel} tells how deeply the groups are currently nested. Macro \texttt{@\test} just compares the previously stored group level with the current one:

\begin{verbatim}
\shipout :=
  \edef\saved@grouplevel{\number\currentgrouplevel}
  \afterassignment\@test
  \setbox\mybox=
\@test :=
  \ifnum\saved@grouplevel=\currentgrouplevel
    \% case: indirect box, the assignment is completed
    \@output
  \else
    \% case: direct box, we are inside the box
    \aftergroup\@outbox
  \fi
\end{verbatim}

2.3.2 Without \texttt{-\LaTeX}

Life becomes complicate without \texttt{-\LaTeX}. We cannot ask the group level. However, if we are inside a direct box, the box register \texttt{\mybox} is not yet changed by \texttt{\setbox}. Thus we need a special initial value and compare it in \texttt{@\test} with the current value of the box.

What can be used as initial value? Arbitrary box contents cannot be compared. \LaTeX{} only tells us a few properties:

- Box type: \texttt{\ifhbox}, \texttt{\ifvbox}
- Dimensions: \texttt{\wd}, \texttt{\ht}, \texttt{\dp}
- Voidness: \texttt{\ifvoid}

Unhappily all these qualities even combined are not sufficient for constructing an initial box value, because \texttt{\shipout} can be called with a box that is accidently just the same as the choosen initial value.

Nevertheless we have two alternatives for an initial value:

- A box of some type with some funny settings that are unlikely to occur in real life, e.g a height of \texttt{4911sp-\maxdimen}.
- A void box.

A collision between this initial value and an indirect \texttt{\shipout} box with just the same value is possible. Then \texttt{@\test} will make a wrong decision that it is executed inside a direct box and delays \texttt{\@output} by \texttt{\aftergroup}. Thus \texttt{\@output} is not called at the place we want. In contrary, the result is an uncertainty about the place:

- \texttt{\shipout} is used in a group that perhaps closes some pages later. A bad place for \texttt{\@output}.
- Without a surrounding group \texttt{\aftergroup} effectively kills its argument.

In the first case of a box with special dimensions we can even loose the page. However in the case of the void box, this effect is even desired, because the original \texttt{\shipout} does not output void boxes. All we have to do is to ensure that our box \texttt{\mybox} is always void except for the phase when the overloaded \texttt{\shipout} is executed. And secondly we must keep this semantics of \texttt{\shipout} for the void case in our macros, namely \texttt{\@output}. 

7
The nasty case is \shipout\box\voidb@x where the indirect box is void and that must not generate an output page. If a surrounding group is missing the output is ignored because of \aftergroup. Otherwise output is called some time later when the surrounding group closes. But \mybox is void outside the execution phase of the redefined \shipout. Also \@output checks for a void box and cancels the page output. The disadvantage remains that the hook in \@output is called for a page that will not be output.

2.3.3 \lastkern method

At the beginning of a new box, there is no \kern, the contents of the box is still empty and \lastkern returns 0 pt. This can be used to distinguish between direct and indirect boxes: We execute \setbox in a box with a preceding non-zero kern. After an indirect box, \lastkern sees this kern, otherwise it returns 0 pt.

We have two \setbox commands. The first creates a controlled context box where we can safely insert a \kern. We get rid of this temporarily used context box by putting the local \setbox in a group.

After the group we want to have our shipout box in \mybox. Therefore we use a global assignment here.

2.4 Output

With or without \-\TeX we ensure the original behaviour of \shipout that void boxes do not generate output pages.

Now we can place the hook \@hook for the user code that wants to manipulate the output box.
2.5 Separate box register

So far we have said nothing about the box number of \mybox. The following case that outputs the same page twice shows that we are not free in the use of the box register:

\shipout\copy<num> \shipout\box<num>

We manipulate the box by the hook and without \TeX the box must even be voided. However, the use case above requires that the box contents does not change at all. Therefore we must reserve a separate box register to avoid collisions with user box registers.

Note: Box register number 255 is special for the output routine, because \TeX complains if this box is not voided by the output routine. However, this requirement does not apply to \shipout at all. In fact \shipout does not change any box register. This is usually done by a call of \box, but the output routine can do it later after invoking of \shipout.

2.6 Summary

2.6.1 With \TeX

Putting the pieces together we get for \TeX:

\newbox\mybox
\let\original@shipout\shipout

\shipout :=
  \edef\saved@grouplevel\number\currentgrouplevel
  \afterassignment\@test
  \setbox\mybox=

\@test :=
  \ifnum\saved@grouplevel<\currentgrouplevel
    \expandafter\aftergroup
  \fi
  \@output

\@output :=
  \ifvoid\mybox
    % cancel output of void box
  \else
    \@hook
      \ifvoid\mybox
        % user code in \@hook could have voided the box
      \else
        \original@shipout\box\mybox
      \fi
    \fi
  \fi

9
2.6.2 Without -T\TeX, traditional way

And for T\TeX without \varepsilon-T\TeX:

