Parentheticals (`parens.dtx`)

Charles Duan

August 16, 2024

Parentheticals are information appended to the text of citation items, and are used to convey additional information about the reference being cited. The following example includes two classic examples of parentheticals, one for court and date information and the other for characterizing the holding of a case:

> *See* Plaintiff v. Defendant, 123 F.4th 567 (1st Cir. 2000) (holding that the sky is blue).

Legal citation uses parentheticals extensively, both in highly structured ways to include required bibliographic data about references and more freeform ways that allow authors an additional avenue to converse with readers.

This section describes the nature of parentheticals as understood by this package, how parentheticals are entered for citation items, and how parentheticals are formatted.

1 Data Model

1(a) Types At the time that a citation item is being formatted, the package begins building up a list of parentheticals. The parentheticals can originate from two places. First, they may be associated with the reference itself, providing bibliographic information or commentary to be generally associated with the reference (e.g., publication dates, judicial opinion types, or alternate sources for finding references). These parentheticals are typically displayed only with the full citation of the reference.

Parentheticals may be provided as part of the citation item input, as described in `parse.dtx`. These "citation-level" parentheticals are generally meant to explain the writer's use of the reference in a particular citation—characterizing the holding of a case to demonstrate its relevance to the writer's argument, quoting the relevant portion of a reference, or identifying sources that the reference itself quotes or cites.

"Parentheticals" as used in this package are not merely text set inside parentheses. They also include items such as URLs, prior and subsequent case history, and other material typically set off with a comma after the main text of a citation item. Thus, there are three parentheticals in the following example for the date, the website, and a characterization:

> John Doe, A Book Online (2018), http://www.johndoe.xyz/book (providing information).

As this example shows, it is best to treat comma-offset text as parentheticals just like parentheses-enclosed text, because the two are intermingled in presentation. URLs, for example, appear after the date parenthetical to a citation but before parenthetical commentary text.

1(b) Content and Ordering Every parenthetical is associated with two items: (1) the text of the parenthetical, and (2) a priority number. The priority number serves two purposes. First, it establishes an ordering of parentheticals. For example, the date parenthetical for a citation always should precede other parentheticals, so it has the lowest priority number. Second, the priority number determines the appearance of the parenthetical. Other than hereinafter parentheticals (which unusually are set inside square brackets), parentheticals with even priority numbers are set inside parentheses while those with odd numbers are set off with a comma. As a result, there is consistency among even citations with many parentheticals.

The priority number for any parenthetical is set automatically based on the context in which the package receives the parenthetical. Generally writers should not need to deal with them, but the mechanism may be useful to know for authors who wish to change the default ordering or appearances.

In this package, macros are defined as `\hi@parens@`⟨*⟩ and parenthetical priority numbers are `\hi@paren@`⟨*⟩.

For any citation, the list of parentheticals is stored in `\hi@parens`. The format of this list is that each parenthetical is three grouped items: the macro `\@paren`, the priority number, and the text. The command `\hi@parens@reset` resets that list, and is called at the beginning of the routine for drawing a citation item.

```
\def\hi@parens@reset{\let\hi@parens\@empty}
```

Add a parenthetical to the parenthetical list, in the appropriate order. #1 is the priority number macro, and #2 is the text. This iterates through the current list of parentheticals and adds the parenthetical in order.

```
\DeclareRobustCommand\hi@parens@add[2]{%
    \expandafter\hi@parens@add@iterate\expandafter{\hi@parens}{}{#1}{#2}%
}
```

#1 is the remaining list of parentheticals to consider; #2 is the list of parentheticals already tested; #3 is the parenthetical class number to be added; #4 the parenthetical text.

```
\def\hi@parens@add@iterate#1#2#3#4{%
    \ifstrempty{#1}{%
        \def\hi@parens{#2\@paren{#3}{#4}}%
    }{%
        \hi@parens@add@test#1\@stop{#2}{#3}{#4}%
    }%
}
```

#1-#3 are `\@paren{`⟨*priority*⟩`}{`⟨*text*⟩`}` to be tested; #4 is the remaining list of parentheticals; #5-#7 are #2-#4 of `\hi@parens@add@iterate`.

For a new parenthetical to be placed before the one currently being tested, the new parenthetical's priority number (#6) must be less than the priority of the test parenthetical (#2). This ensures that for parentheticals of the same priority class, the most recently added one is placed last.

```
\def\hi@parens@add@test#1#2#3#4\@stop#5#6#7{%
    \@test\ifnum#2>#6\fi{%
        % The currently tested parenthetical should follow this one.
        \def\hi@parens{#5\@paren{#6}{#7}\@paren{#2}{#3}#4}%
    }{%
        % The currently tested parenthetical should precede this one. Add it to
        % the list of already tested parentheticals and do the next iteration.
        \hi@parens@add@iterate{#4}{#5\@paren{#2}{#3}}{#6}{#7}%
    }%
}
```

