

The nameauth package

Charles P. Schaum
`charles dot schaum at att dot net`

v0.7 from 2011/12/26

Abstract

These macros help an author to encode the (optional) forenames and (mandatory) surnames of people in order to normalize occurrences in the text and entries in the index. This package is intended for Western-style naming conventions.

1 Introduction

Suppose you were working on a collection of essays. Depending on the permissions governing the essays, you might need to allow variance in the spelling of people's names. You would use either a house reference, a research library, or an encyclopedic source to decide on authoritative name forms. Your index might use abbreviated versions of those forms. Either a substantive or copy editor would keep track of the main name form, the variants, and their relationship to the index. For optimal quality, that would require a proofreading pass with queries to the editor. That adds time and cost to the job. If the author or editor can handle these details in advance, the result trims overhead cost. That could make a prospective author or freelance editor more likely to be selected for publication.

1.1 Design Decisions

This package assumes that an author or editor wants to minimize keystrokes. The default behavior sets the first occurrence of a person's name in small caps and prints out that full name. Subsequent occurrences set the last name only in the running text font. Considerable variation from the defaults is possible.

1.2 Thanks

Thanks to MARC VAN DONGEN, ENRICO GREGORIO, PHILIPP STEPHANI, HEIKO OBERDIEK, and UWE LUECK for their invaluable assistance. Marc showed me the basic structure using the `xparse` package. Enrico and Philipp helped with generating control sequences and sanitizing. Heiko gave a space-removing solution that could be passed as an argument in a macro. Code adapted from Uwe's work on the `texhax` list enabled the routines to function with the `microtype` package.

2 Usage

2.1 Package Options

If the default behavior is not desired, the following options easily alter it. They are, as follows:

<code>mainmatter</code>	The default behavior triggers special typesetting of the first occurrence of a name, starting at the beginning of a document.
<code>frontmatter</code>	This option suppresses the special typesetting of the first occurrence of a name before the invocation of <code>\NamesActive</code> . This option fits well with front matter from a contributor who may not intend the same formatting and emphasis found in the main matter. The indexing and aliasing features of the package remain operative. <i>Note</i> : One can switch at will between formatted and non-formatted sections; see Section 2.6.
<code>smallcaps</code>	The default behavior when a name is first encountered is to print it in small caps.
<code>italic</code>	This option causes the first occurrence of the name to be italicized.
<code>boldface</code>	This option causes the first occurrence of the name to be set in boldface.
<code>noformat</code>	This option suppresses document formatting after the invocation of <code>\NamesActive</code> . If an author wants the indexing and aliasing functions without any special typesetting, this option accomplishes that easily. <i>Note</i> : One can override the name typesetting options manually; see Section 2.5.1. This option is <i>not</i> equivalent to <code>frontmatter</code> . The latter functions independently and never produces any formatting. This approach allows one to toggle formatting on and off without retyping the document. It anticipates variation among publishers' house styles.

2.2 Basic Macros

2.2.1 `\Name` and `\Name*`

`\Name` This macro generates two forms of the name: a printed form in the text and a
`\Name*` form of the name that occurs in the index. The general syntax is:

```
\Name[\langle forename(s) \rangle]{\langle surname(s) \rangle}[\langle alternate forenames \rangle]  
\Name*[\langle forename(s) \rangle]{\langle surname(s) \rangle}[\langle alternate forenames \rangle]
```

From now on we will abbreviate *forename(s)* with *FNN* and *surname(s)* with *SNN* in the syntax descriptions. The unstarred form prints the “full name” at the first occurrence, then only the surname(s) thereafter. A “full name” in the body texts consists of either the regular or the alternate forename(s), followed by the surname(s). Only the “regular” forenames are used in the index. The starred form always prints the full name. Forenames are optional. Surnames are mandatory. Both macros usually apply a different “font attribute” to the name when it first appears in the running text. For example:

```
\Name*[Johann Wolfgang von]{Goethe}  
Print JOHANN WOLFGANG VON GOETHE the first time it appears.  
Later, print the full name Johann Wolfgang von Goethe.  
  
\Name[Johann Wolfgang von]{Goethe}  
Print JOHANN WOLFGANG VON GOETHE the first time it appears.  
Later, print only the last name Goethe.  
  
\Name{Boethius} and \Name*{Boethius}  
Print BOETHIUS the first time it appears. Later, print Boethius.
```

Note: Here I am playing a “dirty trick” by making each line begin as if the name had not yet occurred. See Section 2.7 for more.

