## The ccool package\*

## Erwann Rogard<sup>†</sup>

## Released 2020/04/17

#### Abstract

ccool is, at its core, a key-value interface built upon xparse[4], for generating commands, with a view to reduce the overhead<sup>1</sup>. Specifically,  $\cool{keyval list}$  creates for each i the command  $\cool{key}_i$ , with  $\cool{val}_i$  as its implementation. For instance, the side effect of { Real =  $\mbox{mathbb{R}}$ } is that  $\mbox{Real}$  expands to  $\mbox{R}$ . A  $\mbox{keyval list}$  can optionally be expanded in place (append \*), according to default or inline rules, and interspersed with optional arguments that are destined exclusively for expansion ([Let~] and [~denote real numbers.]). Thus, in theory, a document could be typeset, starting with a single  $\cool{lecool}$  declaration. An optional parameter prepended to the  $\mbox{keyval list}$ ,  $\cool{list}$ , allows to parameterize the keys (for instance, one parameter value for the style, another for a property). In conjunction with lamba expressions, the commands created through this interface can accept arguments the same way as  $\mbox{NewDocumentCommand}$ . A possible use of this, in Math, could be to encode the way functions and operators are typeset, (say (.) and [.] around their arguments). Optionally, the macros can be written to a file, and read, which can be useful for typesetting documents sharing the same notation.

## Contents

Ι	Usage
0	Convention
1	Loading the package
2	\Ccool
	$2.1  \llbracket \langle t l_1  angle  bracket  bracket$
	$2.2 < \langle tl_2 \rangle > \dots$
	$2.3$ i $\{\langle code_1 \rangle\}$
	$2.4  \{\langle kvl_1 \rangle\}$
	2.5 +
	2.6 *
	2.7 $s\{\{\langle tl_3\rangle\} \{\langle tl_4\rangle\} \{\langle tl_4\rangle\} \{\langle tl_4\rangle\}\{\langle tl_4\rangle\}\}\}$
	$2.8  o\{\langle code_2\rangle\}$
	$2.9  \lceil \langle tl_e \rangle \rceil$

<sup>\*</sup>This file describes version v2.2, last revised 2020/04/17.

<sup>†</sup>firstname dot lastname AusTria gmail dot com

<sup>&</sup>lt;sup>1</sup>See section 2 for the initial motivation

3 Other	5
4 Do's and dont's	6
II Listing	7
1. Preamble	7
2.\CcoolVers	7
3. Let $\mathbb N$ and $\mathbb R$ Plain	7
4. Let $\mathbb N$ and $\mathbb R$ Equivalent to 3 with $\mathbb N$ ewDocumentCommand	7
5. Let $\mathbb N$ and $\mathbb R$ Equivalent to 4 with $\backslash \mathbb C$ cool	7
6. Let $\mathbb N$ and $\mathbb R$ Equivalent to 5 with $\mathtt{i}\{\langle code_1 \rangle\}$	8
7. Let $\mathbb N$ and $\mathbb R$ Equivalent to 6 with $\mathfrak{s}\{\langle tl_3\rangle\}$	8
8. Let $\mathbb N$ and $\mathbb R$ Extends 7 with descriptive keys, with $\langle tl_2 \rangle >$	8
9. Let $\mathbb N$ and $\mathbb R$ Equivalent to 8 with $\mathfrak{s}\{\langle tl_3\rangle\}$	8
10. Separators.	8
11. Hello, world! (test case)	9
12. Listing 11 read from file	9
13. Probability space.	9
14. Listing 13 read from file	10
15. Mittelwertsatz für $n$ Variable.	10
16. Listing 15 read from file	11
17. Fonction et fonctionelle	11
18. Listing 17 read from file	11
19. CUSUM statistic.	11
20. Listing 19 read from file	12
III Other	13
1 Acknowledgment	13
2 Genealogy	13