\newbox\mybox
\begingroup
  \setbox\mybox=\box\mybox \% ensure \mybox is void
\endgroup
\let\original@shipout\shipout
\shipout :=
  \% trick to get a void box \mybox
  \begingroup
    \setbox\mybox=\box\mybox
  \endgroup
  \afterassignment\@test
  \setbox\mybox=
\@test :=
  \ifvoid\mybox
    \expandafter\aftergroup
  \fi
  \@output
\@output :=
  \ifvoid\mybox
    \% cancel output of void box
  \else
    \@hook
    \ifvoid\mybox
      \% user code in \@hook could have voided the box
    \else
      \original@shipout\box\mybox
    \fi
  \fi

2.6.3 \texttt{\%lastkern} method

And for T\TeX without \varepsilon-T\TeX using the \texttt{\%lastkern} method:

\newbox\mybox
\let\original@shipout\shipout
\shipout :=
  \begingroup
    \setbox\mybox=\hbox\bgroup
    \kern1pt
  \endgroup
  \afterassignment\@test
  \setbox\mybox=
\@test :=
  \ifdim\lastkern=0pt
  \expandafter\aftergroup
  \fi
  \@output
\@output :=
  \egroup
  \endgroup
  \ifvoid\mybox
    \% cancel output of void box
  \else
    \@hook
    \ifvoid\mybox
  \fi
% user code in \@hook could have voided the box
\else
  \original@shipout\box\mybox
\fi
\fi

3 Implementation

Package \atbegshi uses \eTeX's \currentgrouplevel, if it is available. Otherwise the \lastkern method is used.

56 (*package)

3.1 Reload check and package identification

Reload check, especially if the package is not used with \LaTeX.

57 \begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode39=12 % '
\catcode40=12 % (  
\catcode41=12 % ) 
\catcode44=12 % ,
\catcode45=12 % -
\catcode46=12 % .
\catcode47=12 % /
\catcode58=12 % : 
\catcode64=11 % @
\expandafter\let\expandafter\x\csname ver@atbegshi.sty\endcsname
\ifx\x\relax % plain-\TeX, first loading
\else
\ifx\empty\def\empty{}
\else
\expandafter\ifx\csname PackageInfo\endcsname\relax
\immediate\write-1{Package #1 Info: #2.}\
\else
\PackageInfo{#1}{#2, stopped}\
\fi
\fi
\x{atbegshi}{The package is already loaded}\
\aftergroup\endinput
\fi
\fi
\endgroup%

Package identification:

58 \begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode39=12 % '
\catcode40=12 % (  
\catcode41=12 % ) 
\catcode44=12 % ,
\catcode45=12 % -
\catcode46=12 % .
\catcode47=12 % /
\catcode58=12 % : 
\catcode64=11 % @

3.2 Catcodes

\begingroup\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode123=1 % { 
\catcode125=2 % }
\def\x#1#2#3[#4]{\endgroup
\immediate\write-1{Package: #3 #4}%
\xdef#1{#4}%
}%
\else
\def\x#1#2[#3]{\endgroup
#2[#3]%
\ifx#1\undefined
\xdef#1{#3}%
\fi
\ifx#1\relax
\xdef#1{#3}%
\fi
}
\fi
\expandafter\x\csname ver@atbegshi.sty\endcsname
\ProvidesPackage{atbegshi}%
[2011/10/05 v1.16 At begin shipout hook (HO)]%
3.3 Preparations

\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname RequirePackage\endcsname\relax
\def\TMP@RequirePackage#1[#2]{%
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname ver@#1.sty\endcsname\relax
\input #1.sty\relax
\fi
}%
\TMP@RequirePackage{infwarerr}[2007/09/09]%
\TMP@RequirePackage{ltxcmds}[2010/03/01]%
\else
\RequirePackage{infwarerr}[2007/09/09]%
\RequirePackage{ltxcmds}[2010/03/01]%
\fi
\AtBegShi@CheckDefinable
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname currentgrouplevel\endcsname\relax
\catcode`X=9 % ignore
\catcode`E=14 % comment
\ifAtBegShi@Discarded
\ltx@newif\ifAtBegShi@Discarded
\AtBeginShipoutDiscard
\AtBegShi@CheckDefinable\AtBeginShipoutDiscard
\def\AtBeginShipoutDiscard{%
\deadcycles=\ltx@zero
\global\AtBegShi@Discardedtrue
}%
\fi
\else
\def\AtBegShi@CheckDefinable#1{%
\@ifundefined{#1}{\ltx@one
\ltx@zero
\ltx@one
\else
\ltx@zero
\fi
\PackageError{atbegshi}{%string#1\space is already defined}%\@ehd
}\fi
\else
\def\AtBegShi@CheckDefinable#1{%
\@ifdefinable{#1}{}%
\fi
\fi
\ifAtBegShi@Discarded
\ltx@newif\ifAtBegShi@Discarded
\AtBeginShipoutDiscard
\AtBegShi@CheckDefinable\AtBeginShipoutDiscard
\def\AtBeginShipoutDiscard{%
\deadcycles=\ltx@zero
\global\AtBegShi@Discardedtrue
}%
\fi
\else
\def\AtBegShi@CheckDefinable#1{%
\@ifundefined{#1}{\ltx@one
\ltx@zero
\ltx@one
\else
\ltx@zero
\fi
\PackageError{atbegshi}{%string#1\space is already defined}%\@ehd
}\fi
\else
\def\AtBegShi@CheckDefinable#1{%
\@ifdefinable{#1}{}%
\fi
\fi