2.2.2 Suffix Removal

Observe that `\Name` and `\Name*` appear to have the same effect with Boethius. That is not exactly true. When `\Name` is used with a name containing a comma and a suffix, it removes the suffix after the first use, while `\Name*` does not do this. That results in the following behavior:

```
\Name[Martin Luther]{King, Jr.}  
Print MARTIN LUTHER KING, JR. when it appears and King there-  
after. Use the starred form to get Martin Luther King, Jr.
```

These macros keep track of whether the name ends with the period of an abbreviation like “Jr.” and “Sr.” That should also work with abbreviations like “d. Ä.” (*der Ältere*). Two periods are not printed when the full name is printed at the end of a sentence. The following example shows the possible combinations:

<code>\Name[Martin Luther]{King, Jr.}</code>	MARTIN LUTHER KING, JR.
<code>\Name[Martin Luther]{King, Jr.}</code>	King.
<code>\Name[Martin Luther]{King, Jr.}</code>	King
<code>\Name*[Martin Luther]{King, Jr.}</code>	Martin Luther King, Jr.
<code>\Name*[Martin Luther]{King, Jr.}</code>	Martin Luther King, Jr.

2.3 Advanced Features

2.3.1 Alternate forenames in the Text

In some cultures people receive three or more forenames. It is often inadvisable to use so many names in the index. Yet you might wish to introduce a person with all of his or her names. If nothing else, this is a grandiose flourish that can add a bit of *gravitas* to a historical figure’s role in the text. The “alternate forenames” optional argument comes to the rescue. The following examples show how one might use alternate forenames:

```

\Name[Wilhelm M.L.]{De Wette}[Wilhelm Martin Leberecht]
WILHELM MARTIN LEBERECHE DE WETTE (text, first occurrence)
De Wette (text, second occurrence)
The abbreviated name “De Wette, Wilhelm M.L.” is in the index.

\Name*[Wilhelm M.L.]{De Wette}
WILHELM M.L. DE WETTE (text, first occurrence)
Wilhelm M.L. De Wette (text, second occurrence)
The abbreviated name “De Wette, Wilhelm M.L.” is in the index.

\Name*[Wilhelm M.L.]{De Wette}[W.M.L.]
W.M.L. DE WETTE (text, first occurrence)
W.M.L. De Wette (text, second occurrence)
The abbreviated name “De Wette, Wilhelm M.L.” is in the index.

```

Observe that the alternate forenames override the regular forenames in the text. They do not get printed, however, in subsequent occurrences of `\Name`. The surname argument is always printed. There is, however, the possibility that you might refer to a person with a nickname only. The next macro facilitates this.

2.3.2 `\IndexName`

`\IndexName` This macro allows one to print any text in the body—such as a nickname—yet it will create a standard index entry for the person so named. Its syntax is:

```

\IndexName[<FNN>]{<SNN>}[<text>]

```

`\IndexName` will not interfere with the *nom de plume* mechanism described below. It allows considerable flexibility while retaining the helpful indexing features. For example, a historian might wish to refer to a person with a familiar name when the history text describes close familial relationships:

```

\Name[C.F.W.]{Walther}[Carl Ferdinand Wilhelm] was a major
figure in nineteenth-century American Lutheran history.

```

`\Name[C.F.W.]{Walther}` used his full name when signing official documents. To his family, friends, and close colleagues, however, he was simply `\IndexName[C.F.W.]{Walther}{Ferdinand}`.

CARL FERDINAND WILHELM WALTHER was a major figure in nineteenth-century American Lutheran history. Walther used his full name when signing official documents. To his family, friends, and close colleagues, however, he was simply Ferdinand.

`\IndexName` may be helpful in a pinch, but it lacks certain safeguards for handling pseudonyms that the following macros introduce.

2.3.3 `\PName` and `\AKA`: Pen Names

`\PName` People may use pseudonyms or *noms de plume*. Sometimes they become known by epithets or monikers. There are two macros to handle this. The first of these macros, `\PName`, introduces the name of the person under which he or she will be indexed, followed by the pseudonym in parentheses. Like `\Name` it has starred and unstarred forms, although the star only operates on the main name that is indexed, not the pseudonym. It is likely that one is known more by the pen name, so an abbreviation mechanism for lesser-known information seems counterintuitive for clarity. The syntax is:

`\PName[⟨FNN⟩]{⟨SNN⟩}[⟨alternate FNN⟩]{⟨alternate SNN⟩}`

The following examples should make clear its use:

`\PName*[Mark]{Twain}[Samuel L.]{Clemens}`

Print MARK TWAIN (Samuel L. Clemens) the first time it appears.
Later, print Mark Twain (Samuel L. Clemens).