3	Install	13
4	Issue	13
5	Support	13
6 Refe	Testing 6.1 Technicality 6.2 Platform 6.3 Engine 6.4 Results 6.5 Other	13 13 14 14 14 14 14
Cha	ange History	15
Ind	ex	16
IV	Implementation	19
1	Opening	19
2	aux	19
3	lambda	21
4	log	22
5	make_key	23
6	make_ccool	23
7	msg	25
8	option	<b>25</b>
9	prop	26
10	seq	27
11	sys	28

12	Front-end	29
	12.1 \CcoolClear	29
	12.2 \CcoolHook	29
	12.3 \CcoolLambda	29
	12.4 \CcoolOption	29
	12.4.1 Expans	30
	12.4.2 File	30
	12.4.3 Inner	30
	12.4.4 Param	30
	12.4.5 Outer	31
	12.4.6 Separ	31
	12.4.7 Write	31
	12.5 \CcoolRead	32
	12.6 \CcoolVers	32
13	Closing	32

## 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 5.
- 2. Declare it in the *preamble*

```
 \begin{array}{lll} & \label{eq:cool} & \label{eq:cool} & \\ \hline & [\langle t1_1 \rangle] & \\ & & \langle \langle t1_2 \rangle \rangle & \\ & & & i \{\langle code_1 \rangle\} & \\ & & & \{\langle kv1_1 \rangle\} & \\ & & & \\ & & & \\ & & s \{\{\langle t1_3 \rangle\} | \{\langle t1_3 \rangle\} \{\langle t1_4 \rangle\} | \{\langle t1_3 \rangle\} \{\langle t1_4 \rangle\} \{\langle t1_5 \rangle\}\} & \\ & & & o \{\langle code_2 \rangle\} & \\ & & [\langle t1_6 \rangle] & \end{array}
```

**Requirement**  $\langle kvl_1 \rangle$  is specified (all others optional).

 $\langle \mathtt{tl}_1 \rangle$ 

```
Example Let~
            Semantics Expands \langle tl_1 \rangle
   \langle tl_2 \rangle
            Default Param's
            Example Default, Style and Describe, ModelA and ModelB
\langle code_1 \rangle
            Default Inner's
            Example \mathbb{#1}
 \langle \mathtt{kvl}_1 \rangle
            Example Real={\mathbb{R}}}
            Semantics
                      1) \langle val_i \rangle \leftarrow \langle code_1 \rangle applied to \langle val_i \rangle
                      2) \langle key_i \rangle \langle (tl_2) \rangle \leftarrow \langle val_i \rangle defined in step 1), using Expans for expansion.
                      3) If Write, writes the input used by step 2) to File
            Semantics Repeats step 1), step 2), and step 3), at \CcoolHook (useful inside a local
                    group)
            Semantics 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, and \langle tl_5 \rangle.
   \langle t1_3 \rangle
            Default Separ's
            Example {~\in~}
   \langle \mathtt{tl}_4 
angle
            Default Separ's
            Example {,~}
   \langle tl_5 \rangle
            Default Separ's
            Example {~\&~}
\langle code_2 \rangle
            Default Outer's
            Example $\left\{#1\right\}$
   \langle {\tt tl}_6 \rangle
            Semantics \cool[\langle tl_6 \rangle]
```

## Other

Continued in Part IV, section 12.

## Do's and dont's

```
1)
Don't: \Ccool{ A = a, B = b }[Hello, world!]
   Do: \Ccool{ A = a, B = b }[Hello, world!]{}
       or \C ool{ A = a, B = b } Hello, world!
2)
Don't: \langle key_i \rangle < x.
   Do: \langle key_i \rangle \{<\} x
3)
Don't: [a, b)
   Do: {[}a, b{)}
Don't: \cal F.
   Do: \c \{F\} or \m \{F\}
5)
Don't: \[x_0,x\]
   Do: \left[x_0,x\right]
6) Also see Part III, section 4
```

## Part II

# Listing

Listing 1 are settings to replicate the listings. For exhaustivity, check the documentation section of ccool.dtx.

Listing 2 is self explanatory.

