The ccool package*

Erwann Rogard[†]

Released 2020/04/12

Abstract

ccool stands for Custom COntent Oriented for IATeX, a concept pioneered by $\mathsf{cool}[1]^1$. This is done using a minimalist interface built upon $\mathsf{xparse}[4]$. Specifically, $\mathsf{Ccool} < \langle name \rangle >$ begins a series of instructions alternating between 'text' and macro definitions, that themselves optionally expand using predefined or inline rules. For example,

```
\Ccool<Math>[Let~]
i{\mathbb{#1}}{ Nat = N, Real = R }*s{{~\rm{and}~}}
[~denote the natural and real numbers.]{}
```

expands to: "Let \mathbb{N} and \mathbb{R} denote the natural and real numbers." As a side effect, $\$ \Nat<\Math>\$ encodes "\N" (and likewise for \Real). Math being the default for $\langle name \rangle$, <\Math> can be dropped. Optionally, the macros can be written to a file, and read, which can be useful for typesetting documents sharing the same notation.

Contents

I	Usage
0	Convention
L	Loading the package
2	\Ccool
	$2.1 < \langle tl_1 \rangle > \dots$
	$2.2 [\langle tl_2 \rangle]$
	2.3 $i\{\langle code_1 \rangle\}$
	$2.4 \{\langle \dot{k}vl_1 \rangle\}^{\frac{1}{2}} \dots $
	2.5 +
	2.6 *
	2.7 $s\{\{\langle tl_3\rangle\} \{\langle tl_4\rangle\} \{\langle tl_3\rangle\}\{\langle tl_4\rangle\}\{\langle tl_5\rangle\}\}$
	$2.8 o\{\langle code_2 \rangle\}$
	$2.9 [\langle tl_6 \rangle] \dots \dots \dots \dots \dots \dots \dots \dots \dots $

^{*}This file describes version v1.8, last revised 2020/04/12.

 $^{^\}dagger {\it firstname}$ dot lastname Aus
Tria gmail dot com

¹Whereas cool provided predefined macros, ccool is tool for making macros, hence "custom".

4	\CcoolHook	5
5	\CcoolLambda	5
6	\CcoolOption 6.1 Expans 6.2 File 6.3 Inner 6.4 Name 6.5 Outer 6.6 Separ 6.7 Write	5 6 6 6 6 6
7	\CcoolRead	6
8	\CcoolVers	6
9	Do's and dont's	6
II	Listing	8
	ing 1. Version.	8
	ing 2. Preamble.	8
	ing 3. Separators.	8
	ing 4. Hello, world!	8
	ing 5. Listing 4 read from file.	9
List	ing 6. Probability space.	9
List	ing 7. Listing 6 read from file.	9
List	ing 8. Mittelwertsatz für n Variable.	10
List	ing 9. Listing 8 read from file.	10
List	ing 10. Lambda expression.	10
List	ing 11. Listing 10 read from file.	11
List	ing 12. CUSUM statistic.	11
List	ing 13. Listing 12 read from file.	12
ш	Other	13
1	Acknowledgment	13

2	Install	13
3	Issue	13
4	Support	13
5	Testing 5.1 Technicality 5.2 Platform 5.3 Engine 5.4 Results 5.5 Other 5.5 Other	13 13 13 13 13 14
Cha	ange History	14
Ind	ex	15
IV	Implementation	19
1	aux	19
2	lambda	21
3	log	22
4	make_key	23
5	make_ccool	23
6	msg	25
7	option	25
8	prop	26
9	seq	28
10	Front-end	28
11	Misc	30

Part I

Usage

Convention

1. Loosely, those of [2] and [4], for example as to the meaning of $\langle token\ list \rangle$.

2. If unspecified, the environment in which a macro must be declared is document.

\usepackage

\usepackage{ccool}

Requirement

- 1. ccool.sty is in the path of the LATEX engine. See Part III, section 4.
- 2. Declare it in the preamble

```
\Ccool
                \cool<\langle tl_1\rangle>
                [\langle t1_2 \rangle]
                i\{\langle code_1 \rangle\}
                \{\langle kvl_1 \rangle\}
                \mathtt{s}\{\{\langle\mathtt{t}1_3\rangle\}|\{\langle\mathtt{t}1_3\rangle\}\{\langle\mathtt{t}1_4\rangle\}|\{\langle\mathtt{t}1_3\rangle\}\{\langle\mathtt{t}1_4\rangle\}\{\langle\mathtt{t}1_5\rangle\}\}
                o\{\langle code_2 \rangle\}
                [\langle t1_6 \rangle]
                Requirement \langle kvl_1 \rangle is specified (all others optional).
   \langle \mathtt{tl}_1 \rangle
                Example Math, ModelA, ModelB
                Semantics Identifies a group of macros
   \langle {\tt tl}_2 \rangle
                Example Let~
                Semantics Expands \langle tl_2 \rangle
\langle code_1 \rangle
                Example \mathbb{#1}
                Semantics
                             1. \langle val_i \rangle \leftarrow \langle code_1 \rangle applied to \langle val_i \rangle
  \langle kvl_1 \rangle
                Example Elems={\omega_1, \dots, \omega_n}, Sample=\Omega
                Semantics
                             2. \langle key_i \rangle \langle tl_1 \rangle \rightarrow \langle val_i \rangle defined in step 1, using Expans for expansion.
                             3. If Write, writes the input used by step 2 to File
```

Other Needed to make \Ccool's side effect within a local group persist thereafter Semantics Appends step 2 and step 3 to \CcoolHook

*

Expans

Semantics

4. Expands $\langle code_2 \rangle$ applied to the list created in step 1, using the separator specified by $\langle tl_3 \rangle$, $\langle tl_4 \rangle$, $\langle tl_5 \rangle$.

```
\langle tl_3 \rangle
                     Example {~\in~}
           \langle t1_4 \rangle
                     Example {,~}
           \langle tl_5 \rangle
                     Example {~\&~}
        \langle code_2 \rangle
                      \textbf{Example $\left\{ \text{$1\right\}} } 
           \langle t1_6 \rangle
                     Semantics \cool<\langle tl_1\rangle>[\langle tl_6\rangle]
                     \coolClear<\langle keyval\ list \rangle>
 \CcoolClear
                     Semantics Clears any data created by \langle Ccool\{\langle tl_1\rangle \}, for all \langle tl_1\rangle in \langle keyval\ list\rangle
   \CcoolHook
                     \CcoolHook
                     Example \AfterEndEnvironment{theorem}{\CcoolHook}
\CcoolLambda
                     \verb|\CcoolLambda[|\langle integer|\rangle]| \{\langle code|\rangle\}|
                     Example \Ccool{ EvalAt = \CcoolLambda{(#1)} }
                     Semantics Creates a lambda expression with \langle integer \rangle arguments for \langle code \rangle
\CcoolOption
                     \verb|\CcoolOption{|} \langle kvl0 \rangle \}|
                     Semantics Set default options for \Ccool
```

```
Default xo
             Syntax Either of eo, ee, ex, xe, xo, xe, xx
      File
             Default ccool\pdfcreationdate
             Syntax Expands to a valid path
     Inner
             Default ####1
             Semantics Default for \langle code_1 \rangle
             \mathbf{Syntax} Use ####1 as the argument to be replaced
      Name
             Default Math
             Semantics Default for \langle tl_1 \rangle
     Outer
             Default \ensuremath{####1}
             Semantics Default for \langle code_2 \rangle
             Syntax Use ####1 as the argument to be replaced
     Separ
             Semantics Default for separators' parameter
             Syntax That of 'separators' in [2, Section 8 of I3seq]
     Write
             Default \BooleanFalse
             Syntax Boolean
\CcoolRead
             \CcoolRead[\langle path \rangle]
             Other The default for \langle path \rangle is the last write-file (see \langle kvl_1 \rangle)
             Semantics
                     1. Reads the definitions in \langle path \rangle.
                     2. Writes to ccool.log: 'read from \langle path \rangle'
\CcoolVers
             \CcoolVers
```