`\PName[Mark]{Twain}[Samuel L.]{Clemens}`

Print MARK TWAIN (Samuel L. Clemens) the first time it appears.
Later, print Twain (Samuel L. Clemens).

`\PName*[Willebrord]{Snel van Royen}{Snellius}`

Print WILLEBRORD SNEL VAN ROYEN (Snellius) the first time it appears. Later, print Willebrord Snel van Royen (Snellius).

`\PName[Willebrord]{Snel van Royen}{Snellius}`

Print WILLEBRORD SNEL VAN ROYEN (Snellius) the first time it appears. Later, print Snel van Royen (Snellius).

`\PName{Voltaire}[François-Marie]{Arouet}`

Print VOLTAIRE (François-Marie Arouet) the first time it appears.
Later, print Voltaire (François-Marie Arouet). The starred version also produces Voltaire (François-Marie Arouet)

Observe carefully that `\PName` does not allow the “optional forenames” parameter used in `\Name`. This is done intentionally to reduce the possible confusion of a double alias.

`\AKA` The second macro that handles aliases is `\AKA`. It is the same alias mechanism used in `\PName`, but separated from the normal indexing feature for more flexibility. Its syntax is similar to `\PName`:

```
\AKA[⟨FNN⟩]{⟨SNN⟩}[⟨alternate FNN⟩]{⟨alternate SNN⟩}
```

`\AKA` only prints the pseudonym, not the indexed name. The macro also will prevent the double-printing of a period after an abbreviation. Here are a few examples of how to use it:

```
\AKA[George]{Eliot}[Mary Anne]{Evans}...
```

```
\Name[George]{Eliot}
```

Today we consider Mary Anne Evans and her literary contributions as
GEORGE ELIOT.

```
\PName{Thomas Aquinas}{Thomas of Aquino}...
```

Perhaps the greatest medieval theologian was THOMAS AQUINAS
(Thomas of Aquino), known as *Doctor Angelicus*. His name “Aquinas”
is not a surname, so many modern scholars refer to him as Thomas.

To index *Doctor Angelicus* as a pseudonym one must use the following method in order to format the index entry correctly. Note that the “actual” character @ may be redefined.

```
\index{Doctor Angelicus@\textit{Doctor Angelicus}|see {Thomas Aquinas}}
```

`\AKA` creates an index entry that cross-references to the “main” name given. It is assumed that a `\Name` macro occurs somewhere in reference to the indexed name. No error checking otherwise occurs for this. As a general rule, the more one has to account for exceptions, the shakier the rules get. My own use of these macros involves people with English, French, German, Italian, Latin, and Central European names. If all you ever index are John Smith and Jerry Jones, you may never need any macros beyond `\Name`.

Pseudonyms cannot be indexed with page numbers. They only work as “see” references that point to name entries in the index that do have page references. `\Name` will print a pseudonym if it is used and emit a warning. If you want to index a pseudonym with page numbers, use `\Name` for both the main name and the pseudonym, then manually cross-reference them.

`\AKA` will only create one *see* reference per pseudonym. This has the side effect of excluding the special case where a pseudonym applies to multiple people: Willebrord Snel van Royen (Snellius) and his son RUDOLPH SNEL VAN ROYEN (Snellius) have the same Latin moniker.

You will see in the index that Snellius points to two entries. I had to add a manual index entry to work around it:

```
\index{Snellius|see {Snel van Royen, Rudolph}}
```

Otherwise the moniker Snellius would point in the index only to the first occurrence, which is Willebrord Snel van Royen (Snellius). This is one case when *not* to use `\IndexName`—it would involve more typing and still would not create the desired index reference.

Be careful with name collisions, especially with more than one person having the same last name:

```
\PName[Willebrord]{Snel van Royen}{Snellius}
Snel van Royen (Snellius)

\PName[Rudolph]{Snel van Royen}{Snellius}
Snel van Royen (Snellius)

\Name[Willebrord]{Snel van Royen}
Snel van Royen

\Name[Rudolph]{Snel van Royen}
Snel van Royen
```

2.4 Accented Names

The following Unicode accents will work in names using UTF8 and `inputenc`:

À Á Â Ã Ä Å Æ	Ç È É Ê Ë	Ì Í Î Ï Ð Ñ	FIRST USE
À Á Â Ã Ä Å Æ	Ç È É Ê Ë	Ì Í Î Ï Ð Ñ	second use
Ò Ó Ô Õ Ö Ø	Ù Ú Û Ü Ý	Þ ß	FIRST USE
Ò Ó Ô Õ Ö Ø	Ù Ú Û Ü Ý	Þ ß	second use
À Á Â Ã Ä Å Æ	Ç È É Ê Ë	Ì Í Î Ï Ð Ñ	FIRST USE
à á â ã ä å æ	ç è é ê ë	ì í î ï ð ñ	second use
Ò Ó Ô Õ Ö Ø	Ù Ú Û Ü Ý	Þ ÿ	FIRST USE
ò ó ô õ ö ø	ù ú û ü ý	þ ÿ	second use
Ǻ ǻ Ǽ Ǿ ǿ ǿ ǿ	Ǿ ǿ ǿ ǿ ǿ ǿ ǿ	Ǿ ǿ ǿ ǿ ǿ ǿ ǿ	FIRST USE
Ǻ ǻ Ǽ Ǿ ǿ ǿ ǿ	Ǿ ǿ ǿ ǿ ǿ ǿ ǿ	Ǿ ǿ ǿ ǿ ǿ ǿ ǿ	second use
Ĳ Ĳ Ĳ Ĳ Ĳ Ĳ	Ń Ń Ń Ń Œ œ	Ř ř Ŕ ŕ	FIRST USE
Ĳ Ĳ Ĳ Ĳ Ĳ Ĳ	Ń Ń Ń Ń Œ œ	Ř ř Ŕ ŕ	second use
Š š Š š Ť ť Ť ť	Ů ů Ů ů	Ž ž Ž ž Ž ž	FIRST USE
Š š Š š Ť ť Ť ť	Ů ů Ů ů	Ž ž Ž ž Ž ž	second use

Other accents will not work unless you use `TEX` control sequences or `XYLATEX`. You can also include the `TS1` encoding and do something like the following with the `inputenc` package, the `newunicodechar` package, and `UTF8`:

```
\DeclareTextSymbolDefault{\textlongS}{TS1}
\DeclareTextSymbol{\textlongS}{TS1}{115}
\newunicodechar{f}{\textlongS}
\newunicodechar{ā}{\=a}
\newunicodechar{ṁ}{\d{m}}
```

Please remember that the appropriate font packages, such as `lmodern` or the TeX Gyre fonts, are needed to obtain some TS1 glyphs. Also there may be points where `pdflatex` will accept the input (e.g., Ghāzali), but `makeindex` will have problems with that. In “normal” operation (I have used `article` and `memoir`) these control sequences appear to work. Yet in generating this style file that was not the case. You may need to use manual index entries for those cases.

2.5 Name Formatting

2.5.1 Font Attributes

`\NamesFormat` The first time a name is printed, it is formatted with the font attribute stored in `\NamesFormat`. This is set with the class options or manually. `\NamesFormat` can use either the command form or the declaration form of selecting font attributes, e.g., `\textsc` or `\scshape`. By redefining this macro, one can “hook” into the special typesetting of the first occurrence of a name. Consider the following:

```
\renewcommand{\NamesFormat}[1]{\textsc{#1}%
\ifinner\else\marginpar{#1}\fi}
```

2.6 Formatting Certain Sections

`\NamesActive` Using the `frontmatter` option will deactivate formatting until `\NamesActive` ac-
`\NamesInactive` tivate it. Another macro, `\NamesInactive`, will deactivate formatting. These two
 macros toggle formatting on and off. They can be used throughout the document
 Here we switch to the “front matter” mode with `\NamesInactive`:

```
\Name[Rudolph]{Carnap}      Rudolph Carnap
\Name[Rudolph]{Carnap}      Carnap
\Name[Nicolas]{Malebranche} Nicolas Malebranche
\Name[Nicolas]{Malebranche} Malebranche
```

Then we switch back to “main matter” mode with `\NamesActive`:

```
\Name[Rudolph]{Carnap}      RUDOLPH CARNAP
\Name[Rudolph]{Carnap}      Carnap
\Name[Nicolas]{Malebranche} NICOLAS MALEBRANCHE
\Name[Nicolas]{Malebranche} Malebranche
```

See how the mechanism works in a complementary, yet independent manner. The index entries all will have the same format.

2.7 Tweaks:\ForgetName

`\ForgetName` The macro `\ForgetName` is a “dirty trick” of sorts that takes the optional and mandatory parameters used by `\Name` to create an index entry. Thus it does not take the final optional parameter. It causes the macros to “forget” prior uses of a name with respect to typesetting. The next use will print as if it were a “first use.” Index entries and pseudonyms (see above) are *never* forgotten.