\AtBegShi@Shipout

\AtBegShi@Test

\AtBegShi@Output

\AtBegShi@GetBoxSize

\AtNextShipout
\edef\AtBeginShipoutBoxHeight{\the\ht#1}\
\edef\AtBeginShipoutBoxDepth{\the\dp#1}\
\AtBeginShipoutBoxWidth\
\def\AtBeginShipoutBoxWidth{0pt}\
\AtBeginShipoutBoxHeight\
\def\AtBeginShipoutBoxHeight{0pt}\
\AtBeginShipoutBoxDepth\
\def\AtBeginShipoutBoxDepth{0pt}\
\catcode`\X=11 %
\catcode`\E=11 %
\AtBegShi@First\
\def\AtBegShi@First{%
  \ifx\AtBegShi@HookFirst\ltx@empty
    \else\AtBeginShipoutAddToBox{\AtBegShi@HookFirst}%
    \fi
  \global\let\AtBegShi@First\ltx@empty
  \global\let\AtBeginShipoutFirst\AtBegShi@FirstDisabled
}\
\AtBegShi@Hook\
\gdef\AtBegShi@Hook{}\
\AtBegShi@HookNext\
\gdef\AtBegShi@HookNext{}\
\AtBegShi@HookFirst\
\gdef\AtBegShi@HookFirst{}\
\AtBeginShipout\
\AtBegShi@CheckDefinable\AtBeginShipout\
\def\AtBeginShipout{%
  \AtBegShi@AddHook\AtBegShi@Hook
}\
\AtBeginShipoutNext\
\AtBegShi@CheckDefinable\AtBeginShipoutNext\
\def\AtBeginShipoutNext{%
  \AtBegShi@AddHook\AtBegShi@HookNext
}\
\AtBeginShipoutFirst\
\AtBegShi@CheckDefinable\AtBeginShipoutFirst\
\def\AtBeginShipoutFirst{%
  \AtBegShi@AddTo\AtBegShi@HookFirst
}\
\AtBegShi@FirstDisabled\
\long\def\AtBegShi@FirstDisabled#1{%
  \PackageWarning{atbegshi}{First page is already shipped out, ignoring\MessageBreak
  \string\AtBeginShipoutFirst}\
}
\AtBeginShipAddTo
305 \begin{group}\expandafter\expandafter\expandafter\endgroup
306 \expandafter\ifx\csname g@addto@macro\endcsname\relax
307 \long\def\AtBeginShipAddTo#1#2{%
308 \begingroup
309 \toks\ltx@zero\expandafter{#1#2}\
310 \xdef#1{\the	oks\ltx@zero}\
311 \endgroup
312 }%
313 \else
314 \let\AtBeginShipAddTo\g@addto@macro
315 \fi
\AtBeginShipAddHook
316 \long\def\AtBeginShipAddHook#1#2{%
317 \AtBeginShipAddTo\AtBeginShipItem{#2}\
318 }
\AtBeginShipItem
319 \long\def\AtBeginShipItem#1{%
320 \ifAtBeginShipDiscarded
321 \else
322 #1%
323 \ifAtBeginShipDiscarded
324 \else
325 \ifvoid\AtBeginShipoutBox
326 \@PackageWarning{atbegshi}{Shipout box was voided by hook, ignoring shipout box}%
327 \AtBeginShipoutDiscard
328 \fi
329 \fi
330 \fi
331 \fi
332 \fi
333 \fi
334 }
\AtBeginShipoutInit
335 \AtBeginShipCheckDefinable\AtBeginShipoutInit
336 \def\AtBeginShipoutInit{%
337 \ltx@ifundefined{newbox}{%
338 \@PackageError{atbegshi}{\string\AtBeginShipoutInit\space failed because of missing \expandafter\string\csname newbox\endcsname\endgroup
339 }\@ehc
340 }%
341 \AtBeginShipCheckDefinable\AtBeginShipoutBox
342 \AtBeginShipCheckDefinable\AtBeginShipoutOriginalShipout
343 \global\let\AtBeginShipoutOriginalShipout\shipout
344 \global\let\shipout\AtBeginShipAddTo
345 \AtBeginShipoutInit
346 \%
347 \gdef\AtBeginShipoutInit{}%
348 }
349 }
\begin{group}\expandafter\expandafter\expandafter\endgroup
350 \expandafter\ifx\csname AtBeginDocument\endcsname\relax
351 \AtBeginShipoutInit
352 \else
353 \AtBeginDocument{\AtBeginShipoutInit}\%
354 \fi
3.4 Additions to the shipout box