Semantics Expands to the package's version

Do's and dont's

Part II

Listing

```
Listing 1.

% \CcoolVers
%
2020/04/12 v1.8 cool — A tool for encoding mathematical notation
```

```
Listing 2. Preamble<sup>a</sup>

These are the settings to replicate the listings. For exhaustivity, check the documentation section of ccool.dtx.

"\text{\usepackage{amsmath, amsthm, commath}}} \usepackage[T1]{fontenc}\"\char`[
```

```
Listing 3. Separators
        \CcoolOption{
        ^^A% spaces betw. inner and outer brackets matter!->
%
        Separ=\{\{\ \char`@\ \}\{\ \'\%\ \}\{\ \char`@\ \}\}\}
        \ \Ccool<Test>\{ X = x, Y = y \}*[\]
        \{ X = x, Y = y, Z = z \} * [ \setminus  ]
        \{ X = x, Y = y \}*s\{\{ \setminus \& \setminus \}\}[ \setminus ]
        \{ X = x, Y = y \}*s\{\{ \setminus \& \setminus \}\{, \setminus \}\}[ \setminus ] 
        \{ X = x, Y = y, Z = z \}*s\{\{\setminus \&\setminus \}\}[\setminus\setminus]
%
%
        \{ X = x, Y = y, Z = z \}*s\{\{\ \ \ \ \ \}\{,\ \ \}\}[\ \ ]
        { X = x, Y = y, Z = z }*s{{\ \&\ }{,\ }{\ \&\ }}\\
x @ y
x~\%~y~@~z
x \& y
x \& y
x \& y \& z
x, y \& z
x, y \& z
```

```
%
       \Ccool<Test>
       \{ \text{ KeyA = } \{.\}, \text{ KeyB = } \{!\}, \text{ KeyC = } \{\\} \}[]
       \{ \text{ KeyD} = \{d\}, \text{ KeyE} = \{\\%\} \}[]i\{\\#1\\}
       \{ KeyF = \{H\}, KeyG = \{e\}, KeyH = \{1\} \}*[]
       { KeyI = {\\%}, KeyJ = {\\%}, KeyK = {\\%} }[.\\{1\\}.\\{o\\}]
%
       { KeyL = {1}, KeyM = {\char`[}, KeyN = {\char`]} }[]
       \{ \text{ KeyO} = \{o\}, \text{ KeyP} = \{\'\}, \text{ KeyQ} = \{\'\} \}[\{,\ \}]
       { KeyR = \{w\}, KeyS = \{o\}, KeyT = \{r\} \}*s\{{\}}{\}}o\{{\hat {p}} = {\{r\}\}}
       \{ \text{ KeyU = } \{\\\\\\\\\\\\\\\\\\} \{ \text{ KeyV = } \{\\\\\\\\\\\\\\\\\} \}[]
       { KeyX = {\%}, KeyY = {\%}, KeyZ = {\KeyB<Test>} }\nobreak
%
       \CcoolOption{ Write = \BooleanFalse }
\{H\}.\{e\}.\{l\}.\{o\}, [world!]
```

```
Listing 5. Listing 4 read from file.
      \CcoolRead
%
      \verb|\KeyF<Test>\KeyA<Test>\\nobreak|
%
%
      \KeyG<Test>\KeyA<Test>\nobreak
%
      \KeyH<Test>\KeyA<Test>\nobreak
%
      \KeyH<Test>\KeyA<Test>\nobreak
%
      {\{\}\} (X) = {\}, {\}\}, {\} \
%
      \KeyM<Test>\KeyR<Test>\nobreak
      \Key0<Test>\nobreak
%
      \KeyT<Test>\nobreak
%
      \KeyL<Test>\nobreak
%
      \KeyD<Test>\nobreak
%
      \KeyZ<Test>\nobreak
      \KeyN<Test>\nobreak
%
\{H\}.\{e\}.\{l\}.\{o\}, [world!]
```

```
Listing 7. Listing 6 read from file.
       \CcoolRead \tab $\Omega$ $\Field$ $\Meas$
%
%
                                                \Omega \mathcal{F} \mathcal{P}
```

```
Listing 8. Mittelwertsatz für n Variable[3, 17.3]
%
      \CcoolOption{ Write = \BooleanTrue }
%
      \newtheorem{theorem}{Theorem}
      \AfterEndEnvironment{theorem}{\CcoolHook}
%
      \Ccool i{\mathbb{#1}}
%
%
      \{ N = \{ N \} , R = \{ R \} \} + []
%
      { Grad = { \operatorname{grad} } }+
%
      [\begin{theorem}
         [Mittelwertsatz f\"ur $n$ Variable]Es~sei~]
         { OffMenge = {D}, Ci = {C^{1}}, Strecke = { [x_0,x] } }+
%
         [$n\in\mathbb{N}, -$\setminus OffMenge\setminus Subseteq\setminus\mathbb{N}^n$ eine offene Menge und
    f\in Ci(\Omega_{R}, R).
%
        Dann gibt es auf jeder Strecke $\Strecke\subset\OffMenge$ einen
    Punkt $\xi\in\Strecke$,~]
%
         { Steig = { \f(x)-f(x_0) \} \{ x-x_0 \} \}, Punkt = { \xi \} \}+
         [so dass gilt
%
%
         \begin{equation*}
%
           \Steig = \Grad f(\Punkt)^{\perp}
%
         \end{equation*}
%
      \end{theorem}]
%
%
      (Check: N\, \Punkt$)
%
      \CcoolOption{ Write = \BooleanFalse }
%
Theorem 1 (Mittelwertsatz für n Variable) Es sei n \in \mathbb{N}, D \subseteq \mathbb{N}^n eine
offene Menge und f \in C^1(D,\mathbb{R}). Dann gibt es auf jeder Strecke [x_0,x] \subset D einen
```

Punkt $\xi \in [x_0, x]$, so dass gilt

$$\frac{f(x) - f(x_0)}{x - x_0} = \operatorname{grad} f(\xi)^{\top}$$

(Check: \mathbb{N}, ξ)

```
Listing 9. Listing 8 read from file.
```

```
%
```

 $\mathbb{N} \ \mathbb{R} \ D \ C^1 \ [x_0, x]$

```
Listing 10. Lambda expression.

% \CcoolOption{ \( \text{Write} = \text{BooleanTrue} \) }
% \Ccool{ \( \text{EvalAt} = \text{CcoolLambda}((#1)) \), \( ApplyOp = \text{CcoolLambda}[2] \{#1[#2]} \) }
% \( [Supposons une fonction \( \frac{\text{FevalAt}{t}}\), \( et \'etudions le probl\'eme \) o\'u la fonctionnelle \( \frac{\text{ApplyOp}{S}{f}}\) \( est \ donn\'ee \ par\dots] \{ \}
% \( \text{CcoolOption}{\text{Write} = \text{BooleanFalse}} \) \( \text{Supposons une fonction } f(t), \) et \( \text{étudions le problème où la fonctionnelle } S[f] \) est \( \text{donn\( \text{e} par. . . } \)
```