The list of priority numbers for parentheticals now follows.

```
\chardef\hi@paren@parallel=1             % Parallel citation
\chardef\hi@paren@date=2                 % Date parenthetical
\chardef\hi@paren@origdate=4             % Original date parenthetical
\chardef\hi@paren@docno=10               % Document number for court docs
\chardef\hi@paren@country=20             % Country name parenthetical
\chardef\hi@paren@casedocdocket=20       % Docket number for case document
\chardef\hi@paren@hia=30                 % Hereinafter
\chardef\hi@paren@casealt=40             % Alternate case name
\chardef\hi@paren@bookreview=50          % Book review citation
\chardef\hi@paren@opinion=60             % Opinion disposition (en banc, etc.)
\chardef\hi@paren@preparedby=70          % "prepared by" (R. 21.7.3)
\chardef\hi@paren@codified=80            % "to be codified at" for Fed. Reg.
\chardef\hi@paren@forthcomingms=80       % Page number for forthcoming article
\chardef\hi@paren@manuscript=90          % "unpublished manuscript"
\chardef\hi@paren@doc=100                % Document-level parenthetical
\chardef\hi@paren@url=111                % URL parenthetical
\chardef\hi@paren@accessdate=112         % URL access date
\def\reserved@a{link}\ifx\hi@urlstyle\reserved@a
    \chardef\hi@paren@url=110
    \def\hi@paren@showone#1#2{%
        \ifnum#1=\hi@paren@url
            ~\chop@space{#2}%
        \else
            \space(\chop@space{#2})%
        \fi
    }%
\fi
\chardef\hi@paren@cite=120               % Citation-level parenthetical
\chardef\hi@paren@prior=131              % Prior history of case
\chardef\hi@paren@subseq=133             % Subsequent history of case
\chardef\hi@paren@comment=205            % Commentary
\chardef\hi@paren@reprint=207            % "Reprinted in" parenthetical
```

2 Input Syntax

For parentheticals entered as parameters to references, the syntaxes are explained in the main documentation. This section describes the syntax for parentheticals entered as components of citation items, expanding on `parse.dtx`. These are called “citation-level” parentheticals.

Generally parentheticals are entered in citation items as plain text. Citation commands may be used inside parentheticals to produce nested citations, as described in `iface.dtx`. In addition, several special syntaxes are provided to allow for different types of common parentheticals.

The following input will convert ⟨*cite*⟩ to `\clause{`⟨*cite*⟩`}` and remove the colon. This is a convenient shortcut for the common types of parentheticals that start with “quoting,” “cited by,” or “discussed in” followed by a citation.

```
(⟨verb⟩ing: ⟨cite⟩)
(⟨verb⟩ in: ⟨cite⟩)
(⟨verb⟩ by: ⟨cite⟩)
(⟨verb⟩ at: ⟨cite⟩)
(codified at: ⟨cite⟩)
(codified as amended at: ⟨cite⟩)
```

For example, the following are largely equivalent:

```
\sentence{ref at 3 (quoting: source at 15)}
\sentence{ref at 3 (quoting \clause{source at 15})}
```

In both cases, the citation to `ref` will include a parenthetical with the word "quoting" followed by a formatted citation to `source`.

The main difference between the colon forms and using `\clause` directly is that the package can set priorities of the parentheticals correctly. As currently defined, the codification parentheticals will come first and subsequent commentary (using `in`, `by`, and `at`) comes last, with ⟨*verb*⟩`ing`-type parentheticals and other citation-level parentheticals in the middle.

Given a parenthetical text, parse it to figure out how to format it, and add it to the parentheticals list.

```
\def\hi@paren@parse#1{%
    \find@try\find@in{%
        {ing: }{\hi@parens@parse@ing}%
        {codified at: }{\hi@parens@parse@codified{codified at}}%
        {codified as amended at: }{%
            \hi@parens@parse@codified{codified as amended at}%
        }%
        { in: }{\hi@parens@parse@cmt{ in}}%
        { by: }{\hi@parens@parse@cmt{ by}}%
        { at: }{\hi@parens@parse@cmt{ at}}%
        {footnote: }{\hi@parens@parse@fn}%
    }{#1}{\hi@parens@add\hi@paren@cite{#1}}%
}
\make@find@in{ing: }
\make@find@in{ in: }
\make@find@in{ by: }
\make@find@in{ at: }
\make@find@in{codified at: }
\make@find@in{codified as amended at: }
\make@find@in{footnote: }
```

The following macros actually add the parentheticals, with proper priorities set.

```
\def\hi@parens@parse@ing#1#2{%
    \hi@parens@add\hi@paren@cite{#1ing \clause{#2}}%
}
\def\hi@parens@parse@codified#1#2#3{%
    \hi@parens@add\hi@paren@codified{#2#1 \clause{#3}}%
}
\def\hi@parens@parse@cmt#1#2#3{%
    \hi@parens@add\hi@paren@comment{\hi@fn@sig{#2#1} \clause{#3}}%
}
\def\hi@parens@parse@fn#1#2{%
    \ifstrempty{#1}{%
        \hi@defer@content{#2}%
    }{%
        \hi@parens@add\hi@paren@cite{#1footnote: #2}%
    }%
}
```

