Abbreviations (abbrevs.dtx)

Charles Duan

May 2, 2023

Legal citations use a variety of abbreviation schemes for different types of texts. Case party names, court documents, and journal titles are not only abbreviated using different word lists, but with different abbreviation algorithms. Knowledge of these will generally not be important to writers, but they are discussed briefly below.

\AbbreviateFor

`\AbbreviateFor{`⟨*scheme*⟩`}{`⟨*text*⟩`}` abbreviates a text based on a given scheme. The schemes available are described below.

```
\def\AbbreviateFor#1#2{%
    \@nameuse{hi@abbrev@#1}{#2}{\@iden}%
}
```

1 Standard Abbreviation Algorithm

The standard abbreviation algorithm is described in the package `abbrev.sty`, which is of general applicability. The rules applied there are as follows:

- There exist different classes of abbreviations, each of which contains a map associating full words with their abbreviated forms.
- To abbreviate a text against a class, each word in the text is considered in sequence to find the longest matches. This avoids ambiguity when there are overlapping potential matches (a classic example being for law review articles, where "Lawyer" abbreviated to "Law.," which then theoretically abbreviated to "L..").
- Duplicative spaces are removed, to deal with situations where a word abbreviates to nothing (i.e., it should not be included).
- Two forms of dot management are performed. First, a space between two single-letter initials, or between a single-letter initial and a number, is removed. Thus, "West Virginia Railroad" would become "W.V.R.R." with no spaces. Second, two dots are collapsed into one, dealing with situations where an abbreviated word is followed by a dot in the original text (e.g., in a URL).

2 Cases and Other Names

Case names must be abbreviated in two ways: for inline citations (with only a few words abbreviated), and for non-inline citations. In addition to the standard abbreviation rules, there is a set of words that are unabbreviatable and also disfavored as short names (primarily country names and identifiers of the state in criminal cases).

The case name abbreviation rule is also applied to other names, such as institutional authors and agencies.

The case abbreviation scheme is called `name`, and the inline case abbreviation scheme is `inlcase`.

```
\abb@new{cases}
\def\NotMainParty#1{\expandafter\let\csname nocite@#1\endcsname\NotMainParty}
\def\hi@abbrev#1#2{%
    \abb@add{cases}{#1}{#2}%
    \expandafter\let\csname hi@noabb@#1\endcsname\@empty
    \NotMainParty{#1}%
}
\input hi-places
%
% These tell us what not to put as the single-party citation.
%
\NotMainParty{People}
\NotMainParty{United States}
\NotMainParty{Commonwealth}
\NotMainParty{State}
%
\def\hi@abbrev#1#2{%
    \abb@add{cases}{#1}{#2}%
    \expandafter\let\csname hi@noabb@#1\endcsname\@empty
}
\input hi-names
\input hi-casenames
```

`\hi@abbrev@name` Abbreviates a case name or other name for regular citation.

```
\def\hi@abbrev@name#1#2{%
    \def\reserved@a{#1}%
    \@ifundefined{hi@noabb@\expandafter\strip@prefix\meaning\reserved@a}{%
        \abb@abbrev{cases}{#1}{\abb@initialdots}{.}{#2}%
    }{#2{#1}}%
}
```

`\hi@abbrev@inlcase` Abbreviates a case name for inline citation.

```
\def\hi@abbrev@inlcase#1#2{%
    \abb@abbrev{inlcases}{#1}{\abb@initialdots}{.}{#2}%
}
\abb@new{inlcases}
\abb@add{inlcases}{and}{\&}
\abb@add{inlcases}{&}{\&}
\abb@add{inlcases}{Association}{Ass'n}
\abb@add{inlcases}{Brothers}{Bros.}
\abb@add{inlcases}{Company}{Co.}
\abb@add{inlcases}{Corporation}{Corp.}
\abb@add{inlcases}{Incorporated}{Inc.}
\abb@add{inlcases}{Limited}{Ltd.}
\abb@add{inlcases}{Number}{No.}
```

3 Legislative Materials

Legislative materials use the standard abbreviation rule, with just a different table. However, “Senate,” “House,” and “House of Representatives” alone are not abbreviated. The scheme is called `leg`.