\AtBeginShipoutAddToBox

\def\AtBeginShipoutAddToBox#1{%  
  \ifhbox\AtBeginShipoutBox  
    \edef\AtBegShi@restore{%  
      \hfuzz=\the\hfuzz\relax  
      \hbadness=\the\hbadness\relax  
    }%  
    \hfuzz=1073741823sp\relax  
    \hbadness=2147483647\relax  
    \setbox\AtBeginShipoutBox\hbox to \wd\AtBeginShipoutBox{%  
      \setbox\txt@zero\hbox{%  
        \begingroup  
        \AtBegShi@restore  
        #1%  
        \endgroup  
      }%  
      \wd\txt@zero=0pt\relax  
      \ht\txt@zero=0pt\relax  
      \dp\txt@zero=0pt\relax  
      \raise\ht\AtBeginShipoutBox\copy\txt@zero  
      \unhcopy\AtBeginShipoutBox  
    }%  
  \else  
    \ifvbox\AtBeginShipoutBox  
      \edef\AtBegShi@restore{%  
        \vfuzz=\the\vfuzz\relax  
        \vbadness=\the\vbadness\relax  
        \dimen\txt@zero=\the\dimen\txt@zero\relax  
      }%  
      \vfuzz=1073741823sp\relax  
      \vbadness=2147483647\relax  
      \dimen\txt@zero=\ht\AtBeginShipoutBox  
      \advance\dimen\txt@zero by \dp\AtBeginShipoutBox  
      \setbox\AtBeginShipoutBox\vbox to \dimen\txt@zero{%  
        \setbox\txt@zero\hbox{%  
          \begingroup  
          \AtBegShi@restore  
          \AtBegShi@restorebox%  
          \ht\AtBeginShipoutBox=\the\ht\AtBeginShipoutBox\relax  
          \dp\AtBeginShipoutBox=\the\dp\AtBeginShipoutBox\relax  
        }%  
        \vfuzz=1073741823sp\relax  
        \vbadness=2147483647\relax  
        \dimen\txt@zero=\ht\AtBeginShipoutBox  
        \advance\dimen\txt@zero by \dp\AtBeginShipoutBox  
        \setbox\AtBeginShipoutBox\vbox to \dimen\txt@zero{%  
          \setbox\txt@zero\hbox{%  
            \begingroup  
            \AtBegShi@restore  
            \AtBegShi@restorebox%  
            \ht\AtBeginShipoutBox=\the\ht\AtBeginShipoutBox\relax  
            \dp\AtBeginShipoutBox=\the\dp\AtBeginShipoutBox\relax  
          }%  
          \vfuzz=1073741823sp\relax  
          \vbadness=2147483647\relax  
          \dimen\txt@zero=\ht\AtBeginShipoutBox  
          \advance\dimen\txt@zero by \dp\AtBeginShipoutBox  
          \setbox\AtBeginShipoutBox\vbox to \dimen\txt@zero{%  
            \setbox\txt@zero\hbox{%  
              \begingroup  
              \AtBegShi@restore  
              \AtBegShi@restorebox%  
              \ht\AtBeginShipoutBox=\the\ht\AtBeginShipoutBox\relax  
              \dp\AtBeginShipoutBox=\the\dp\AtBeginShipoutBox\relax  
            }%  
            \vfuzz=1073741823sp\relax  
            \vbadness=2147483647\relax  
            \dimen\txt@zero=\ht\AtBeginShipoutBox  
            \advance\dimen\txt@zero by \dp\AtBeginShipoutBox  
          }%  
        }%  
      }%  
    \fi  
  }%  
\fi
3.5 Positioning

\begin{group}\expandafter\expandafter\expandafter\endgroup
\expandafter\if\csname RequirePackage\endcsname\relax
\begingroup\expandafter\expandafter\expandafter\endgroup
\if\csname ver@\endcsname\relax
\input \endcsname\relax
\fi
\def\TMP@RequirePackage#1[#2]{%
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\if\csname ver@#1.sty\endcsname\relax
\input #1.sty\relax
\fi
\TMP@RequirePackage{ifpdf}[2011/01/30]%
\else
\RequirePackage{ifpdf}[2011/01/30]%
\fi
\ifpdf
\def\AtBegShi@horigin{\pdfhorigin}%
\def\AtBegShi@vorigin{\pdfvorigin}%
\else
\def\AtBegShi@horigin{72.27pt}%
\def\AtBegShi@vorigin{72.27pt}%
\fi
\begingroup
\ifcase
\expandafter\if\csname picture\endcsname\relax
1%
\else
\expandafter\if\csname endpicture\endcsname\relax
1%
\else
0%
\fi
\fi
\endgroup
\def\AtBegShi@BeginPicture{%
\begingroup
\picture(0,0)\relax
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\if\csname unitlength\endcsname\relax
\else
\unitlength=1pt\relax
\ignorespaces
\}%
\def\AtBegShi@EndPicture{%
\endpicture
\endgroup
\ignorespaces
\}%
\def\AtBegShi@BeginPicture{%
\setbox\ltx@zero=\hbox{\begingroup
\ignorespaces
\}%
\def\AtBegShi@EndPicture{%
\endgroup
\ht\ltx@zero=0pt\relax
\dp\ltx@zero=0pt\relax
\copy\ltx@zero
\endgroup
\ifcase
\expandafter\if\csname picture\endcsname\relax
1%
\else
\expandafter\if\csname endpicture\endcsname\relax
1%
\else
0%
\fi
\fi
\else
\endgroup
\def\AtBegShi@BeginPicture{%
\begingroup
\hbox{\begingroup
\ignorespaces
\}%
\def\AtBegShi@EndPicture{%
\endgroup
\ht\ltx@zero=Opt\relax
\dp\ltx@zero=Opt\relax
\copy\ltx@zero
\endgroup
\}
3.6 Patches

Patches for \LaTeX\ packages that redefine \texttt{\shipout}. \LaTeX\ is now supposed to use \epsilon-\TeX. Thus we do not patch, without \LaTeX\ and \epsilon-\TeX.