3 Implementation

3.1 Class Options and Required Packages

```
1 \newif\if@nameauth@DoFormat
2 \newif\if@nameauth@Punct
```

These boolean values control whether or not formatting of first entries occurs and whether or not a name ends with a period.

```
3 \DeclareOption{mainmatter}{\@nameauth@DoFormattrue}
4 \DeclareOption{frontmatter}{\@nameauth@DoFormatfalse}
5 \DeclareOption{smallcaps}{\newcommand{\NamesFormat}{\scshape}}
6 \DeclareOption{italic}{\renewcommand{\NamesFormat}{\itshape}}
7 \DeclareOption{boldface}{\renewcommand{\NamesFormat}{\bfseries}}
8 \DeclareOption{noformat}{\renewcommand{\NamesFormat}{}}
9 \ExecuteOptions{smallcaps,mainmatter}
10 \ProcessOptions\relax
11 \RequirePackage{etoolbox}
12 \RequirePackage{xparse}
```

3.2 Internal Macros

`\@CleanName`

```
13 \newcommand*{\@CleanName}[1]{\expandafter\zap@space\detokenize{#1}\@empty}
```

Thanks to Heiko Oberdiek, this macro produces a “sanitized” string based on the forename/surname parameters of `\Name` and friends. With this we can construct a control sequence name (`\csname`). Testing for the presence of that control sequence determines the existence of pseudonyms and the first occurrence of a name.

`\@Zapsuffix`

```
14 \newcommand{\@Zapsuffix}[1]{%
15 \def\@@Zapsuff##1,##2*{##1}%
16 {\@@Zapsuff#1,*}%
17 }
```

This macro does what it says. Anything starting with a comma and ending with the end of the name is stripped off. That includes “Sr.,” “Jr.,” “III,” and so on.

`\@CheckDot`

```
18 \def\@CheckDot{\futurelet\@token\@EvalDot}
```

This macro assigns the lookahead token `\@token` to be evaluated by `\@EvalDot` while keeping `\@token` non-destructively on the list of input tokens. I use this method instead of `\@nextchar` because I do not want to gobble spaces.

`\@EvalDot`

```
19 \def\@EvalDot{\let\@period=. \ifx\@token\@period\expandafter\@gobble \fi}
```

\@EvalDot checks if \@token is a period. If so it gobbles it by using \expandafter to get past the grouping. Another \expandafter occurs immediately before the invocation of \@CheckDot in \Name and \AKA.

\@TestDot

```

20 \newcommand*\@TestDot}[1]{%
21 \def\TestDot##1.\TestEnd##2\TestStop{\TestPunct{##2}}%
22 \def\TestPunct##1{\ifx\TestPunct##1\TestPunct\else\@nameauth@Puncttrue\fi}%
23 \@nameauth@Punctfalse%
24 \TestDot#1\TestEnd.\TestEnd\TestStop%
25 }

```

While \@CheckDot looks *ahead* for a period, \@TestDot—based on a snippet by Uwe Lueck—checks for a terminal period in the name passed to it, ignoring medial periods. This string test is compatible with the microtype package.

\@FmtName

```

26 \DeclareDocumentCommand\@FmtName{s m}%
27 {%
28 \@TestDot{#2}%
29 \IfBooleanTF{#1}{#2}{\bgroup\NamesFormat{#2}\egroup}%
30 }

```

\@FmtName is where the first occurrences of a name are formatted. Notice how \NamesFormat sits between a \bgroup and an \egroup to localize the font change. It is adjacent to the brackets that enclose the second parameter, allowing one to hook into the name formatting procedure.

\@FName

```

31 \DeclareDocumentCommand\@FName{o m o}%
32 {%
33   \IfValueTF{#1}%
34   {\IfValueTF{#3}{\def\@ForeNames{#3}}{\def\@ForeNames{#1}}%
35   \ifcsname\@CleanName{#1#2!PN!}\endcsname%
36     \@FmtName*{#1 #2}%
37     \PackageWarning{nameauth}%
38     {You cannot create a page reference from the pen name: #1 #2.}%
39   \else%
40     \if@nameauth@DoFormat%
41       \ifcsname\@CleanName{#1#2!MN!}\endcsname%
42       \@FmtName*{\@ForeNames{\space}#2}\index{#2,{\space}#1}%
43     \else%
44       \csgdef{\@CleanName{#1#2!MN!}}{}%
45       \@FmtName*{\@ForeNames{\space}#2}\index{#2,{\space}#1}%
46     \fi%
47   \else%
48     \ifcsname\@CleanName{#1#2!NF!}\endcsname%
49     \@FmtName*{\@ForeNames{\space}#2}\index{#2,{\space}#1}%
50   \else%