`\hi@abbrev@leg` Abbreviate legislative materials.

```
\def\hi@abbrev@leg#1#2{%
    \@ifundefined{hi@abbrev@leg@noabbrev@#1}{%
        \abb@abbrev{legis}{#1}{\abb@initialdots}{.}{#2}%
    }{#2{#1}}%
}
\abb@new{legis}
\def\hi@abbrev#1#2{%
    \abb@add{legis}{#1}{#2}%
}
\let\hi@abbrev@leg@noabbrev@Senate\@empty
\let\hi@abbrev@leg@noabbrev@House\@empty
\cslet{hi@abbrev@leg@noabbrev@House of Representatives}\@empty
\input hi-legis
```

4 Court Documents

Court documents use the standard abbreviation rule, with just a different table. The scheme is called `cdoc`.

`\hi@abbrev@cdoc` Abbreviate court documents.

```
\def\hi@abbrev@cdoc#1#2{%
    \abb@abbrev{cdoc}{#1}{\abb@initialdots}{.}{#2}%
}
\abb@new{cdoc}
\def\hi@abbrev#1#2{%
    \abb@add{cdoc}{#1}{#2}%
}
\input hi-courtdocs
```

5 Journals

Journal titles require the most complex scheme for abbreviation, because they do not follow the usual abbreviation rules in several ways. The scheme for these abbreviations is `journal`.

Initially, some reference types perform a check on journal names to see if they were entered already abbreviated (based on the presence of any dots in the name). If so, then a warning is issued and the journal name is not further abbreviated. (The parameter `noabbrevjrn` will suppress this.) If not, then abbreviation proceeds.

`\hi@abbrev@journal@check` Check to see if a name given in parameter `rep` is already abbreviated. If so, issue a warning unless `noabbrevjrn` is set. Otherwise, abbreviate the name.

```
\def\hi@abbrev@journal@check{%
    \hi@ifset\hi@kv@noabbrevjrn{}{%
        \expandafter\hi@abbrev@journal@check@\expandafter{\hi@kv@rep}%
    }%
}
\def\hi@abbrev@journal@check@#1{%
    \find@in{.}{#1}{%
        \hi@abbrev@journal@check@warn{#1}\@gobbletwo
    }{}%
    \hi@abbrev@journal{#1}{\def\hi@kv@rep}%
}
\make@find@in{.}
\def\hi@abbrev@journal@check@warn#1{%
    \PackageWarning\hi@pkgname{%
        Journal name `#1'\MessageBreak
        should be entered unabbreviated.\MessageBreak
        If this is okay, add the parameter\MessageBreak
        `noabbrevjrn'. This occurred%
    }%
}
```

Next, there is the special case of single-word journal titles or journal titles of "The ⟨*word*⟩." These titles are never abbreviated and the word "The" is retained

to prevent titles from becoming incomprehensible (*The Register* would become merely "*Reg.*").

\hi@abbrev@journal Journal name abbreviations.

```
\def\hi@abbrev@journal#1#2{%
    \find@start{The }{#1}{%
        \hi@abbrev@journal@word{The }%
    }{%
        \hi@abbrev@journal@word{}{#1}%
    }{#2}%
}
```

Finally, there is an exception for the deletion of spaces between single-letter initials: The space cannot be removed when the one of the letters is geographic or institutional, and the other letter is for subject matter. For example, "North Carolina Law Journal" would be abbreviated "N.C. L.J.," deleting all spaces except that between "C." and "L." To implement this, the tables effectively use two types of dots, normal periods for geographic/institutional words and a special dot macro for subject matter words, and the dot management algorithm is executed twice.

```
\def\hi@abbrev@journal@word#1#2#3{%
    \find@in{ }{#2}{%
        \@firstofthree{%
            \abb@abbrev{jab}{#2}{\abb@initialdots}{.}%
                    {\abb@initialdots}{\abb@dot}{#3}%
        }%
    }{#3{#1#2}}%
}
\make@find@start{The }
\make@find@in{ }
```