Listing 12. CUSUM statistic[5]

```
%
                \newtheorem{definition}{Definition}
%
                \AfterEndEnvironment{definition}{\CcoolHook}
%
                \CcoolOption{ Write = \BooleanTrue }
%
                \Ccool{ SuchThat = { ;~ }, Time = { t }, Process = { \xi }, StopT = { \x
            \{T\}, EvalAt = \CcoolLambda\{(\#1)\}
%
                [The CUSUM statistic process and the corresponding one-sided CUSUM
           stopping time are defined as follows:
%
                \begin{definition}\label{the CUSUM statistic}. Let~]
                     { Scale = { \lambda }, Real = {\mathcal{R}} }+*s{{~\in~}}[~and~]
%
%
                     { CUSUMthresh = { \langle u \rangle } +*o{$\#1\leq ^{+}$.}
%
                     [~Define the following processes:]
%
                     { LogWald = { u }, CUSUMst = { \StopT_{c} }, CUSUM = { y },
           LogWaldInf = { m } }+
%
                      [\begin{enumerate}
%
                      \int {\int {Scale } = Scale \Pr (Scale)} 
            - \frac{1}{2}\Scale^2\Time$;
%
                          \Lambda = \prod_{N \in \mathbb{N}} EvalAt\{ Scale \} = \inf_{N \in \mathbb{N}} O \le N \in \mathbb{N}
           }\CUSUM_{s} \EvalAt{ \Scale }$.}
%
                     \left( \frac{s\CUSUM_{\Omega}}{Time}\right) =
           \label{logWaldInf_{\Time}\EvalAt{ \Scale } - \LogWald_{\Time}\EvalAt{ }
            \Scale }\geO$, which is the CUSUM statistic process.}
                     \left( \frac{s\c Scale}{\c Scale} \right) = \inf\left( \frac{Time}{\c Scale} \right)
%
            \ge 0 \SuchThat \CUSUM_{\Time}\EvalAt{\Scale} \ge \LogWaldInf
            \right]$, which is the CUSUM stopping time.}
            \end{enumerate}\end{definition}\par]{}
%
%
                (Check: $\Scale$, $\CUSUM$)
%
                \CcoolOption{ Write = \BooleanFalse }
```

%

The CUSUM statistic process and the corresponding one-sided CUSUM stopping time are defined as follows:

Definition 1 . Let $\lambda \in \mathcal{R}$ and $\nu \in \mathcal{R}^+$. Define the following processes:

- 1. $u_t(\lambda) = \lambda \xi_t \frac{1}{2}\lambda^2 t$; $m_t(\lambda) = \inf_{0 \le s \le t} y_s(\lambda)$.
- 2. $y_t(\lambda) = m_t(\lambda) u_t(\lambda) \ge 0$, which is the CUSUM statistic process.
- 3. $T_c(\lambda, m) = \inf[t \ge 0; y_t(\lambda) \ge m]$, which is the CUSUM stopping time.

(Check: λ, y)

Listing 13. Listing 12 read from file.

% \CcoolRead \tab \$\Time \$ \$\Process\$ \$\Scale\$ \$\Real\$ \$\CUSUMthresh\$ \$\LogWald\$ \$\CUSUMst\$ \$\CUSUM\$ \$\LogWaldInf\$

%

 $\bar{t} \xi \bar{\lambda} \bar{\mathcal{R}} \nu u \bar{T}_c y m$

Part III

Other

1 Acknowledgment

This work has benefited from Q&A's from the LATEX community[6]. Specific attributions are made in the implementation and References.

2 Install

Compiling ccool.dtx² will generate ccool.sty and ccool.pdf

3 Issue

```
1. Don't: Inner={\{####1\}
    Symptom: \CcoolRead fails
    Do: Inner={\char'{####1\char'}}
```

4 Support

This package is available from https://www.ctan.org/pkg/ccool and https://github.com/rogard/ccool.

5 Testing

5.1 Technicality

Not possible to compile-check the expansion of a certain class of macros against predefined values[8]. Instead, one can visually check Part II, as generated in section 2 on one's own machine, against that of the repository for the same version.

5.2 Platform

```
1. Linux laptop 4.15.0-20-generic #21-Ubuntu SMP Tue Apr 24

→ 06:16:15 UTC 2018 x86_64 x86_64 x86_64 GNU/Linux
```

5.3 Engine

1. pdfTeX 3.14159265-2.6-1.40.20 (TeX Live 2019)

5.4 Results

1. ccool v1.8 satisfactory on platform 1 and engine 1

²Under Unix, \$tex ccool.dtx

5.5 Other

Check [5] for using coool with lines

References

- [1] Nick Setzer The cool package, 2005, https://www.ctan.org/pkg/cool
- [2] The LATEX3 Project Team *The LATEX3 interfaces*, 2019, http://ftp.math.purdue.edu/mirrors/ctan.org/macros/latex/contrib/l3kernel/interface3.pdf
- [3] Thomas F. Sturm *The tcolorbox package*, 2019, http://www.texdoc.net/texmf-dist/doc/latex/tcolorbox/tcolorbox.pdf
- [4] The LATEX3 Project Team *The xparse package*, 2020, http://ftp.math.purdue.edu/mirrors/ctan.org/macros/latex/contrib/l3packages/xparse.pdf
- [5] Erwann Rogard and Olympia Hadjiliadis *Typesetting a math thesis with ccool*, 2020, https://github.com/rogard/ccool/blob/master/thesis.pdf
- [6] User erwann at tex.stackexchange.com, https://tex.stackexchange.com/users/ 112708/erwann?tab=questions
- [7] @sean-allred's answer to "How to create lambda expressions?", https://tex.stackexchange.com/a/188053/112708
- [8] "Checking a function's expansion against a string", https://tex.stackexchange.com/a/534100
- [9] @frougon's answer to "Journaling calls to a function []", https://tex.stackexchange.com/a/536620

Change History

v1.0	Replaced: GenericObject by Name	14
General: Initial version 14	Replaced: Separators by Separ	14
v1.1	Revamped: much of the	
General: Added: Save	implementation	14
Added: Listing 1., 2., 3., 4., 6., and	v1.2	
9	General: Deleted: \OopsTest	14
Added:\OopsRestore 14	Deleted: $\langle kvl_2 \rangle$ and $\langle code_2 \rangle$	14
Added:\OopsTest 14	Deleted: Listing 2-3 from v1.1	14
Deleted: Listing 1-5 from v1.0 14	Replaced: $\{\langle tl_1 \rangle\}$ by	
Fixed: apparent anomaly in v1.0's	$\label{list} $$ \OopsClear[\langle keyval\ list \rangle] \ \dots \ .$	14
Listing 4, see Listing 3 14	Replaced: \Restore by \Read	14
Replaced:	Replaced: \Save by \Write	14
\OopsOptions by \OopsOption 14	v1.3	
Replaced:	General: Replaced: \OopsNew by \Oops	14
$\{\langle kvl_2\rangle\}$ by $\langle kvl_2\rangle$ given that	Replaced: $\{\langle tl_1 \rangle\}$ and $[\langle tl_1 \rangle]$ by	
option type G not recommended[4] 14	$\langle \langle tl_1 \rangle \rangle$	14

v1.4	Renamed: \OopsHook to	
General: Added: section 9	14 \CcoolHook 1	4
Added: \OopsDebug		
Added: \OopsHook	•	.4
Added: Expans (for debugging'	Renamed: \OopsRead to	
sake, but)	(CCCCINCAA	١4
Added: Listing 1., 2., and 3		14
Deleted: Listing 1., and 2	Renamed: oops to ccool (better	
Replaced: $s\{\{\langle tl_3\rangle\}\{\langle tl_4\rangle\}\{\langle tl_5\rangle\}\}$	describes the purpose) 1	14
by $\mathbf{s}\{\{\langle tl_3\rangle\} \{\langle tl_3\rangle\}\{\langle tl_4\rangle\} \{\langle tl_3\rangle\}\{\langle tl_4\rangle\}\}$	V(+1-)V1.7	
\$\\\u3/5\\\u3/5\\\u4/5\\\\u3/5\\\u4/5\\\\\		4
v1.5	Added: Listing 12 (CUSUM) 1	
General: Added: File	_ ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `	
	14	
Deleted: dependence on datetime .	Deleted: Listing 5 from v1.6 1	۱4
v1.6	v1.8	
General: Added: Listing 2 (preamble)	General: Added: \CcoolLambda 1	14
Renamed: \OopsClear to	A 1.1. 1. \ O 1 V 1	
\CcoolClear	14	
Renamed: \OopsDebug to	Added: Listing 10, Listing 11 1	
\CcoolDebug	14 Added: Listing 1 1	١4