3.6.1 Package crop

Fix of method and box.

\AtBeginShipoutUpperLeft A surrounding \texttt{\rlap} is not necessary, because the stuff is put in an \texttt{\hbox} with zero width.

\AtBeginShipoutUpperLeftForeground
3.6.2 Package `everyshi`

Fix of method. Use of box 255 is not changed.

```latex
\def\AtBegShi@PatchEveryshi{%
  \begingroup
  \long\def\AtBegShi@Everyshi@shipout{%\afterassignment\@EveryShipout@Test
  \global\setbox\@cclv= %
  }
  \long\def\AtBegShi@Everyshi@Test{%\ifvoid\@cclv\relax
  \aftergroup\@EveryShipout@Output
  \else
  \@EveryShipout@Output
  \fi
  \fi
}\if\AtBegShi@Everyshi@Test\relax
  \let\AtBegShi@found\relax
  \if\shipout\AtBegShi@Everyshi@shipout
    \def\AtBegShi@found{\shipout}
  \else
    \if\AtBeginShipoutOriginalShipout\AtBegShi@Everyshi@shipout
      \def\AtBegShi@found{\AtBeginShipoutOriginalShipout}
    \else
      \if\CROP@shipout\AtBegShi@Everyshi@shipout
        \def\AtBegShi@found{\CROP@shipout}
      \else
        \if\GPTorg@shipout\AtBegShi@Everyshi@shipout
          \def\AtBegShi@found{\GPTorg@shipout}
        \else
          \if\THBorg@shipout\AtBegShi@Everyshi@shipout
            \def\AtBegShi@found{\THBorg@shipout}
          \else
            \if\mem@oldshipout\AtBegShi@Everyshi@shipout
              \def\AtBegShi@found{\mem@oldshipout}
            \else
              \if\expandafter\ifx\csname @EveryShipout@Org@Shipout\endcsname
                \relax
                \if\EveryShipout@Shipout\AtBegShi@Everyshi@shipout
                  \def\AtBegShi@found{\EveryShipout@Shipout}
                \fi
                \fi
              \fi
            \fi
          \fi
        \fi
      \fi
    \fi
  \fi
\fi
\expandafter\endgroup
\expandafter\def\AtBegShi@found{%
  \edef\AtBegShi@GroupLevel{\number\currentgrouplevel}
  \afterassignment\@EveryShipout@Test
  \setbox\AtBeginShipoutBox= %
}%
```

22
\def\@EveryShipout@Test{% 
  \ifnum\AtBegShi@GroupLevel=\currentgrouplevel
  \else
  \expandafter\aftergroup
  \fi
  \AtBegShi@Everyshi@Output
}\}%
\def\AtBegShi@Everyshi@Output{% 
  \ifvoid\AtBeginShipoutBox
    \else
      \global\setbox\ltx@cclv\box\AtBeginShipoutBox
      \expandafter\@EveryShipout@Output
    \fi
  \%
  \@PackageInfoNoLine{atbegshi}{Package `everyshi' patched}\%
  \begingroup
  \fi
  \fi
  \endgroup
  \let\AtBegShi@PatchEveryshi\relax
%}
\@ifpackageloaded{everyshi}{
  \AtBegShi@PatchEveryshi
}{
  \AtBeginDocument{\AtBegShi@PatchEveryshi}%
}

\subsection{Class memoir}

Fix of method and box.