In terms of abbreviation of words, there are three sources: the table of geographic locations, the table of institution names, and the table of common words in journal titles. These tables implement the differentation between institutional and subject-matter word dots.

```
\let\hi@jrnabb@inferlvl\tw@
\abb@new{jab}
\def\hi@abbrev#1#2{%
    \abb@add{jab}{#1}{#2}
}
\input hi-names
\input hi-places
\input hi-jrnplaces
\input hi-jrnwords
\hi@abbrev{The}{}
\hi@abbrev{the}{}
\hi@abbrev{of}{}
\hi@abbrev{at}{}
\hi@abbrev{de}{}
\hi@abbrev{in}{}
\hi@abbrev{,}{}
```

6 Procedural Phrases

Procedural phrases use a unique set of word-based abbreviations, but there is also a defined list of standard procedural phrases. Additionally, there is a rule about use of commas at the end of phrases. Thus, the abbreviation algorithm must consider a few additional rules. If a phrase to be abbreviated is a predefined one, then the predefined abbreviated phrase is returned, with comma usage

established. If it is not, then the phrase is abbreviated, but a warning is issued because the correctness of any trailing comma cannot be established.

The abbreviation scheme is called `expl`.

\ExplanatoryPhrase The macro `\ExplanatoryPhrase{`⟨*phrase*⟩`}` defines a new explanatory phrase. The unabbreviated phrase should be given, including any trailing comma if needed.

```
\def\ExplanatoryPhrase#1{%
    \find@end{,}{#1}{\hi@explanatoryphrase{,}}{\hi@explanatoryphrase{}{#1}}%
}
\make@find@end{,}
% \#1 is any trailing comma; \#2 is the phrase with no comma.
\def\hi@explanatoryphrase#1#2{%
    \abb@abbrev{expl}{#2}{\hi@explanatoryphrase@{#1}{#2}}%
}
```

Defines acceptable input forms as both unabbreviated and abbreviated forms, with or without a comma. Thus, for the phrase “affirmed,” acceptable input could be `affirmed`, `affirmed,`, `aff'd`, or `aff'd,`. #1 is the comma, #2 the unabbreviated phrase, #3 the abbreviated form.

```
\def\hi@explanatoryphrase@#1#2#3{%
    \@namedef{hi@exp@#2}{#3#1}%
    \@namedef{hi@exp@#3}{#3#1}%
    \@namedef{hi@exp@#2,}{#3#1}%
    \@namedef{hi@exp@#3,}{#3#1}%
}
```

Now input the abbreviations table.

```
\abb@new{expl}
\def\hi@abbrev#1#2{\abb@add{expl}{#1}{#2}}
\let\hi@multi@abbrev\ExplanatoryPhrase
\input hi-explanatory
```

\hi@abbrev@expl The abbreviation scheme macro for `expl` is now defined. #1 is the phrase to be abbreviated, and #2 the callback.

```
\def\hi@abbrev@expl#1#2{%
    \@ifundefined{hi@exp@#1}{%
        \hi@abbrev@expl@warn{#1}%
        \abb@abbrev{expl}{#1}{#2}%
    }{%
        \@expand{#2}{\csname hi@exp@#1\endcsname}{ii}%
    }%
}
\def\hi@abbrev@expl@warn#1{%
    \PackageWarning\hi@pkgname{%
        In defining reference \@this@case, the explanatory\MessageBreak
        phrase `#1' was not previously\MessageBreak
        known. Use \string\ExplanatoryPhrase{#1} to\MessageBreak
        define it. Also, make sure your comma usage conforms\MessageBreak
        to T.9. This occurred%
    }%
}
```