URLs (`urls.dtx`)

Charles Duan

August 16, 2024

Every reference type can include a URL. *Hereinafter* formats URLs, makes them clickable links, and inserts line breaks into them. In that last respect, *Hereinafter*'s procedure is superior to other LaTeX URL packages, which fail when a hyperlinked URL in a footnote breaks across a page, a common occurrence for law review articles.

1 Input Syntax

The url and opturl parameters add a URL to a reference. The URL should be entered as unformatted text, except that percent signs should be escaped with backslashes to avoid being interpreted as comment symbols.

URLs will allow line breaks after main punctuation, similar to the package `url.sty`.[1] If there is a long string of alphanumeric characters, then this package may not find a suitable breakpoint. Including `\␣` (that is, a backslashed space) in the URL will force a breakpoint to be inserted into the URL, with no effect on appearance.

The command `\url{`⟨*url*⟩`}` can be used to format a URL in text.

User command to display a URL.

```
\def\url#1{%
    \hi@parseurl{#1}\reserved@a\reserved@a
}%
```

Parses the url (#1) and puts it into control sequence (#2). The resulting URL should have appropriate breakpoints put in, and be totally unexpandable or protected.

```
\def\hi@parseurl#1#2{%
    \hi@parseurl@forlink{#1}%
    \def#2{\leavevmode\hi@parseurl@use}%
    \expandafter\addto@macro\expandafter#2\expandafter{\expandafter{%
        \the\hi@url@count
    }}%
    \hi@parseurl@{#1}{#2}%
    \if@endswithdot{#1}{\addto@macro#2{\protect\@hi@dottrue}}%
    \addto@macro#2{\hi@parseurl@enduse}%
}
\DeclareRobustCommand\hi@parseurl@use[1]{%
    \@ifundefined{hi@pdfurl@#1}{%
        \edef\reserved@a{%
            \noexpand\special{%
                pdf:object @HIURL#1 <<
                    /S /URI
                    /URI (\csname hi@linkurl@#1\endcsname)
                >>
```

[1] An exception is for the percent sign, where the line break will be inserted *before* the symbol, so that URL-encoded entities are kept intact.

```
            }%
        }\reserved@a
    }{}%
    \edef\hi@parseurl@insert##1{%
        \noexpand\special{pdf:bann <<
            /Type /Annot
            /Subtype /Link
            /Border [0 0 0]
            /A @HIURL#1
        >>}##1\noexpand\special{pdf:eann}%
    }%
}
\DeclareRobustCommand\hi@parseurl@insertbox[1]{%
    \hbox{\hi@parseurl@insert{#1}}%
}
\let\hi@parseurl@insert\@firstofone
\DeclareRobustCommand\hi@parseurl@enduse{}
\newcount\hi@url@count
\make@find@start{http://}
\make@find@start{https://}
\make@find@start{ftp://}
\make@find@start{mailto:}
```

Parses a URL for purposes of forming it into a link URL. This primarily strips the URL of backslash escape characters.

```
\def\hi@parseurl@forlink#1{%
    \let\reserved@a\@empty
    % Force the link to start with https:// unless it has a protocol already
    \find@try\find@start{%
        {http://}\@gobble{https://}\@gobble{ftp://}\@gobble{mailto:}\@gobble
    }{#1}{\def\reserved@a{https://}}%
    \hi@parseurl@forlink@{#1}%
    \global\advance\hi@url@count\@ne
    \global\expandafter\let\csname hi@linkurl@\the\hi@url@count\endcsname \reserved@a
}
\def\hi@parseurl@forlink@#1{%
    \find@word{#1}%
        \hi@parseurl@forlink@chars
        \hi@parseurl@forlink@groups
        \hi@parseurl@forlink@words
        \relax
}
\def\hi@parseurl@forlink@words#1#2{%
    \addto@macro\reserved@a{#1}%
    \hi@parseurl@forlink@{#2}%
}
\def\hi@parseurl@forlink@groups#1#2{%
    \hi@parseurl@forlink@{\{#1\}#2}%
}
\def\hi@parseurl@forlink@chars#1#2{%
    \@test \ifcat\relax\noexpand#1\fi{%
        % If the char is a macro
        \@ifundefined{hi@url@\string#1}{%
            \expandafter\addto@macro\expandafter\reserved@a\expandafter{%
                \string#1%
            }%
        }{%
            \@expand{\addto@macro\reserved@a}{%
                \expandafter\@gobble\string#1%
            }{ii}%
        }%
    }{%
        % All other non-word chars
        \ifblank{#1}{}{% Ignore bare spaces
            \addto@macro\reserved@a{#1}%
        }%
    }%
    \hi@parseurl@forlink@{#2}%
}
```