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

${f Symbols}$	${f A}$
* (option)	\AtEndDocument 94
+ (option)	_
$\langle key_i \rangle$	В
$\langle code_1 \rangle$ (option)	\begin 296
$\langle code_2 \rangle$ (option)	\begingroup 179
$\langle kvl_1 \rangle$ (option)	bool commands:
$\langle t1_1 \rangle$ (option)	\bool_gset_false:N 99
$\langle t1_2 \rangle$ (option)	\bool_gset_true:N 106
	\bool_if:nTF 121, 175, 199, 359
$\langle t1_3 \rangle$ (option)	\bool_set_false:N 95
$\langle t1_4 \rangle$ (option)	\BooleanFalse 364, 365
$\langle tl_5 \rangle$ (option)	
$\langle \mathtt{tl}_6 \rangle$ (option)	C
	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209
$\langle \mathtt{tl}_6 \rangle$ (option)	\Ccool 1, 4, 4, 5, 5, 5, 5, 27, 180, 188, 209 ccool internal commands:
$\langle t1_6 \rangle$ (option)	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219
\(\tau t \tau_6 \) (option) 5 Expans (option) 5 File (option) 6	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165
$\langle t1_6 \rangle$ (option) 5 Expans (option) 5 File (option) 6 Inner (option) 6	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165 \ccool_aux_key:N 17, 169
$\begin{array}{llllllllllllllllllllllllllllllllllll$	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165 \ccool_aux_key:N 17, 169 \ccool_aux_key:n 13, 20
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165 \ccool_aux_key:N 17, 169 \ccool_aux_key:n
$\begin{array}{llllllllllllllllllllllllllllllllllll$	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165 \ccool_aux_key:N 17, 169 \ccool_aux_key:n 13, 20 \ccool_aux_key:w 9, 15 \gccool_aux_key_seq
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165 \ccool_aux_key:N 17, 169 \ccool_aux_key:n 13, 20 \ccool_aux_key:w 9, 15 \gccool_aux_key_seq 11, 19, 150, 170, 272, 273
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	\Ccool 1, 4, 4, 5, 5, 5, 27, 180, 188, 209 ccool internal commands: \ccool_aux_inner:n 6, 7, 38, 219 \ccool_aux_inner_set:n 4, 165 \ccool_aux_key:N 17, 169 \ccool_aux_key:n 13, 20 \ccool_aux_key:w 9, 15 \gccool_aux_key_seq

\ccool_aux_outer_set:n $\underline{22}$, 151	\ccool_prop_item:nn 135, <u>279</u>
\gccool_aux_prop 26, 29, 46, 168	\ccool_prop_name:n
\ccool_aux_prop:N <u>44</u> , 167	83, 264, 268, 277, 281, 283, 286
\ccool_aux_prop:n <u>40</u> , 50	\ccool_prop_new:n 164, <u>284</u>
\ccool_aux_prop:nn <u>26</u>	\ccool_seq_from_prop:n 290, 294
\ccool_aux_prop:w <u>32</u> , 42	\ccool_seq_from_prop:NNn $83, \underline{288}$
$\c \c \$	$\c \sum_{\text{ccool_seq_use:Nn}} 155, 300$
$\cline{1.5}$ _ccool_aux_separ:nn $\underline{53}$, 78	\CcoolClear 1, 5, 27, <u>381</u>
$\c \sum_{\text{ccool}} \text{aux_val}: \text{Nn} \dots \dots \underbrace{80}, 150$	\CcoolDebug
\gccool_aux_val_seq 82, 83, 156	\CcoolHook
$\c \sum_{\text{ccool_erw_seq_use:Nn}} 296$	\CcoolLambda 2, 5, 27, <u>386</u>
$\cline{1.5}$ \ccool_lambda:nn 85, 388	\CcoolOption 2, 5, 27, 390
$\c \c \$	\CcoolRead
\ccool_log_close: <u>93</u> , 361	\CcoolVers
\ccool_log_entry 180, 181	cs commands:
\gccool_log_file_tl . 101, 104, 313	\cs_generate_variant:Nn
\gccool_log_iow	\ldots 7, 31, 75, 113, 128, 138, 143, 261
$\dots \dots 93, 94, 98, 105, 123, 126$	\cs_gset:Npn 6, 24, 247
\ccool_log_open: <u>101</u> , <u>360</u>	\cs_new:Nn 53, 76, 227, 275, 279, 300
\g_ccool_log_open_bool	\cs_new:Npn 283
0.00000000000000000000000000000000000	\cs_new_protected:Nn 4, 13, 17,
\ccool_log_read: <u>114</u> , 400	22, 27, 40, 44, 80, 96, 102, 108, 114,
\ccool_log_read:n <u>108</u> , <u>116</u> , <u>399</u>	119, 129, 139, 144, 148, 160, 172,
\g_ccool_log_to_tl 104, 105, 116, 118	219, 237, 245, 262, 266, 270, 284, 288
\ccool_log_write:n <u>118</u> , 177	\cs_new_protected:Npn
\ccool_make_ccool:nnnn	
185, 319, 333, 347, 369	\cs_set:Nn 255
\ ccool make ccool exp:nnn 148, 197	
_ccool_make_ccool_exp:nnn	\cs_set_protected:Nn 290
\cdots _ccool_make_ccool_key:nnn $\underline{160}$, 174	
_ccool_make_ccool_key:nnn 160, 174 _ccool_make_ccool_sideeffect:nnn	\cs_set_protected:\n\ \cs_set_D
\ccool_make_ccool_key:nnn	$\begin{tabular}{lllllllllllllllllllllllllllllllllll$
_ccool_make_ccool_key:nnn	\cs_set_protected:\n\ \cs_set_D
_ccool_make_ccool_key:nnn	$\begin{tabular}{lllllllllllllllllllllllllllllllllll$
_ccool_make_ccool_key:nnn	\cs_set_protected:\n
_ccool_make_ccool_key:nnn	\cs_set_protected:\n
_ccool_make_ccool_key:nnn	\cs_set_protected:\n
_ccool_make_ccool_key:nnn	\cs_set_protected:\n 290 D \DeclareDocumentCommand 88, 188 \def 180, 185 E \end 299 \endgroup 181 \ensuremath 356, 357
_ccool_make_ccool_key:nnn	\cs_set_protected:\n 290 D \DeclareDocumentCommand 88, 188 \def 180, 185 E \end 299 \endgroup 181 \ensuremath 356, 357 exp commands:
_ccool_make_ccool_key:nnn	\cs_set_protected:\n 290 D \DeclareDocumentCommand 88, 188 \def 180, 185 E \end 299 \endgroup 181 \ensuremath 356, 357 exp commands: \exp_args:\N\f 154
_ccool_make_ccool_key:nnn	\cs_set_protected:\n 290 D \DeclareDocumentCommand 88, 188 \def 180, 185 E \end 299 \endgroup 181 \ensuremath 356, 357 exp commands: \exp_args:\n\n 154 \exp_args:\n\n 87, 131, 188
_ccool_make_ccool_key:nnn	\cs_set_protected:\n 290 D \DeclareDocumentCommand 88, 188 \def 180, 185 E \end 299 \endgroup 181 \ensuremath 356, 357 \exp commands: \exp_args:\n\n\n 154 \exp_args:\n\n\n 87, 131, 188 \exp_args:\n\n 165, 268
_ccool_make_ccool_key:nnn	D \DeclareDocumentCommand
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} 88, 188 \tag{def} 180, 185 \tag{E} \end 299 \endgroup 181 \ensuremath 356, 357 \exp commands: \exp_args:NNf 154 \exp_args:NNx 87, 131, 188 \exp_args:NNx 87, 131, 188 \exp_args:Nx 35 \exp_args:Nx 35 \exp_last_unbraced:Nf 318, 332, 346, 368 \exp_last_unbraced:NNf 302 \exp_not:N 71, 225, 235, 243, 251
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} & 88, 188 \text{def} & 180, 185 \text{E} \text{end} & 299 \text{endgroup} & 181 \text{ensuremath} & 356, 357 \text{exp_args:NNf} & 154 \text{exp_args:NNx} & 87, 131, 188 \text{exp_args:NNx} & 87, 131, 188 \text{exp_args:Nx} & 35 \text{exp_last_unbraced:Nf} & 356, 368 \text{exp_last_unbraced:Nf} & 302 \text{exp_not:N} & 302 \text{exp_not:N} & 30313 \text{exp_not:N} & 209, 313 \text{exp_not:N} & 200, 313 exp_not
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} & 88, 188 \\def & 180, 185 \\ E \tag{end} & 299 \\end{group} & 181 \\ensuremath & 356, 357 \\exp_args:NNf & 154 \\exp_args:NNx & 87, 131, 188 \\exp_args:Nx & 35 \\exp_args:Nx & 35 \\exp_last_unbraced:Nf & 38, 332, 346, 368 \\exp_last_unbraced:Nf & 302 \\exp_not:N & 71, 225, 235, 243, 251 \\exp_not:N & 209, 313 \\expandafter & 180 \exp_andafter & 180 \exp_andafter \end{group} \tag{18}
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} & 88, 188 \\def & 180, 185 \\ E \end & 299 \\end & 299 \\end & 356, 357 \\exp_args:NNf & 154 \\exp_args:NNx & 87, 131, 188 \\exp_args:NNx & 87, 131, 188 \\exp_args:NN & 35 \\exp_args:NN & 368 \\exp_args:NN & 302 \\exp_a
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} & 88, 188 \\def & 180, 185 \\ E \tag{end} & 299 \\end{group} & 181 \\ensuremath & 356, 357 \\exp_args:NNf & 154 \\exp_args:NNx & 87, 131, 188 \\exp_args:Nx & 35 \\exp_args:Nx & 35 \\exp_last_unbraced:Nf & 38, 332, 346, 368 \\exp_last_unbraced:Nf & 302 \\exp_not:N & 71, 225, 235, 243, 251 \\exp_not:N & 209, 313 \\expandafter & 180 \exp_andafter & 180 \exp_andafter \end{group} \tag{18}
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} 88, 188 \tag{def} 180, 185 \tag{E} \end 299 \tag{endgroup} 181 \tag{ensuremath} 356, 357 \tag{exp_args:NNf} 154 \tag{exp_args:NNx} 87, 131, 188 \tag{exp_args:NNx} 87, 131, 188 \tag{exp_args:Nx} 35 \tag{exp_last_unbraced:Nf} 358, 368 \tag{exp_last_unbraced:Nnf} 302 \tag{exp_not:N} 71, 225, 235, 243, 251 \tag{exp_not:n} 209, 313 \tag{expandafter} 180 \tag{explSyntaxOff} 405 \tag{exp}
_ccool_make_ccool_key:nnn	D \text{DeclareDocumentCommand}
_ccool_make_ccool_key:nnn	D \tag{DeclareDocumentCommand} 88, 188 \tag{def} 180, 185 \tag{E} \end 299 \tag{endgroup} 181 \tag{ensuremath} 356, 357 \tag{exp_args:NNf} 154 \tag{exp_args:NNx} 87, 131, 188 \tag{exp_args:NNx} 87, 131, 188 \tag{exp_args:Nx} 35 \tag{exp_last_unbraced:Nf} 358, 368 \tag{exp_last_unbraced:Nnf} 302 \tag{exp_not:N} 71, 225, 235, 243, 251 \tag{exp_not:n} 209, 313 \tag{expandafter} 180 \tag{explSyntaxOff} 405 \tag{exp}