\def\AtBegShi@PatchMemoir{% 
\begingroup
  \def\AtBegShi@Memoir@shipout{% 
    \afterassignment\mem@shipi
    \setbox\@cclv=%
  }%
  \def\AtBegShi@Memoir@shipi{% 
    \ifvoid\@cclv
      \expandafter\aftergroup
    \fi
    \mem@shipii
  }%
  \def\AtBegShi@Memoir@shipiiA{% 
    \mem@oldshipout\vbox{% 
      \trimmarks
      \unvbox\@cclv
    }%
  }%
  \def\AtBegShi@Memoir@shipiiB{% 
    \ifvoid\@cclv
      \mem@oldshipout\box\@cclv
    \else
      \mem@oldshipout\vbox{% 
        \trimmarks
        \unvbox\@cclv
      }%
    \fi
  }%
  \def\AtBegShi@Memoir@PatchAB{% 
    \ifvoid\AtBeginShipoutBox
      \else
        \setbox\AtBeginShipoutBox=\vbox{%
    }%
\trimmarks
\Ifvbox\AtBeginShipoutBox
\unvbox\AtBeginShipoutBox
\else
\box\AtBeginShipoutBox
\fi
\AtBegShi@GetBoxSize\AtBeginShipoutBox
\expandafter\mem@oldshipout
\expandafter\box\expandafter\AtBeginShipoutBox
\fi
\def\AtBegShi@Memoir@shipiiC{% 2008/08/07 v1.6180339a
\ifvoid@clv
\mem@oldshipout\box@clv
\else
\ifshowtrims
\mem@oldshipout\vbox{\trimmarks\unvbox@clv}%
\else
\mem@oldshipout\box@clv
\fi
\fi
\def\AtBegShi@Memoir@shipiiD{% 2011/03/06 v3.6j
\ifvoid@clv
\mem@oldshipout\box@clv
\else
\ifshowtrims
\mem@oldshipout\vbox{%
\trimmarks
\nointerlineskip
\box@clv
}%
\else
\mem@oldshipout\box@clv
\fi
\fi
\def\AtBegShi@Memoir@PatchCD{%
\ifvoid@AtBeginShipoutBox
\else
\ifshowtrims
\setbox@AtBeginShipoutBox=\vbox{%
\trimmarks
\nointerlineskip
\box@AtBeginShipoutBox
}%
\fi
\AtBegShi@GetBoxSize\AtBeginShipoutBox
\expandafter\mem@oldshipout
\expandafter\box\expandafter\AtBeginShipoutBox
\fi
\Ifx\AtBegShi@Memoir@shipi\mem@shipi
\let\AtBegShi@found@ltx@one
\Ifx\AtBegShi@Memoir@shipii\mem@shipii
\let\AtBegShi@found@ltx@zero
\global\let\AtBegShi@Memoir@PatchX\AtBegShi@Memoir@PatchAB
\else\Ifx\AtBegShi@Memoir@shipiiB\mem@shipii
\let\AtBegShi@found@ltx@zero
\fi
4 Test

4.1 Catcode checks for loading

\catcode`\{=1 %

\catcode`\}=2 %
\catcode`#=6 %
\catcode`@=11 %
\expandafter\ifx\csname count@\endcsname\relax
\countdef\count@=255 %
\fi
\expandafter\ifx\csname @gobble\endcsname\relax
\long\def\@gobble#1{}%
\fi
\expandafter\ifx\csname @firstofone\endcsname\relax
\long\def\@firstofone#1{#1}%
\fi
\expandafter\if\csname loop\endcsname\relax
\else
\expandafter\@gobble\fi
\else%
\def\loop#1\repeat{%
\def\body{#1}%
\iterate
\}%
\def\iterate{%
\body
\let\next\iterate
\else
\let\next\relax
\fi
\next
\}%
\let\repeat=\fi
\}%
\def\RestoreCatcodes{}% 
\count@=0 %
\loop
\edef\RestoreCatcodes{\RestoreCatcodes
\catcode\the\count@=\the\catcode\count@\relax}
\ifnum\count@<255 %
\advance\count@ 1 %
\repeat
\def\RangeCatcodeInvalid#1#2{%
\count@=#1\relax
\loop
\catcode\count@=15 %
\ifnum\count@<#2\relax
\advance\count@ 1 %
\repeat
\def\RangeCatcodeCheck#1#2#3{%
\count@=#1\relax
\loop
\ifnum#3=\catcode\count@% 
\else
\errmessage{Character \the\count@ with wrong catcode \the\catcode\count@ instead of \number#3}
\fi
\fi
\ifnum\count0<#2\relax
\advance\count0 1\% 
\repeat 
\def\space{ }
\expandafter\ifx\csname LoadCommand\endcsname\relax
\def\LoadCommand{\input atbegshi.sty\relax}\
\fi 
\def\Test{\RangeCatcodeInvalid{0}{47}\
\RangeCatcodeInvalid{58}{64}\
\RangeCatcodeInvalid{91}{96}\
\RangeCatcodeInvalid{123}{255}\
\catcode`\@=12\% 
\catcode`\\=0\% 
\catcode`\%=14\% 
\LoadCommand 
\RangeCatcodeCheck{0}{36}{15}\
\RangeCatcodeCheck{37}{37}{14}\
\RangeCatcodeCheck{38}{47}{15}\
\RangeCatcodeCheck{48}{57}{12}\
\RangeCatcodeCheck{58}{63}{15}\
\RangeCatcodeCheck{64}{64}{12}\
\RangeCatcodeCheck{65}{90}{11}\
\RangeCatcodeCheck{91}{91}{15}\
\RangeCatcodeCheck{92}{92}{0}\
\RangeCatcodeCheck{93}{96}{15}\
\RangeCatcodeCheck{97}{122}{11}\
\RangeCatcodeCheck{123}{255}{15}\
\RestoreCatcodes
} 
\Test 
\csname @@end\endcsname
\end  
⟨/𭗍𭖾𭗌𭗍𭟣⟩
⟨*𭗍𭖾𭗌𭗍𭟤⟩
\input atbegshi.sty\relax
\def\msg#1\#2{\msg{}\msg{*** Test with box (#1), expected page output [#2]}% hash-ok} 
\newbox\voidbox 
\def\void{\box\voidbox} 
\begingroup 
\setbox\voidbox=\void 
\endgroup 
\count0=0\relax 
\AtBeginShipout{% 
\global\advance\count0 by 1\relax 
\msg{** Inside \string\AtBeginShipout: [\the\count0]}\% } 
\AtBeginShipoutFirst{% 
\msg{** Inside \string\AtBeginShipoutFirst}% 
Hello World% 
} 
\testmsg{\string\null}{1}
\shipout\null
\AtBeginShipoutFirst{%
  This is too late%
%
}%
\testmsg{void}{}
\shipout{void}
\testmsg{\string\copy255 (not void)}{2}
\setbox255\hbox{\vrule height 10bp width 10bp}
\shipout{\copy255}%
\testmsg{\string\copy255 (again)}{3}
\shipout{\copy255}%
\testmsg{\string\box255}{4}
\shipout{\box255}%
\testmsg{\string\box255 (again)}{}
\shipout{\box255}%
\testmsg{\string\hbox}{5}
\shipout{\hbox{\vrule height 5bp width 20bp}}%
\testmsg{\string\vbox}{6}
\shipout{\vbox{\hrule height 20bp width 5bp}}%
\testmsg{\string\null, voided by hook}{}
\begingroup
  \setbox\AtBeginShipoutBox=\box\AtBeginShipoutBox
\endgroup
\AtBeginShipout{\VoidBox} %
\shipout\null
\def\VoidBox{}
\begingroup
  \testmsg{void}{}
  \shipout{void}
\endgroup
\testmsg{output routine}{9}
Hello World
\vfill
\eject
\testmsg{\string\null\space(discarded)}{}
\AtBeginShipout{}
\msg{* Inside \string\AtBeginShipout: DISCARD}%
\AtBeginShipoutDiscard
\shipout{null}
\end
\end{test2}
\end{test3}
\NeedsTeXFormat{LaTeX2e}
\ProvidesFile{atbegshi-test3.tex}[2011/10/05 v1.16 Test file for LaTeX]
\RequirePackage{color}
\pagecolor{yellow}
\documentclass[a5paper,showtrims]{memoir}
\usepackage{atbegshi}
\AtBeginShipout{\setbox\AtBeginShipoutBox=\vbox{\vbox to 0pt\kern-1.5in\hbox to 0pt\kern-1.5in\color{blue}\rule{1in}{1in}\hss\vss\hrule\hbox{\vrule\box\AtBeginShipoutBox\vrule}\hrule}}
\usepackage{eso-pic}
\makeatletter\@ifundefined{@EveryShipout@Init}{\typeout{Test skipped}\@@end}{\@EveryShipout@Init\let\@EveryShipout@Init\relax\makeatother\AddToShipoutPicture{\hspace{.52\paperwidth}\colorbox{cyan}{\rule{0mm}{\paperheight}\hspace{.48\paperwidth}}}}\begin{document}\shipout{null}\shipout{\box\csname voidb@x\endcsname}\section{Hello World}\end{document}
Newer versions of class memoir emulate package crop and prevents its loading. This is undone in next line for this test file.\expandafter\let\csname ver@crop.sty\endcsname\relax\usepackage[color=red,cross,a4,center]{crop}\begin{document}\shipout{null}\shipout{\box\csname voidb@x\endcsname}\section{Hello World}\end{document}
5 Installation