#1 is the remaining part of the URL to parse; #2 is the macro to put the formatted URL into.

```
\def\hi@parseurl@#1#2{%
    \find@word{#1}{%
        \hi@parseurl@chars{#2}%
    }{%
        \hi@parseurl@groups{#2}%
    }{%
        \hi@parseurl@words{#2}%
    }{}%
}
```

#1 is the macro for the formatted URL; #2 is the text to process; #3 is the rest of the URL.

```
\def\hi@parseurl@words#1#2#3{%
    \addto@macro#1{\hi@parseurl@insertbox{#2}}%
    \hi@parseurl@{#3}{#1}%
}
\def\hi@parseurl@groups#1#2#3{%
    \hi@parseurl@{\{#2\}#3}{#1}%
}
```

Parse a non-word character in a URL. #1 is the macro to which the formatted URL will be added; #2 is the character, and #3 is the rest of the URL to be processed.

```
\def\hi@parseurl@chars#1#2#3{%
    \@test \ifcat\relax\noexpand#2\fi{%
```

If the character (#2) is a command (e.g., `\%`), see if a definition is given for this command. If so, then execute the definition for the command (e.g., `\hi@url@\%`). Otherwise, treat the command like text in the URL and parse it along with the remainder of the URL.

```
        \@ifundefined{hi@url@\string#2}{%
            \expandafter\hi@parseurl@\expandafter{\string#2#3}{#1}%
        }{%
            \csname hi@url@\string#2\endcsname{#1}{#3}%
            \hi@parseurl@{#3}{#1}%
        }%
    }{%
```

If #2 is not a command, then see if a definition is given for this character. If so, execute that definition. Otherwise, just add the character to the URL. In all cases, continue with parsing the remainder of the string.

```
        \@ifundefined{hi@url@\@backslashchar\string#2}{%
            \addto@macro#1{\hi@parseurl@insertbox #2}%
        }{%
            \csname hi@url@\@backslashchar\string#2\endcsname{#1}{#3}%
        }%
        \hi@parseurl@{#3}{#1}%
    }%
}
```

We now define macros of the form `\hi@url@`⟨*char*⟩ for each character that will receive special processing. Each macro will take two arguments: #1 is the URL macro for adding content, and #2 is the trailing text of the URL.

The original lists of characters were borrowed from url.sty, by Donald Arsenau. At this point the code no longer resembles the original beyond standard TeX programming conventions, but I'd like to credit the original package.

These characters accept a breakpoint after them.

```
\def\do#1{%
    \@namedef{hi@url@\string #1}##1##2{%
        \addto@macro##1{\hi@parseurl@insertbox}%
        \@expand{\addto@macro##1}{\expandafter\@gobble\string #1}{ii}%
        \find@word{##2}\@gobbletwo\@gobbletwo{%
            \addto@macro##1{\hi@url@break}\@gobbletwo
        }{}%
    }%
}
\do\.\do\@\do\!\do\;\do\]\do\)\do\,\do\+\do\=\do\#\do\:
```

These characters accept a breakpoint *and* add a little extra space.

```
\def\do#1{%
    \@namedef{hi@url@\string #1}##1##2{%
        \addto@macro##1{\hi@url@spacenobreak\hi@parseurl@insertbox}%
        \@expand{\addto@macro##1}{\expandafter\@gobble\string #1}{ii}%
        \find@word{##2}{%
            \addto@macro##1{\hi@url@spacenobreak}\@gobbletwo
        }{%
            \addto@macro##1{\hi@url@spacenobreak}\@gobbletwo
        }{%
            \addto@macro##1{\hi@url@spacebreak}\@gobbletwo
        }{}%
    }%
}
\do\/\do\?\do\&\do\-
```

These characters accept no breakpoints, but are often backslashed; including them forces the removal of the backslash.

```
\def\do#1{%
    \@namedef{hi@url@\string #1}##1##2{%
        \addto@macro##1{\hi@parseurl@insertbox}%
        \@expand{\addto@macro##1}{\expandafter\@gobble\string #1}{ii}%
```

```
    }%
}
\do\^\do\$
```

These characters require special treatment when they are seen, and a breakpoint is inserted if the next character is a word character.

```
\def\do#1#2{%
    \@namedef{hi@url@\string #1}##1##2{%
        \addto@macro##1{\hi@parseurl@insertbox{#2}}%
        \find@word{##2}\@gobbletwo\@gobbletwo{%
            \addto@macro##1{\hi@url@break}\@gobbletwo
        }{}%
    }%
}
\do\<{$\protect\langle$}%
% There's some bug in doc.sty that requires this
\expandafter\do\csname|\endcsname{$\protect\mid$}%
\do\{{$\protect\lbrace$}%
\do\\{$\protect\backslash$}%
\do\ {}%
\do\_{\_}%
```