G	⟨tl ₅ ⟩ 5
\gappto 201	$\langle \mathtt{t1}_6 \rangle$
	Expans 5
I	File 6
\IfBooleanT 195	Inner 6
\IfValueT	Name 6
\IfValueTF 398 int commands:	Outer 6
\int_case:nnTF	Separ 6
iow commands:	Write 6
\iow_close:N 94, 98	_
\iow_new:N 93	P
\iow_now:Nn 123	\pdfcreationdate 314, 315
$\verb \iow_open:Nn 105 $	prg commands:
\item 297, 298	\prg_replicate:nn 58, 89
V	prop commands:
K keys commands:	\prop_clear_new:N
\lambda_keys_choice_tl 310	\prop_gclear_new:N
\keys_define:nn 306	\prop_if_exist:NTF 277
\keys_set:nn 393	\prop_item:\n
· • •	\prop_map_function:NN 259
${f M}$	\prop_new:N
\meta 297, 298	\ProvideDocumentCommand 132, 386
msg commands:	
\msg_error:nnn 126, 233	${f Q}$
\msg_error:nnnn	quark commands:
\msg_new:nnn 213, 214, 215, 216, 217, 218	\q_stop 9, 15, 33, 42
	• 1— • • 1
\msg_warning:nnn 225, 243, 251	1_ 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
\msg_warning:mm 223, 243, 231	${f R}$
${f N}$ \NeedsTeXFormat	R \Read 27
N	${f R}$
${f N}$ \NeedsTeXFormat	R \Read 27
$\begin{tabular}{lllllllllllllllllllllllllllllllllll$	R \Read
N \NeedsTeXFormat	R \Read
N \NeedsTeXFormat 2 \NewDocumentCommand 381, 390, 395, 402 O \Oops 27 \OopsClear 27 \OopsDebug 27 \OopsHook 27 \OopsNew 27 \OopsOpsOption 27 \OopsOption 27 \OopsOpsRead 27 \OopsRestore 27	R \Read
N \NeedsTeXFormat	R \Read
N \NeedsTeXFormat	R \Read
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	R \Read

${f U}$	\use_ii:nn 62
use commands:	\usepackage
\use:N	\mathbf{W}
\use i:nn 61,63	\Write 27

Part IV

1 (@@=ccool)