5.1 Download

Package. This package is available on CTAN¹:


Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/oberdiek.tds.zip

TDS refers to the standard “A Directory Structure for \TeX Files” (CTAN:tds/tds.pdf). Directories with texmf in their name are usually organized this way.

5.2 Bundle installation

Unpacking. Unpack the oberdiek.tds.zip in the TDS tree (also known as texmf tree) of your choice. Example (linux):

unzip oberdiek.tds.zip -d ~/?texmf

Script installation. Check the directory TDS:scripts/oberdiek/ for scripts that need further installation steps. Package attachfile2 comes with the Perl script pdfatfi.pl that should be installed in such a way that it can be called as pdfatfi. Example (linux):

chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/

5.3 Package installation

Unpacking. The .dtx file is a self-extracting docstrip archive. The files are extracted by running the .dtx through plain \TeX:

tex atbegshi.dtx

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as texmf tree):

atbegshi.sty → tex/generic/oberdiek/atbegshi.sty
atbegshi.pdf → doc/latex/oberdiek/atbegshi.pdf
atbegshi-example1.tex → doc/latex/oberdiek/atbegshi-example1.tex
atbegshi-example2.tex → doc/latex/oberdiek/atbegshi-example2.tex
test/atbegshi-test1.tex → doc/latex/oberdiek/test/atbegshi-test1.tex
test/atbegshi-test2.tex → doc/latex/oberdiek/test/atbegshi-test2.tex
test/atbegshi-test3.tex → doc/latex/oberdiek/test/atbegshi-test3.tex
atbegshi.dtx → source/latex/oberdiek/atbegshi.dtx

If you have a docstrip.cfg that configures and enables docstrip’s TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

5.4 Refresh file name databases

If your \TeX distribution (\TeXe, \MiKTeX, …) relies on file name databases, you must refresh these. For example, \TeXe users run texhash or mktexlsr.