These characters require special treatment. The first ones are not legal breakpoints; the percent sign is but the breakpoint comes before the character. (Ideally the percent would be recognized as an entity escape, collecting the next two characters, but that would require modifying the URL parser.)

```
\def\do#1#2{%
    \@namedef{hi@url@\string #1}##1##2{%
        \addto@macro##1{\hi@parseurl@insertbox{#2}}%
    }%
}
\do\>{$\protect\rangle$}
\do\}{$\protect\rbrace$}
\do\~{\char`\~}%
\@namedef{hi@url@\string\%}#1#2{%
    \ifstrempty{#2}{}{\addto@macro#1{\hi@url@spacebreak}}%
    \addto@macro#1{\hi@parseurl@insertbox{\%}}%
}%
```

Macros for types of breakpoints and spaces.

```
\newskip\urlbreakskip \urlbreakskip=0pt plus .4pt
\DeclareRobustCommand\hi@url@break{%
    \penalty\exhyphenpenalty
}
\DeclareRobustCommand\hi@url@spacenobreak{%
    \unskip\unpenalty\nobreak
    \hskip\urlbreakskip
}
\DeclareRobustCommand\hi@url@spacebreak{%
    \penalty\exhyphenpenalty\hskip\urlbreakskip
}
```

Now actually apply commands to the character classes.

```
\@namedef{hi@url@\string\ }#1#2{%
    \ifstrempty{#2}{}{\addto@macro#1{\hi@url@break}}%
}
\@namedef{hi@url@\string\^^M}#1#2{%
    \ifstrempty{#2}{}{\addto@macro#1{\hi@url@break}}%
}
```

2 Alternate Presentations of URLs

By default, URLs are shown in text. This format, logical for print publications with no other easy access to hyperlinks, can look lengthy and cumbersome in legal memoranda and other contexts. Accordingly, several other options are provided.

In documents with tables of authorities, URLs can be presented in an alternate format: The URLs are included only in the table listing, and citations in the document include a marker "*available online*." A footnote is added after the first such omitted URL, informing the reader that omitted URLs are in the Table of Authorities. To select this format, use the package option toaurl.

This format has the advantage of producing cleaner-looking briefs, particularly given the large font-to-text-field ratios that most courts require of briefs (such that URLs could take up a large fraction of a page). The main disadvantage is that it actually increases word count, since URLs typically count only as one word.

The text to be shown in place of URLs can be redefined with the macro `\ToaUrlMark`. The text to be shown in the footnote after the first such replaced URL can be redefined with the macro `\ToaUrlText`.

All displays of URLs are filtered through `\hi@showurl{`⟨*text*⟩`}`. By default, it just displays its argument.

```
\def\hi@urlstyle@full{%
    \def\hi@showurl##1{##1}%
}
```

Provides for URL being shown only in the table of authorities. #1 is the pre-URL signal; #2 is the URL itself.

```
\def\hi@urlstyle@toa{%
    \def\hi@showurl##1{%
        \if@hi@in@toa
            ##1%
        \else
            \ToaUrlMark
            \hi@showurl@toatext
        \fi
    }
    \def\hi@showurl@toatext{%
        \hi@defer@content{\ToaUrlText}%
        \global\let\hi@showurl@toatext\@empty
    }
    \def\ToaUrlText{%
        Locations of authorities available online are shown in the Table of
        Authorities.
    }
    \def\ToaUrlMark{\emph{available online}}
}
```

Alternatively, in documents intended to be used only electronically, URLs can be replaced with a word that is hyperlinked. To select this format, use the package option linkurl.

The text to be used in place of the URL is defined in the macro `\LinkText`. No line break protection is performed on the word, so it should be an unbreakable word or the macro should be defined to include an unbreakable `\hbox`.

```
\def\hi@urlstyle@link{%
    \def\hi@parseurl##1##2{%
        \hi@parseurl@forlink{##1}%
        \edef##2{%
            \noexpand\leavevmode
            \noexpand\special{%
                pdf:object @HIURL\the\hi@url@count <<
                    /S /URI
                    /URI (\csname hi@linkurl@\the\hi@url@count\endcsname)
                >>
            }%
            \noexpand\special{pdf:bann <<
                /Type /Annot
                /Subtype /Link
                /Border [0 0 0]
                /A @HIURL\the\hi@url@count
            >>}\noexpand\LinkText\noexpand\special{pdf:eann}%
        }%
    }
    \def\LinkText{\emph{link}}
}
```