Implementation

```
2 \NeedsTeXFormat{LaTeX2e}[2019/10/01]
                              3 \ExplSyntaxOn
                            1
                                  aux
\__ccool_aux_inner_set:n #1: \langle code \rangle
                              4 \cs_new_protected:Nn \__ccool_aux_inner_set:n
                              5 {
                                  \cs_gset:Npn \__ccool_aux_inner:n ##1 {#1}
                                  \cs_generate_variant:Nn \__ccool_aux_inner:n { e }
                            (End definition for \__ccool_aux_inner_set:n.)
      \__ccool_aux_key:w #1: \langle key \rangle
                            #2 : ⟨ value ⟩
                              9 \cs_new_protected:Npn \__ccool_aux_key:w #1 = #2 \q_stop
                                  \seq_gput_right:Nx \g__ccool_aux_key_seq { \tl_trim_spaces:n{#1} }
                             12 }
                            (End\ definition\ for\ \_\_ccool\_aux\_key:w.)
      \__ccool_aux_key:n #1: \langle key = value \rangle
                             13 \cs_new_protected:Nn \__ccool_aux_key:n
                                  (End definition for \__ccool_aux_key:n.)
      \__ccool_aux_key:N #1: \langle seq \rangle
                             17 \cs_new_protected:Nn \__ccool_aux_key:N
                                  \verb|\seq_gclear_new:N \ \g_ccool_aux_key_seq| \\
                                  \seq_map_function:NN #1 \__ccool_aux_key:n
                             20
                            (End definition for \__ccool_aux_key:N.)
\__ccool_aux_outer_set:n #1: \langle inline code \rangle
                             22 \cs_new_protected:Nn \__ccool_aux_outer_set:n
                                  \cs_gset:Npn \__ccool_aux_outer:n ##1 {#1}
                             25 }
                            (End\ definition\ for\ \verb|\_\_ccool\_aux\_outer\_set:n.|)
```

```
\__ccool_aux_prop:nn
                            _{26} \prop_new:N \g_ccool_aux\_prop
                            _{\rm 27} \cs_new_protected:Nn \__ccool_aux_prop:nn
                                 \prop_gput:Nnn \g__ccool_aux_prop{#1}{#2}
                            29
                            30 }
                            _{\mbox{\scriptsize 31}} \cs_generate_variant:\n \__ccool_aux_prop:nn { eo, ee, ex, xo, xe, xx }
                           (End\ definition\ for\ \verb|\_\_ccool\_aux\_prop:nn.)
  \__ccool_aux_prop:w #1: \langle key \rangle
                           #2: \langle value \rangle
                            32 \tl_new:N \g__ccool_option_expans_tl
                            33 \cs_new_protected:Npn \__ccool_aux_prop:w #1 = #2 \q_stop
                            34 {
                                 \exp_args:Nx
                            35
                                \use:c{__ccool_aux_prop:\g__ccool_option_expans_tl}
                                 { \tl_trim_spaces:n{#1} }
                                 { \__ccool_aux_inner:n{ \tl_trim_spaces:n{#2} } }
                            38
                            39 }
                           (End\ definition\ for\ \verb|\_\_ccool\_aux\_prop:w.|)
  \__ccool_aux_prop:n #1: \langle key = value \rangle
                           40 \cs_new_protected:Nn \__ccool_aux_prop:n
                                 \__ccool_aux_prop:w #1 \q_stop
                           43 }
                           (End definition for \__ccool_aux_prop:n.)
  \__ccool_aux_prop:N #1: \langle keyval \ list \rangle
                            44 \cs_new_protected:Nn \__ccool_aux_prop:N
                                 \prop_gclear_new:N \g__ccool_aux_prop
                                \seq_if_empty:NTF #1
                                 { \c_empty_tl }
                            49
                                   \seq_map_function:NN #1 \__ccool_aux_prop:n
                            50
                                 }
                           51
                           52 }
                           (End\ definition\ for\ \verb|\_\_ccool\_aux\_prop:N.)
\__ccool_aux_separ:nn #1: \langle int \rangle
                           #2: \langle tokens \rangle
                            53 \cs_new:Nn \__ccool_aux_separ:nn
                                 \int_case:nnTF {#1}
                            56
                                 {
                                   {1}
                            57
                                   { \prg_replicate:nn{ 3 }{#2} }
                                   {2}
                            59
                                   {
                            60
```

```
{ \use_i:nn #2 }
                         61
                                  { \use_ii:nn #2 }
                         62
                                  { \use_i:nn #2 }
                         63
                         64
                                {3}{#2}
                         65
                             }
                         66
                              { \c_empty_tl }
                         67
                         68
                                \msg_error:nnnn { __erw }
                                { separ }
                                { \exp_not:N \__ccool_aux_separ:nn }
                         73
                         74 }
                         75 \cs_generate_variant:Nn \__ccool_aux_separ:nn { e }
                        (End definition for \__ccool_aux_separ:nn.)
\__ccool_aux_separ:n #1: \langle tokens \rangle
                         76 \cs_new:Nn \__ccool_aux_separ:n
                              \__ccool_aux_separ:en{ \tl_count:n{#1} }{#1}
                         79 }
                        (End\ definition\ for\ \verb|\__ccool_aux_separ:n.|)
 \__ccool_aux_val:Nn #1: \langle seq \rangle
                        #2: \langle tl \ var \ name \rangle
                         80 \cs_new_protected:Nn \__ccool_aux_val:Nn
                         81 {
                              \seq_gclear_new:N \g__ccool_aux_val_seq
                              \label{lem:cool_seq_from_prop:NNn } $$ \ccool_aux_val_seq #1 { \ccool_prop_name:n{#2} } $$
                         83
                         84 }
                        (End\ definition\ for\ \verb|\__ccool_aux_val:Nn.|)
                        2
                              lambda
  \__ccool_lambda:nn
                       [7]
                         85 \cs_new_protected:Npn \__ccool_lambda:nn #1 #2
                         86 {
                              \exp_args:NNx
                              \DeclareDocumentCommand \__ccool_lambda_expression
                              { \prg_replicate:nn { #1 } { m } }
                              {#2}
                              91
                         92 }
                        (End\ definition\ for\ \_\_ccool\_lambda:nn.)
```

$3 \log$

```
\__ccool_log_close:
                       93 \iow_new:N \g__ccool_log_iow
                       94 \AtEndDocument{\iow_close:N \g__ccool_log_iow}
                       95 \bool_set_false:N \g__ccool_log_open_bool
                       96 \cs_new_protected:Nn \__ccool_log_close:
                       97 {
                            \in \g_ccool_log_iow
                            \bool_gset_false:N \g__ccool_log_open_bool
                       99
                       100 }
                      (End definition for \__ccool_log_close:.)
  \__ccool_log_open:
                       101 \tl_new:N \g__ccool_log_file_tl
                       102 \cs_new_protected:Nn \__ccool_log_open:
                       103 -
                            \tl_gset:Nx \g_ccool_log_to_tl{\g_ccool_log_file_tl}
                       104
                            \iow_open:Nn \g__ccool_log_iow {\g__ccool_log_to_tl}
                       105
                            \bool_gset_true:N \g__ccool_log_open_bool
                       106
                      (End\ definition\ for\ \_\_ccool\_log\_open:.)
 \__ccool_log_read:n #1: \langle path \rangle
                       \label{loss_new_protected:Nn } $$ \cs_new_protected:Nn \\ -\ccool_log_read:n $$
                       109 {
                            \file_input:n{#1}
                            \tl_log:n{read~from~#1}
                       112 }
                       \cs_generate_variant:\n \__ccool_log_read:n { e }
                       (End\ definition\ for\ \_\_ccool\_log\_read:n.)
  \__ccool_log_read:
                       114 \cs_new_protected:Nn \__ccool_log_read:
                       115
                            \__ccool_log_read:e{\g__ccool_log_to_tl}
                       116
                       117 }
                      (End\ definition\ for\ \verb|\_\_ccool\_log\_read:.)
\__ccool_log_write:n
                       119 \cs_new_protected:Nn \__ccool_log_write:n
                       120 {
                            \bool_if:nTF{ \g__ccool_log_open_bool }
                       122
                              \iow_now:Nn \g__ccool_log_iow {#1}
                       123
                              \tl_log:n{ write~to~#1 }
                       124
                            127 }
                       128 \cs_generate_variant:Nn \__ccool_log_write:n { e }
                      (End definition for \__ccool_log_write:n.)
```

