

# Short Names (short.dtx)

Charles Duan

May 3, 2023

Many types of reference use a short name to identify the reference after the first citation.<sup>1</sup> Cases use a distinctive party name, and books and articles use the authors' last names, for example. Writers can also choose their own short names for references using the `short` and `hereinafter` parameters.

Two problems can arise with these short names. First, two references may use the same text for a short name. In this case, the short name is ambiguous, so one or the other reference should use a different short name. Second, a writer may define a short name for a reference, say using `hereinafter`, but cite that reference only once in the document. A `hereinafter` parenthetical added to that first full citation is thus extraneous and unnecessary, and ought to be removed.

This package deals with both of these problems, by issuing warnings about ambiguous short names and automatically deleting unnecessary short name parentheticals. No user intervention is required other than to respond to warnings by adding distinctive short names.<sup>2</sup>

It is worth briefly noting that this short name management feature exemplifies the benefit of software-automated legal citations. A human cite-checker wanting to solve these two short name problems must keep a table of citations, to be searched and cross-referenced for every short name used. In a law review article citing hundreds or thousands of references, this task is tedious and error-prone, especially since the tabulation must be redone if the document is revised. For a computer, though, this task takes a matter of seconds.

This package works by citation macros executing the following hook macros in appropriate places.

- `\hi@short@register` is called at the time of defining a reference with a short form, to register the short-form text with the system.
- `\hi@short@use` is run any time that a short-form citation is to be displayed.
- `\hi@short@maybeshow` may be used in full-citation macros to optionally show text only if the short form will be used later.

---

<sup>1</sup>Short *names* are to be distinguished from short *citations*, which are described in `state.dtx`. A short name is text used to identify a reference the second and subsequent times, and that name may be used as part of short citations, but most short (non-inline) citations include further information, such as a *supra* note number or locator information.

<sup>2</sup>A possible future improvement would be to generate such short names automatically. To do so, though, it would be necessary to build into the package knowledge of conventional ways of disambiguating short names: Cases traditionally have roman numerals appended to them, while authored works typically have a distinctive word from the title added.

There is an `\AtEndDocument` hook that updates the aux file with which short-form citations were used and omitted incorrectly, to fix upon the next compilation.

## 1 Interface

Register a short citation text. This creates the macro `\hi@short@cite@⟨ref⟩`, which contains instructions to be executed the first time the short citation is used. Two instructions may be added:

- An instruction to test whether the reference's short text is duplicative (see `\hi@short@testdup`)
- If this is a user-specified short name, then an instruction to mark that the short name has been used at least ones (see `\hi@short@userdel`)

For user-specified short names, this also adds to `\hi@short@user@list` above.

This can be called multiple times on a single citation reference; only the first will count.

`#1` is the reference name (which must be fully expanded); `#2` is the short cite text; `#3` is `y` or `n` depending on whether this is a user-specified short name.

```
\def\hi@short@register#1#2#3{%
  \ifundefined{hi@short@cite@#1}{%
```

Defines the instruction macro and adds the first instruction, for short-name duplicate checking.

```
\StripForAlpha{#2}{\hi@short@register@duplicate{#1}}%
\find@eq{y}{#3}{%
```

If a user-defined citation, add to the list and add an instruction to delete from the list once the short form has been used.

```
\global\addto@macro\hi@short@user@list{#1,%
\global\expandafter\addto@list\csname hi@short@cite@#1\endcsname{%
  \hi@short@userdel{#1}%
}%
}%
}{}%
}%
}
```

Runs `#2` only if the user-defined short form for `#1` is not to be suppressed. This should surround short name parentheticals in full citations.

```
\DeclareRobustCommand\hi@short@maybeshow[2]{%
  \ifundefined{hi@short@nouser@#1}{#2}{}%
}
```