Listing 3-9 is a tutorial comprising different ways to typeset "Let  $\mathbb{N}$  and  $\mathbb{R}$ ..." The plain version is prone to errors, should the author change  $\mathbf{R}$  to, say,  $\mathbf{A}$  throughout the document. Listing 4 avoids that by separating style from meaning. Listing 5-7 achieve greater compactness (DRY and expand in place). Listing 8 extends Listing 5 with a description for each key. Listing 9 is to Listing 8, what Listing 7 is Listing 6.

Listing 10 shows the full range of uses of separators' parameter.

Listing 11 and Listing 12 are a contrived "Hello, world!" test case.

Listing 13, Listing 15, and Listing 19 typeset realistic mathematical text, and write it to a file. Listing 14, Listing 16, and Listing 20, read the corresponding files.

## Listing 1. Preamble

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

#### Listing 2. \CcoolVers

\CcoolVers

2020/04/17 v2.2 cool — A key-value interface for generating commands

## Listing 3. Let $\mathbb{N}$ and $\mathbb{R}$ ...Plain

Let  $\mathbb{N}$  and  $\mathbb{R}$  denote the natural and real numbers.

## Listing 4. Let $\mathbb N$ and $\mathbb R$ ... Equivalent to 3 with $\mathbb N$ ewDocumentCommand

 $\verb|\NewDocumentCommand\Nat{}{\mathbb{N}}|$ 

Let~\$\Nat\$ and \$\Real\$ denote the natural and real numbers.

Let  $\mathbb{N}$  and  $\mathbb{R}$  denote the natural and real numbers.

## Listing 5. Let $\mathbb N$ and $\mathbb R$ ... Equivalent to 4 with \Ccool

 Let  $\mathbb N$  and  $\mathbb R$  denote the natural and real numbers.

```
Listing 6. Let \mathbb{N} and \mathbb{R}... Equivalent to 5 but with i\{\langle code_1\rangle\}
\label{eq:cool} \begin{tabular}{l} $$ \cool i{\mathbb{R} athbb{#1}} { Nat = {N}, Real = {R} } $$ Let $$ \and $\cool i {\norm{Real} $$ -denote the natural and real numbers.} $$ Let <math>\mathbb{N} and \mathbb{R} denote the natural and real numbers.
```

```
Listing 7. Let \mathbb{N} and \mathbb{R}... Equivalent to 6 with s\{\langle tl_3 \rangle\}
\label{eq:ccool} $$ \{ \mathbb{R} \} *s\{ - \mathbb{R} \} *s\{ - \mathbb{R} \} *s\{ - \mathbb{R} \} *s \{ \mathbb{R}
```

```
Listing 8. Let \mathbb{N} and \mathbb{R}... Equivalent to 7 with <\!\langle tl_2 \rangle >
 \begin{tabular}{l} $$ \Ccool{ Nat = {\mathbb{N}}, Real = {\mathbb{R}} } [] \\ \club{Cescribe} & \Nat = {\text{natural numbers}}, Real = {\text{the real line}} & \Club{Equivalent} & \Club{Equiv
```

```
Listing 9. Let N and R...Equivalent to 8 with s{\langle tl_3 \rangle}

\[ \Ccool[Let^] \\ i\mathbb{\{\finit}\} \ \ \text{Nat} = \{\text{N}\}, \text{Real} = \{\text{R}\} \\ \*s\{\{\chince\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\fini\{\fini\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\finit\{\fin
```

```
{ X = x, Y = y, Z = z }*s{{\ \&\ }{,\ }{\ \&\ }}\\

x @ y
x % y @ z
x & y
x & y
x & y
x & y
x & y & z
x, y & z
x, y & z
x, y & z
```

```
Listing 11. Hello, world! (test case)
 \CcoolOption{Separ = {{}}{.}{.}}, Outer = {####1}}
 \CcoolOption{ Write = \BooleanTrue }
\Ccool
 <Test>{ KeyA = \{.\}, KeyB = \{!\}, KeyC = \{\\%\} }[]
 <Test>{ KeyD = {d}, KeyE = \{\\%\} }[]
 <Test>{ KeyI = \{\%\}, KeyJ = \{\%\}, KeyK = \{\%\} }[.\{1\}.\{o\}]
 <Test>{ KeyL = {1}, KeyM = {\char`[}, KeyN = {\char`]} }[]
 <Test>{ KeyO = \{0\}, KeyP = \{\\%\}, KeyQ = \{\\%\} \{[\{,\\}]
 <Test>{ KeyR = \{w\}, KeyS = \{o\}, KeyT = \{r\} }*
 s{{}{}}o{{\char`[}#1}[]
 <Test>{ KeyU = \{\\%\}, KeyV = \{\\%\}, KeyW = \{\\%\} }[]
 \label{eq:continuous} $$\operatorname{KeyX} = {\%}, KeyY = {\%}, KeyZ = {\KeyB<Test>} }\nobreak
 \KeyL<Test>\KeyD<Test>\KeyZ<Test>\KeyN<Test>\\
 \CcoolOption{ Write = \BooleanFalse }
\{H\}.\{e\}.\{l\}.\{o\}, [world!]
```

```
CcoolRead
\KeyF<Test>\KeyA<Test>\nobreak
\KeyG<Test>\KeyA<Test>\nobreak
\KeyH<Test>\KeyA<Test>\nobreak
\KeyH<Test>\KeyA<Test>\nobreak
\KeyH<Test>\KeyA<Test>\nobreak
\{\{\}\nobreak\Key0<Test>\{\}\},\{\}\nobreak
\KeyM<Test>\keyR<Test>\nobreak
\KeyU<Test>\nobreak
\KeyU<Test>\nobreak
\KeyU<Test>\nobreak
\KeyT<Test>\nobreak
\KeyD<Test>\nobreak
\KeyD<Test>\nobreak
\KeyD<Test>\nobreak
\KeyD<Test>\nobreak
\KeyD<Test>\nobreak
\KeyD<Test>\nobreak
\KeyN<Test>\nobreak
\KeyN<Test>\nobreak
\KeyN<Test>\nobreak
\KeyN<Test>\nobreak
\KeyN<Test>\nobreak
```