4 make_key

```
\__ccool_make_key:Nn #1: \langle token \rangle
                                #2: \langle key \rangle
                                129 \cs_new_protected:Nn \__ccool_make_key:Nn
                                130 {
                                      \exp_args:NNx
                                131
                                     \ProvideDocumentCommand{#1}
                                132
                                     { D<>{\g_ccool_option_name_tl} }
                                        135
                                     }
                                136
                                137 }
                                138 \cs_generate_variant:Nn \__ccool_make_key:Nn {c}
                                (End definition for \__ccool_make_key:Nn.)
         \__ccool_make_key:n #1: \langle key \rangle
                                139 \cs_new_protected:Nn \__ccool_make_key:n
                                      \cline{1}{make_key:cn{#1}{#1}}
                                142 }
                                143 \cs_generate_variant:Nn \__ccool_make_key:n { e }
                                (End\ definition\ for\ \verb|\__ccool_make_key:n.|)
         \__ccool_make_key:N #1: \langle seq \rangle
                                144 \cs_new_protected:Nn \__ccool_make_key:N
                                145
                                      \seq_map_function:NN #1 \__ccool_make_key:e
                                147 }
                                (End\ definition\ for\ \_\_ccool\_make\_key:N.)
                                5
                                     make_ccool
\__ccool_make_ccool_exp:nnn
                                148 \cs_new_protected: Nn \__ccool_make_ccool_exp:nnn
                                149 {
                                      \__ccool_aux_val:Nn \g__ccool_aux_key_seq {#1}
                                150
                                      \__ccool_aux_outer_set:n{#3}
                                151
                                152
                                      \__ccool_aux_outer:n
                                153
                                        \exp_args:NNf
                                154
                                        \__ccool_seq_use:Nn
                                155
                                        \g__ccool_aux_val_seq
                                156
                                        {#2}
                                157
                                     }
                                158
                                159 }
                                (End\ definition\ for\ \_\_ccool\_make\_ccool\_exp:nnn.)
```

```
\__ccool_make_ccool_key:nnn
                                                                         160 \cs_new_protected:Nn \__ccool_make_ccool_key:nnn
                                                                         161 {
                                                                                      \__ccool_prop_if_exist:nTF{#1}
                                                                         162
                                                                                     { \c_empty_tl }
                                                                         163
                                                                                     { \__ccool_prop_new:n{#1} }
                                                                          164
                                                                                     \exp_args:No \__ccool_aux_inner_set:n{#2}
                                                                          165
                                                                                     \seq_set_from_clist:Nn \g__ccool_aux_keyval_seq {#3}
                                                                          166
                                                                                      \__ccool_aux_prop:N \g__ccool_aux_keyval_seq
                                                                                      \__ccool_prop_append:Nn \g__ccool_aux_prop {#1}
                                                                                      \__ccool_aux_key:N \g__ccool_aux_keyval_seq
                                                                                      \__ccool_make_key:N \g__ccool_aux_key_seq
                                                                          170
                                                                         171 }
                                                                         (End definition for \__ccool_make_ccool_key:nnn.)
                                                                        [9]
           \__ccool_make_ccool_sideeffect:nnn
                                                                         173 {
                                                                                      \cline{1}{make_ccool_key:nnn{#1}{#2}{#3}}
                                                                         174
                                                                                      \bool_if:nTF{ \g__ccool_log_open_bool }
                                                                         175
                                                                                      {%^A https://tex.stackexchange.com/questions/536597
                                                                         176
                                                                                          \__ccool_log_write:n
                                                                          177
                                                                                          {
                                                                          178
                                                                                                \begingroup
                                                                                                \def \__ccool_log_entry { \Ccool<#1>i{#2}{#3} } \expandafter
                                                                                                \endgroup \__ccool_log_entry
                                                                          181
                                                                                     }{\c_empty_tl}
                                                                         183
                                                                         184 }
                                                                        (End\ definition\ for\ \_\_ccool\_make\_ccool\_sideeffect:nnn.)
                                                                        #1: \langle token \ list \rangle
            _ccool_make_ccool:nnnn
                                                                        #2:
                                                                                    \langle seq_1 \rangle
                                                                        #3:
                                                                                     \langle seq_2 \rangle
                                                                         #4:
                                                                                     \langle prop \rangle
                                                                          185 \def\CcoolHook{\c_empty_tl}
                                                                         \mbox{\sc ls6} \cs_{new\_protected:Npn \cs_make\_ccool:nnnn #1 #2 #3 #4 }
                                                                         187 {
                                                                                      \exp_args:NNx \DeclareDocumentCommand \Ccool
                                                                                                                                                      4 5 6 7 8
                                                                                                            2
                                                                                                                        3
                                                                          189
                                                                                          D<>{#1} +o E{ i }{{#2}} m t+ s E{ s o }{{#3}{#4}} +o
                                                                          190
                                                                                     }
                                                                          191
                                                                          192
                                                                                          \IfValueT{##2}{##2}
                                                                          193
                                                                                          \__ccool_make_ccool_sideeffect:nnn{##1}{##3}{##4}
                                                                                          \IfBooleanT{##6}
                                                                                                \cline{1.8} \cli
                                                                          197
                                                                                          }
                                                                          198
                                                                                          \bool_if:nTF{##5}
                                                                          199
                                                                          200
                                                                                                \gappto{\CcoolHook}
                                                                          201
```

```
\label{lem:lemmake_ccool_side} $$ \sum_{mn{\#1}{\#3}{\#4}} $$
                                                                        203
                                                                        204
                                                                                            }
                                                                        205
                                                                                             {\c_empty_tl}
                                                                        206
                                                                                             \IfValueT{##9}
                                                                        207
                                                                        208
                                                                                                   \exp_not:n{ \Ccool<##1>[##9] }
                                                                                             }
                                                                                      }
                                                                        211
                                                                        212 }
                                                                       (End definition for \__ccool_make_ccool:nnnn.)
                                                                       6
                                                                                      msg
                                                                        \label{local_local} $$\max_{213} \msg_new:nnn {__ccool}{ generic }{\#1}$
                                                                        214 \msg_new:nnn {__ccool}{ iow }{#1~is~closed~can't~write}
                                                                        \label{locality} $$\max_{new:nnn {\_ccool}{ keyonly }{\#1$-does-not-take-values;-keyval-is-\#2}} $$
                                                                        \label{local} $$ \msg_new:nnn {\cline{local}{keywrong}}{$\dots$} $$ \msg_new:nnn {\cline{local}{keywrong}} $$
                                                                        \label{local_local} $$\min_{n=1,\dots,n} {\cool}{\separ }{\#1\sim\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\cool}{\c
                                                                        218 \msg_new:nnn {__ccool}{ unset }{#1~unset}
                                                                        7
                                                                                       option
      \__ccool_aux_inner:n #1: \langle code \rangle
                                                                        220 {
                                                                                       \tl_gset:Nn \g__ccool_option_inner_tl {#1}
                                                                        221
                                                                        222 }
                                                                        223 \__ccool_option_inner:n
                                                                                       \msg_warning:nnn{ __ccool }{ unset }{ \exp_not:N \g__ccool_option_inner_tl }
                                                                        225
                                                                       (End\ definition\ for\ \verb|\_\_ccool\_aux\_inner:n.|)
\__ccool_option_name:n #1: \langle token \ list \rangle
                                                                        227 \cs_new:Nn \__ccool_option_name:n
                                                                                       \tl_gset:Nn \g__ccool_option_name_tl{#1}
                                                                        229
                                                                        230 }
                                                                        231
                                                                                \__ccool_option_name:n
                                                                        232 {
                                                                                      \msg_error:nnx{ __ccool }
                                                                                      { generic }
                                                                                       { \exp_not:N\g__ccool_option_name_tl~undefined }
                                                                        235
                                                                        236 }
                                                                        (End\ definition\ for\ \_\_ccool\_option\_name:n.)
```

```
\__ccool_option_outer:n #1: \langle inline \ code \ \rangle
                              237 \cs_new_protected:Nn \__ccool_option_outer:n
                                    \tl_gset:Nn \g__ccool_option_outer_tl {#1}
                               240 }
                               241 \__ccool_option_outer:n
                               242 {
                                    \msg_warning:nnn{ __ccool }{ unset }{ \exp_not:N \g__ccool_option_outer_tl }
                               243
                               244 }
                              (End\ definition\ for\ \verb|\_\_ccool\_option\_outer:n.|)
  \__ccool_option_separ:n #1: \{\langle tl_1 \rangle\}\{\langle tl_2 \rangle\}\{\langle tl_3 \rangle\}
                               245 \cs_new_protected:Nn \__ccool_option_separ:n
                                    \cs_gset:Npn \g__ccool_option_separ_tl {#1}
                               247
                               248 }
                               249
                                  \__ccool_option_separ:n
                               250 {
                                    \msg_warning:nnn{ __ccool }{ unset }{ \exp_not:N \g__ccool_option_separ_tl }
                               251
                              (End definition for \__ccool_option_separ:n.)
                                    prop
                              #1: \langle prop_1 \rangle
  \__ccool_prop_append:NN
                              #2: \langle prop_2 \rangle
                               ^{253} \cs_new_protected:Npn \__ccool_prop_append:NN #1 #2
                                    \cs_set:Nn \__ccool_prop_append:nn
                                      \prop_gput:Nnx #1 {##1}{ \prop_item:Nn #2{##1} }
                               257
                               258
                                    \prop_map_function:NN #2 \__ccool_prop_append:nn
                               259
                               260 }
                               261 \cs_generate_variant:Nn \__ccool_prop_append:NN { cN }
                              (End\ definition\ for\ \_\_ccool\_prop\_append:NN.)
  \__ccool_prop_append:Nn #1: \langle prop \rangle
                              #2: \langle tl \ var \ name \rangle
                               \__ccool_prop_append:cN{ \__ccool_prop_name:n {#2} } #1
                              (End\ definition\ for\ \verb|\_\_ccool\_prop\_append:Nn.)
\__ccool_prop_clear_new:n #1: \langle tl var name \rangle
                              266 \cs_new_protected:Nn \__ccool_prop_clear_new:n
                                    \exp_args:No \prop_clear_new:c{ \__ccool_prop_name:n {#1} }
                               268
                               269 }
```

```
(End\ definition\ for\ \verb|\__ccool_prop_clear_new:n.|)
        \ ccool prop clear new map:n #1: \langle keyval list \rangle
                                  270 \cs_new_protected:Nn \__ccool_prop_clear_new_map:n
                                        \seq_set_from_clist:Nn \g__ccool_aux_key_seq {#1}
                                        \seq_map_function:NN \g__ccool_aux_key_seq \__ccool_prop_clear_new:n
                                  273
                                  274 }
                                  (End definition for \__ccool_prop_clear_new_map:n.)
\__ccool_prop_if_exist:nTF #1: \langle tl_1 \rangle
                                  #2: \langle tl_2 \rangle
                                  #3 : \langle tl_3 \rangle
                                  275 \cs_new:Nn \__ccool_prop_if_exist:nTF
                                        \prop_if_exist:cTF{ \__ccool_prop_name:n {#1} }{#2}{#3}
                                  277
                                  278 }
                                  (End\ definition\ for\ \verb|\__ccool_prop_if_exist:nTF|.)
      \__ccool_prop_item:nn #1: \langle tl \ var \ name \rangle
                                  #2: \langle key \rangle
                                  279 \cs_new:Nn \__ccool_prop_item:nn
                                        \prop_item:cn { \__ccool_prop_name:n {#1} } {#2}
                                  282 }
                                  (End definition for \__ccool_prop_item:nn.)
       \__ccool_prop_name:n #1: \langle tl var name \rangle
                                  283 \cs_new:Npn \__ccool_prop_name:n #1{ __ccool_#1 }
                                  (End definition for \__ccool_prop_name:n.)
        \__ccool_prop_new:n #1: \langle tl var name \rangle
                                  \verb| \cs_new_protected:Nn \cs_new_prop_new:n | \\
                                        \prop_new:c{ \__ccool_prop_name:n {#1} }
                                  286
                                  (End\ definition\ for\ \_\_ccool\_prop\_new:n.)
```

```
9
                                       seq
                                #1: \langle seq_1 \rangle
\__ccool_seq_from_prop:NNn
                                #2: \langle seq_2 \rangle (keys)
                                \#3: \langle prop \rangle
                                 288 \cs_new_protected:Nn \__ccool_seq_from_prop:NNn
                                 289 {
                                       \cs_set_protected: Nn \__ccool_seq_from_prop:n
                                 290
                                 291
                                         \seq_gput_right:No #1 { \prop_item:cn{#3}{##1} }
                                 292
                                 293
                                       \seq_map_function:NN #2 \__ccool_seq_from_prop:n
                                 295 }
                                (End\ definition\ for\ \verb|\__ccool_seq_from_prop:NNn.|)
   \__ccool_erw_seq_use:Nn
                                 296 %
                                             \begin{arguments}
                                 297 %
                                             \item \meta{ seq }
                                             \item \meta{ tokens }
                                 298 %
                                            \verb|\end{arguments}|
                                 299 %
                                 300 \cs_new:Nn \__ccool_seq_use:Nn
                                 301 {
                                       \exp_last_unbraced:NNf
                                 302
                                       \seq_use:Nnnn #1
                                 303
                                       \__ccool_aux_separ:n{#2}
                                 305 }
```