```

```

51             \csgdef{\@CleanName{#1#2!NF!}}{}}}%
52             \@FmtName*{\@ForeNames{\space}#2}\index{#2,{\space}#1}%
53         \fi%
54     \fi%
55 \fi}%
56 {\ifcsname\@CleanName{#2!PN!}\endcsname%
57     \@FmtName*{#1 #2}%
58     \PackageWarning{nameauth}%
59     {You cannot create a page reference from the pen name: #1 #2.}%
60 \else%
61     \if@nameauth@DoFormat%
62         \ifcsname\@CleanName{#2!MN!}\endcsname%
63             \@FmtName*{#2}\index{#2}%
64         \else%
65             \csgdef{\@CleanName{#2!MN!}}{}}}%
66             \@FmtName{#2}\index{#2}%
67         \fi%
68     \else%
69         \ifcsname\@CleanName{#2!NF!}\endcsname%
70             \@FmtName*{#2}\index{#2}%
71         \else%
72             \csgdef{\@CleanName{#2!NF!}}{}}}%
73             \@FmtName*{#2}\index{#2}%
74         \fi%
75     \fi%
76 \fi}%
77 }

```

\@FName (full name) first checks for the optional forenames argument. Two outcomes are possible. Either the forenames and surnames will be printed, or the surnames only will be printed. One cannot have “alternate” forenames with a surname-only entry. In that case, one must use either \AKA or \IndexName.

Next \@FName checks if the “alternate” forenames exist. If so, they are used as forenames for the body text and the “regular” forenames are used in the index. Otherwise the “regular” forenames are used for both.

Third, a check occurs for a control sequence name based on the names given and the suffix !PN! (for pen name) wrapped in a pair of bangs. This format is used by \AKA to create the index *see* cross-references. If such a control sequence exists, then \@FName prints the name given, emits a warning, and creates no index entry. Nor will “alternate forenames” be examined in this case. One cannot use \ForgetName to expunge a pen name. This is a deliberate decision to avoid corruption of the index cross-references.

The fourth branch involves the boolean value @nameauth@DoFormat, which is controlled by \NamesActive and \NamesInactive. If formatting is active, choose the unstarred form of \@FmtName, which applies the formatting hook. Otherwise use the starred form that applies no formatting.

The state of @nameauth@DoFormat also controls the suffix used: !MN! for main name or !NF! for no format. Suffixes are enclosed in pairs of bangs.

\@LName

```
78 \DeclareDocumentCommand\@LName{o m o}%
79 {%
80     \IfValueTF{#1}%
81     {\IfValueTF{#3}{\def\@ForeNames{#3}}{\def\@ForeNames{#1}}%
82     \ifcsname\@CleanName{#1#2!PN!}\endcsname%
83         \@FmtName*{#1 #2}%
84         \PackageWarning{nameauth}%
85         {You cannot create a page reference from the pen name: #1 #2.}%
86     \else%
87         \if@nameauth@DoFormat%
88             \ifcsname\@CleanName{#1#2!MN!}\endcsname%
89                 \@FmtName*{\@Zapsuffix{#2}}\index{#2,{\space}#1}%
90             \else%
91                 \csgdef{\@CleanName{#1#2!MN!}}{}%
92                 \@FmtName{\@ForeNames{\space}#2}\index{#2,{\space}#1}%
93             \fi%
94         \else%
95             \ifcsname\@CleanName{#1#2!NF!}\endcsname%
96                 \@FmtName*{\@Zapsuffix{#2}}\index{#2,{\space}#1}%
97             \else%
98                 \csgdef{\@CleanName{#1#2!NF!}}{}%
99                 \@FmtName*{\@ForeNames{\space}#2}\index{#2,{\space}#1}%
100             \fi%
101         \fi%
102     \fi}%
103     {\ifcsname\@CleanName{#2!PN!}\endcsname%
104         \@FmtName*{#1 #2}%
105         \PackageWarning{nameauth}%
106         {You cannot create a page reference from the pen name: #1 #2.}%
107     \else%
108         \if@nameauth@DoFormat%
109             \ifcsname\@CleanName{#2!MN!}\endcsname%
110                 \@FmtName*{\@Zapsuffix{#2}}\index{#2}%
111             \else%
112                 \csgdef{\@CleanName{#2!MN!}}{}%
113                 \@FmtName{#2}\index{#2}%
114             \fi%
115         \else%
116             \ifcsname\@CleanName{#2!NF!}\endcsname%
117                 \@FmtName*{\@Zapsuffix{#2}}\index{#2}%
118             \else%
119                 \csgdef{\@CleanName{#2!NF!}}{}%
120                 \@FmtName*{#2}\index{#2}%
121             \fi%
122         \fi%
123     \fi}%
124 }
```