2(a) Deferred Footnotes A parenthetical starting with `footnote:` creates a "deferred footnote" inside a citation item. As its name suggests, the footnote is deferred for placement until after the citation item has been formatted and emitted to the document. If the citation item appears in body text, then the deferred footnote is placed after the separating punctuation at the end of the citation item. If the citation item is already in a footnote, though, the deferred footnote text is appended to the end of the footnote. This mechanism is useful primarily for legal memoranda if there is some material that should be connected to a citation in the middle of a citation string, where that material would be too unwieldy to include in a body-text parenthetical.[1]

Stores any deferred footnote content received so far.

```
\let\hi@deferred@note@content\relax
```

Stores deferred footnote content that needs to be displayed at the end of the footnote.

```
\let\hi@deferred@note@endcontent\relax
```

Adds content for a deferred footnote.

```
\def\hi@defer@content#1{%
    \ifblank{#1}{}{%
        \ifx\hi@deferred@note@content\relax
            \gdef\hi@deferred@note@content{#1}%
        \else
            \gappto\hi@deferred@note@content{ #1}%
        \fi
    }%
}
```

Hook to be run after the punctuation following a citation item in the middle of a citation string. In body text, this will place a footnote after that punctuation.

```
\def\hi@deferred@note@enditem{%
    \ifx\hi@deferred@note@content\relax\else
        \if@hi@note\else
            \@expand{%
                \global\let\hi@deferred@note@content\relax
                \footnote
            }\hi@deferred@note@content i%
        \fi
    \fi
}
```

Hook to be run at the end of a citation string, following the punctuation. In body text, this places a footnote after the punctuation. In a footnote, the content is further reserved for the end of the footnote.

```
\def\hi@deferred@note@endstring{%
    \ifx\hi@deferred@note@content\relax\else
        \if@hi@note
            \ifx\hi@deferred@note@endcontent\relax
                \global\let\hi@deferred@note@endcontent\hi@deferred@note@content
            \else
                \xdef\hi@deferred@note@endcontent{%
                    \expandonce\hi@deferred@note@endcontent
                    \space
                    \expandonce\hi@deferred@note@content
                }%
            \fi
            \global\let\hi@deferred@note@content\relax
        \else
            \@expand{%
                \global\let\hi@deferred@note@content\relax
                \footnote
            }\hi@deferred@note@content i%
        \fi
    \fi
}
```

Hook to be run at the end of a footnote. If there is deferred content to be added to the end of the footnote, this does so.

```
\def\hi@deferred@note@endnote{%
    \ifx\hi@deferred@note@endcontent\relax\else
        \unskip\space\hi@deferred@note@endcontent
        \global\let\hi@deferred@note@endcontent\relax
    \fi
}
```

[1] The deferred footnote mechanism is also used to deliver a footnote about URLs being in the Table of Authorities when that option is requested, as described in `urls.dtx`.

3 Formatting

The main task in formatting parentheticals is distinguishing among the different types. As discussed above, there are three parenthetical forms:

- Parentheticals surrounded by parentheses. Emitting these is straightforward.
- Parentheticals offset by a comma. The primary additional consideration here is that punctuation at the end of the parenthetical will be the last punctuation in the citation, so if the parenthetical ends with a dot then it is necessary to remove duplicative periods at the ends of sentences and such.
- Hereinafter parentheticals. In addition to using square brackets, we record use of the hereinafter form as described in `short.dtx` and possibly suppress the parenthetical if the hereinafter form was never otherwise used.

Draw the parentheticals. It will always add an extra space to the front as necessary.

```
\def\hi@parens@show{%
    \begingroup
    \def\@paren##1##2{%
        \@test \ifodd##1\fi{%
```

Dot management here is tricky because parentheticals may end with `\clause{`⟨*cite*⟩`}`. The solution here is to place `\hi@clause@endflag` at the end of the parenthetical text, which `\clause` checks for to suppress trailing punctuation. If the clause does end with a dot, then `\clause` will leave `\@hi@dottrue` on, and an additional check for a trailing period is performed, so hopefully any dots at the end of the parenthetical will be caught.

```
            ,
            \@hi@dotfalse
            ##2\hi@clause@endflag
            \if@endswithdot{##2}{\@hi@dottrue}%
        }{%
            \ifnum##1=\hi@paren@hia
                \if@hi@in@toa\else      % Not in TOA
                    \hi@short@maybeshow\@this@case{%
                        \space[hereinafter ##2]%
                    }%
                \fi
            \else
                \hi@parens@showone{##1}{##2}%
            \fi
            \@hi@dotfalse
        }%
    }%
    \@no@dot@space % This is because of the abbreviations in dates and such.
    \hi@parens
    \hi@dot@after@group
    \endgroup
}
\def\hi@parens@showone#1#2{%
    \space(\chop@space{#2})%
}
```