10 Front-end

 $(End\ definition\ for\ \verb|__ccool_erw_seq_use:Nn.|)$

```
306 \keys_define:nn { __ccool }
307 {
    Expans .multichoices:nn =
308
     { eo, ee, ex, xo, xe, xx }
309
     { \t_{gset_eq:NN \g_ccool_option_expans_tl \l_keys_choice_tl },
310
    Expans .default:n = { xo },
311
     Expans .initial:n = \{ xo \},
312
313
     File .code:n = { \tl_gset:Nn \g__ccool_log_file_tl{ \exp_not:n{ #1 } } },
    File .default:n = { ccool\pdfcreationdate },
314
    File .initial:n = { ccool\pdfcreationdate },
    Name .code:n={
       \verb|\__ccool_option_name:n{#1}|
       \exp_last_unbraced:Nf
318
       \__ccool_make_ccool:nnnn
319
       {
320
         { \g_ccool_option_name_tl }
321
         { \g_ccool_option_inner_tl }
322
         { \g_ccool_option_separ_tl }
323
324
         { \g_ccool_option_outer_tl }
325
       }
    },
```

```
Name .value_required:n = false,
     Name .default:n = { Math },
328
     Name .initial:n = { Math },
329
     Inner .code:n={
330
        \__ccool_option_inner:n{#1}
331
        \exp_last_unbraced:Nf
332
        \__ccool_make_ccool:nnnn
333
334
          { \g__ccool_option_name_tl }
335
          { \g_ccool_option_inner_tl }
336
          { \g_ccool_option_separ_tl }
337
          { \g_ccool_option_outer_tl }
338
339
     },
340
     Inner .value_required:n = false,
341
     Inner .default:n = \{\#\#\#1\},
342
     Inner .initial:n = {\#\#\#1},
343
     Outer .code:n={
        \__ccool_option_outer:n{#1}
        \exp_last_unbraced:Nf
347
        \__ccool_make_ccool:nnnn
        {
348
          { \g_{\text{cool}}\position_{\text{name}}\position_{\position} }
349
          { \g_ccool_option_inner_tl }
350
          { \g_ccool_option_separ_tl }
351
352
            \g__ccool_option_outer_tl }
       }
353
     },
354
     Outer .value_required:n = false,
355
     Outer .default:n = { \ensuremath{####1} },
     Outer .initial:n = { \ensuremath{####1} },
357
     Write .code:n = {
        \bool_if:nTF{#1}
350
        {\__ccool_log_open:}
360
        {\__ccool_log_close:}
361
     },
362
     Write .value_required:n = false,
363
     Write .default:n = \BooleanFalse,
     Write .initial:n = \BooleanFalse,
     Separ .code:n={
        \__ccool_option_separ:n{#1}
        \exp_last_unbraced:Nf
369
        \__ccool_make_ccool:nnnn
370
          { \g_ccool_option_name_tl }
371
          { \g_ccool_option_inner_tl }
372
          { \g_ccool_option_separ_tl }
373
            \g__ccool_option_outer_tl }
374
       }
375
376
     },
377
     Separ .value_required:n = false,
     Separ .default:n = \{ \{ \setminus \} and \{ \setminus \} \} \{ ,\{ \setminus \} \} \{ ,\{ \setminus \} and \{ \setminus \} \},
     Separ .initial:n = { \{\ \}\ and\{\ \}\ \}\ \{\ ,\{\ \}\ \}\ \{\ ,\{\ \}\ and\{\ \}\ \}
379
380 }
```

```
\cline{CcoolClear} #1: \langle tl \ var \ name \rangle
               381 \NewDocumentCommand{ \CcoolClear }
               382 { D<>{\g__ccool_option_name_tl} }
                     \__ccool_prop_clear_new_map:n{#1}
               385 }
               (End definition for \CcoolClear. This function is documented on page 5.)
\CcoolLambda
                386 \ProvideDocumentCommand \CcoolLambda { O{1} m }
                     \__ccool_lambda:nn { #1 } { #2 }
               389 }
               (End definition for \CcoolLambda. This function is documented on page 5.)
\CcoolOption
               390 \NewDocumentCommand{ \CcoolOption }
               391 { m }
               392 {
                     \keys_set:nn{ __ccool }{#1}
               (End definition for \CoolOption. This function is documented on page 5.)
  \CcoolRead
               395 \NewDocumentCommand{\CcoolRead}
               396 {o}
               397 {
                     \IfValueTF{#1}
               399 {\__ccool_log_read:e{#1}}
                     {\__ccool_log_read:}
               401 }
               (End definition for \CcoolRead. This function is documented on page 6.)
  \CcoolVers
               402 \NewDocumentCommand{\CcoolVers}
               403 {}
               404 {\use:c{ver@ccool.sty}}
               (End definition for \CcoolVers. This function is documented on page 6.)
                       Misc
               11
```

405 \ExplSyntaxOff