\@LName (last name only) differs little from \@FName apart from its use of \@ZapSuffix and the last name only *except* in the case of the first occurrence of a name. One could make a good case that \@FName and \@LName should be combined. When I tried that, the result introduced bugs that proved difficult to track down. This version simply worked, so I stuck with it.

Marc van Dongen provided the basic structure for these two macros.

3.3 User Interface Macros

\Name

```
125 \DeclareDocumentCommand\Name{s o m o}%
126 {%
127     \@nameauth@Punctfalse%
128     \IfBooleanTF{#1}{\@FName[#2]{#3}{#4}}{\@LName[#2]{#3}{#4}}%
129     \if@nameauth@Punct\expandafter\@CheckDot\fi%
130 }
```

\Name and \Name* call \@LName and \@FName, respectively. The use of \expandafter before \@CheckDot works with the other use of \expandafter mentioned with \@EvalDot above to move past the closing brace and fetch the period as lookahead. Yet that is only done when the check for a terminal period in the name succeeds.

\PName

```
131 \DeclareDocumentCommand\PName{s o m o m}%
132 {%
133     \IfBooleanTF{#1}{\Name*[#2]{#3}}{\Name[#2]{#3}}%
134     {\space}\AKA[#2]{#3}{#4}{#5}%
135 }
```

\PName is a convenience macro whose starred and unstarred forms call the respective versions of \Name, then \AKA.

\AKA

```
136 \DeclareDocumentCommand\AKA{o m o m}%
137 {%
138     \@nameauth@Punctfalse%
139     \IfValueTF{#3}%
140     {\@FmtName*{#3 #4}\ifcsname\@CleanName{#3#4!PN!}\endcsname\relax\else%
141         \csgdef{\@CleanName{#3#4!PN!}}{}}%
142     \IfValueTF{#1}%
143         {\index{#4,{\space}#3|see{#2,{\space}#1}}}%
144         {\index{#4,{\space}#3|see{#2}}}%
145     \fi}%
146     {\@FmtName*{#4}\ifcsname\@CleanName{#4!PN!}\endcsname\relax\else%
147         \csgdef{\@CleanName{#4!PN!}}{}}%
148     \IfValueTF{#1}%
149         {\index{#4|see{#2,{\space}#1}}}%

```

```

150             {\index{#4|see{#2}}}%
151         \fi}%
152     \if@nameauth@Punct\expandafter\@CheckDot\fi%
153 }

```

\AKA prints a pseudonym and creates index cross-references. It also checks to see if cross-references already have been generated, and if so, it does not do it again. Like \Name it checks for a terminal period, but only for the *pseudonym*.

\IndexName

```

154 \DeclareDocumentCommand\IndexName{o m m}%
155 {%
156     \IfValueTF{#1}%
157         {\ifcsname\@CleanName{#1#2!PN!}\endcsname\else\index{#2,{\space}#1}\fi}%
158         {\ifcsname\@CleanName{#2!PN!}\endcsname\else\index{#2}\fi}%
159     {#3}%
160 }

```

\IndexName creates an index entry that is not already a pseudonym. It also prints text, but not necessarily the name that is indexed.

\ForgetName

```

161 \DeclareDocumentCommand\ForgetName{o m}%
162 {%
163     \IfValueTF{#1}%
164         {\csundef{\@CleanName{#1#2!MN!}}\csundef{\@CleanName{#1#2!NF!}}}%
165         {\csundef{\@CleanName{#2!MN!}}\csundef{\@CleanName{#2!NF!}}}%
166 }

```

\ForgetName uses the ε -TeX facilities to undefine control sequences that \@FName and \@LName use to define them. Using \ifdefined does not create a control sequence equivalent to \relax, thus using less resources.

\NamesInactive

```

167 \newcommand{\NamesInactive}{\@nameauth@DoFormatfalse}

```

This macro deactivates formatting, even as its counterpart below activates it.