This should be run whenever a short cite form is used. `#1` is the reference name.

```
\DeclareRobustCommand\hi@short@use[1]{%
  %
  % Run the short citation instructions
  %
  \csname hi@short@cite@#1\endcsname
  %
  % Ensure that the instructions are only ever run once
  %
  \global\expandafter\let\csname hi@short@cite@#1\endcsname\relax
  \ifundefined{hi@short@nouser@#1}{%
    \PackageWarning{hi}{pkgnam}%
    The user-defined short form for #1\MessageBreak
    was suppressed in the full citation but\MessageBreak
    was later used. Recompile (possibly twice)\MessageBreak
    to correct the full citation form.\MessageBreak
    This occurred%
  }%
}%
}
```

## 2 Duplicate Name Detection

Registers a short citation reference's alphanumeric-stripped text. `#1` is the reference name, `#2` the stripped text.

```
\def\hi@short@register@duplicate#1#2{%
  \global\@namedef{hi@short@cite@#1}{\hi@short@testdup{#1}{#2}}%
}
\make@find@eq{y}%
```

Tests for duplicativeness in short citation usage. This is executed as part of the short citation instructions when the first short citation is encountered. #1 is the reference name, and #2 the alphanumeric-stripped short citation text.

```
\def\hi@short@testdup#1#2{%
  \@ifundefined{hi@short@name@#2}{%
    \global\@namedef{hi@short@name@#2}{#1}%
  }{%
    \expand\hi@short@warnambig{%
      \csname hi@short@name@#2\endcsname
    }{i}{#1}{#2}%
  }%
}
```

Warning to be displayed upon duplicative short names.

```
\def\hi@short@warnambig#1#2#3{%
  \def\reserved@a{#3}%
  \PackageWarning\hi@pkgname{%
    A short citation for reference `#2' is\MessageBreak
    ambiguous with a previously used short citation\MessageBreak
    for `#1'. The citation looks like:\MessageBreak
    \expandafter\strip@prefix\meaning\reserved@a\MessageBreak
    Please use a hereinafter form.\MessageBreak
    This occurred%
  }%
}
```

### 3 User-Defined Short Names

This is a list of all user-defined short citations. It is a comma-separated list, ending with a comma for ease of processing. As short citation forms are used, items are then deleted from this list, so at the end of the document, all that remains are items with a user-defined short form that went unused. The list is constructed such that every item is surrounded by commas on both sides.

```
\def\hi@short@user@list{,}
```

Deletes an item from the user-defined short form list. This is executed as part of the short citation instructions when the first short citation is encountered.

```
\def\hi@short@userdel#1{%
  \def\reserved@a#1,#1,#2\@mark##3\@stop{%
    \ifstrempy{##3}{\gdef\hi@short@user@list{##1,##2}}%
  }%
  \expandafter\reserved@a\hi@short@user@list\@mark,#1,\@mark\@stop
}
```

This is run from the aux file if a user-defined form is to be suppressed.

```
\def\hi@short@nouser#1{%
  \global\@namedef{hi@short@nouser@#1}{}%
}
```

This runs at the end of the document, to emit unused user-defined short names to the aux file and to issue a notice.

```
\def\hi@short@final{%
  \let\reserved@b\empty
  \for\reserved@a:=\hi@short@user@list\do{%
    \ifx\reserved@a\empty\else
      \begingroup
        \let\protect\@unexpandable@protect
        \immediate\write\@auxout{%
          \noexpand\hi@short@nouser{\reserved@a}%
        }%
      \endgroup
      \@ifundefined{hi@short@nouser@\reserved@a}{%
        \addto@macro\reserved@b{\@spaces}%
        \addto@macro\to@macro\reserved@b\reserved@a
        \addto@macro\reserved@b{\MessageBreak}%
      }{}%
    \fi
  }%
  \ifx\reserved@b\empty\else
    \PackageWarning\hi@pkgname{%
      The following short cite forms went unused:\MessageBreak
      \reserved@b
    }%
  \fi
}
```

```
        They will be removed upon next compilation.\MessageBreak
    }%
\fi
}
\AtEndDocument{\hi@short@final}
```