¹ftp://ftp.ctan.org/tex-archive/
5.5 Some details for the interested

**Attached source.** The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```bash
pdftk atbegshi.pdf unpack_files output .
```

**Unpacking with \LaTeX.** The `.dtx` chooses its action depending on the format:

- **plain \TeX:** Run `docstrip` and extract the files.
- \LaTeX: Generate the documentation.

If you insist on using \LaTeX for `docstrip` (really, `docstrip` does not need \LaTeX), then inform the autodetect routine about your intention:

```latex
latex \let\install=y\input{atbegshi.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

**Generating the documentation.** You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```latex
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdf\LaTeX:

```bash
pdflatex atbegshi.dtx
makeindex -s gind.ist atbegshi.idx
pdflatex atbegshi.dtx
makeindex -s gind.ist atbegshi.idx
pdflatex atbegshi.dtx
```

6 Catalogue

The following XML file can be used as source for the \TeX Catalogue. The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `atbegshi.xml`.

```xml
(*catalogue)
<?xml version='1.0' encoding='us-ascii'?>
<!DOCTYPE entry SYSTEM 'catalogue.dtd'>
<entry datestamp='$Date$' modifier='$Author$' id='atbegshi'>
  <name>atbegshi</name>
  <caption>Execute stuff at \shipout time.</caption>
  <authorref id='auth:oberdiek'/>
  <copyright owner='Heiko Oberdiek' year='2007-2011'/>
  <license type='lppl1.3'/>
  <version number='1.16'/>
  <description>
    This package is a modern reimplementation of package
    \xref{everyshi}\texttt{everyshi}, providing various commands
    to be executed before a \texttt{\shipout} command. It makes use of
    \TeX\texttt{e}\-\TeX\texttt{x}\texttt{e}\-\TeX\texttt{x}2019\-\texttt{s} facilities if they are available. The package may
    be used either with \LaTeX or with plain \TeX.
    \xref{everyshi}
  </description>
  <documentation details='Package documentation'
    href='ctan:/macros/latex/contrib/oberdiek/atbegshi.pdf'/>
</entry>
```
7 History

[2007/04/17 v1.0]
• First version.

[2007/04/18 v1.1]
• New method based on \lastkern is used if \eg\TeX{} is missing.
• \AtBeginShipoutDiscard also resets \deadcycles.

[2007/04/19 v1.2]
• \AtBeginShipoutEarly removed for simplification reasons.
• Forgotten definition of \AtBegShi@Info added.
• Patches for packages crop and everyshi and class memoir added.

[2007/04/26 v1.3]
• Use of package infwarerr.
• Catcode section after generic header.

[2007/04/27 v1.4]
• Small optimizations.

[2007/06/06 v1.5]
• \AtBeginShipoutUpperLeft added.
• Example added.
• Fix in second test file for newer version of memoir.

[2007/09/09 v1.6]
• Catcode section rewritten.

[2008/07/18 v1.7]
• Documentation of \AtBeginShipoutUpperLeft fixed and extended.

[2008/07/19 v1.8]
• \AtBeginShipoutUpperLeftForeground added.

[2008/07/31 v1.9]
• Second example (TrimBox for dvipdfmx) added.
• No changes in package code.
[2009/12/02 v1.10]
- \AtBeginShipoutOriginalShipout added.
- Test file fixed.

[2010/03/01 v1.11]
- Compatibility with ini-\TeX{} except for \texttt{\newbox}.

[2010/03/25 v1.12]
- \texttt{\AtBeginShipoutNext} can now be used inside \texttt{\AtBeginShipoutNext}.

[2010/08/18 v1.13]
- Fixes for \texttt{\AtBegShi@CheckDefinable}.

[2010/12/02 v1.14]
- Remove the warning because of void box if the hook calls \texttt{.}

[2011/01/30 v1.15]
- Already loaded package files are not input in plain \TeX{}.

[2011/10/05 v1.16]
- \texttt{\AtBeginShipoutAddToBox}, \texttt{\AtBeginShipoutAddToBoxForeground} added.
- \texttt{\AtBeginShipoutBoxWidth}, \texttt{\AtBeginShipoutBoxHeight}, \texttt{\AtBeginShipoutBoxDepth} added.
- Updates for patches of class \texttt{memoir}.

8 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

| Symbols \hspace{1cm} | \# \hspace{1cm} | \% \hspace{1cm} | @ \hspace{1cm} | @@end \hspace{1cm} | \@EveryShipout@Init \hspace{1cm} | \@EveryShipout@Org@Shipout \hspace{1cm} | \@EveryShipout@Output \hspace{1cm} | \@EveryShipout@Shipout \hspace{1cm} | \@EveryShipout@Test \hspace{1cm} | \@PackageError \hspace{1cm} | \@PackageInfoNoLine \hspace{1cm} | \@PackageWarning \hspace{1cm} | \AddToShipoutPicture \hspace{1cm} | \advance \hspace{1cm} | \afterassignment \hspace{1cm} | \A |
|---------------------|-----------------|--------------|-----------|-------------------|------------------------|---------------------------|----------------------|--------------------------|-----------------------------|-------------------|---------------------------|-------------------|-------------------|-----------------|--------|
|                     | 880             | 956          | 881, 954  | 1099            | 1101, 1102          | 603, 604, 843, 844     | 668, 670, 714           | 690, 691                  | 663, 673, 700, 703     | 194, 338         | 248, 646, 717, 862 | 238, 300, 326     | 1104 | 392, 453, 919, 927, 942, 993 | 223, 574, 617, 663, 700, 732, 851 |