\NamesActive

```

168 \newcommand{\NamesActive}{\@nameauth@DoFormattrue}
169 \endinput

```

Change History

v0.7
 General: Initial version 1

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; numbers in roman refer to the code lines where the entry is used.

Symbols		C	69, 82, 88, 95,
<code>\@@Zapsuff</code>	15, 16	Carnap, Rudolph	8
<code>\@CheckDot</code> <u>18</u> , 129, 152		Clemens, Samuel L. . .	140, 146, 157, 158
<code>\@CleanName</code> <i>see</i> Twain, Mark	<code>\index</code> 42, 45, 49, 52,
. <u>13</u> , 35, 41, 44,		<code>\csgdef</code>	44, 51,
48, 51, 56, 62,		65, 72, 91, 98,	63, 66, 70, 73,
65, 69, 72, 82,		112, 119, 141, 147	89, 92, 96, 99,
88, 91, 95, 98,		<code>\csundef</code>	110, 113, 117,
103, 109, 112,		164, 165	120, 143, 144,
116, 119, 140,			149, 150, 157, 158
141, 146, 147,		D	<code>\IndexName</code>
157, 158, 164, 165		De Wette, Wilhelm	4, <u>154</u>
<code>\@EvalDot</code>	18, <u>19</u>	M.L.	4
<code>\@FName</code>	<u>31</u> , 128	<code>\DeclareDocumentCommand</code>	K
<code>\@FmtName</code>		26, 31, 78, 125,	King, Jr., Martin
. <u>26</u> , 36, 42, 45,		131, 136, 154, 161	Luther
49, 52, 57, 63,		<code>\DeclareOption</code>	3, 4
66, 70, 73, 83,		<code>\detokenize</code>	L
89, 92, 96, 99,		13	Lueck, Uwe
104, 110, 113,		<i>Doctor Angelicus</i> . <i>see</i>	1, 10
117, 120, 140, 146		Thomas Aquinas	M
<code>\@ForeNames</code> 34, 42, 45,		Dongen, Marc van . .	Malebranche, Nicolas . .
49, 52, 81, 92, 99		.. 1, 13	8
<code>\@LName</code>	<u>78</u> , 128	E	N
<code>\@TestDot</code>	<u>20</u> , 28	Eliot, George	6
<code>\@Zapsuffix</code>		Evans, Mary Anne . .	<code>\Name</code>
<u>14</u> , 89, 96, 110, 117		. <i>see</i> Eliot, George	3, <u>125</u> , 133
<code>\@nameauth@DoFormatfalse</code>		<code>\expandafter</code>	name forms
.	4, 167	.. 13, 19, 129, 152	3
<code>\@nameauth@DoFormattrue</code>		F	<code>\Name*</code>
.	3, 168	<code>\ForgetName</code>	3
<code>\@nameauth@Punctfalse</code>		8, <u>161</u>	<code>\NamesActive</code>
.	23, 127, 138	<code>\futurelet</code>	8, <u>168</u>
<code>\@nameauth@Puncttrue</code> 22		18	<code>\NamesFormat</code>
<code>\@period</code>	19	G	5–8, 8, 29
<code>\@token</code>	18, 19	Goethe, Johann Wolf-	<code>\NamesInactive</code>
		gang von	<u>167</u>
		Gregorio, Enrico	<code>\NamesInctive</code>
		1	8
A		I	O
<code>\AKA</code>	6, 134, <u>136</u>	<code>\if@nameauth@DoFormat</code>	Oberdiek, Heiko
Arouet, François-Marie		. 1, 40, 61, 87, 108	1, 9
.	<i>see</i> Voltaire	<code>\if@nameauth@Punct</code> .	P
B		2, 129, 152
Boethius	3	<code>\ifcsname</code>	<code>\PackageWarning</code> . . .
		35,	. . . 37, 58, 84, 105
		41, 48, 56, 62,	<code>\PName</code>
			5, <u>131</u>
			<code>\ProcessOptions</code> . . .
			10
			S
			Snel van Royen,
			Rudolph
			6, 7
			Snel van Royen, Wille-
			brord
			5–7

Snellius	<i>see</i> Snel van Royen, Wilibrord, <i>see</i> Snel van Royen, Rudolph	\TestEnd	21, 24	V	
Stephani, Philipp	\TestPunct	21, 22	Voltaire	5
		\TestStop	21, 24	W	
		Thomas Aquinas	6	Walther, C.F.W.	5
		Thomas of Aquino	<i>see</i> Thomas Aquinas	Z	
T		Twain, Mark	5	\zap@space	13
\TestDot	21, 24				