## Listing 14. Listing 13 read from file

\CcoolRead \tab \$\Omega\$ \$\Field\$ \$\Meas\$

 $\Omega \mathcal{F} \mathcal{P}$ 

```
Listing 15. 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~]
%
      %
      [$n\in\mathbb{N}, -$\setminus OffMenge\setminus Subseteq\setminus\mathbb{N}^n$ eine offene Menge und
    f\in Ci(\Omega_{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)^{\times}
%
      \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:  $\mathbb{N}, \xi$ )

### Listing 16. Listing 15 read from file

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

### Listing 17. Fonction et fonctionelle

```
\CcoolOption{ Write = \BooleanTrue }
\Ccool{ EvalAt = \CcoolLambda{(#1)}, ApplyOp = \CcoolLambda[mm]{#1[#2]} }
[Supposons une fonction $f\EvalAt{t}$, et \'etudions le probl\`eme o\`u
  la fonctionnelle $\ApplyOp{S}{f}$ est donn\'ee par\dots]{}
\CcoolOption{ Write = \BooleanFalse }
```

Supposons une fonction f(t), et étudions le problème où la fonctionnelle S[f] est donnée par...

## Listing 18. Listing 17 read from file

```
% \CcoolRead \tab $f\EvalAt{t}$, $\Apply0p{S}{f}$ %
```

 $f(\bar{t}), \bar{S}[\bar{f}]$ 

## Listing 19. CUSUM statistic[5]

```
\newtheorem{definition}{Definition}
\AfterEndEnvironment{definition}{\CcoolHook}
\CcoolOption{ Write = \BooleanTrue }
\Ccool{ SuchThat = { ;~ }, Time = { t }, Process = { <math>\xi }, StopT = { 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{{\sim n^{-}}[~and~]}
      { CUSUMthresh = { \nu } }+*o{$#1\in\Real^{+}$.}
      [~Define the following processes:]
      { LogWald = { u }, CUSUMst = { \StopT_{c} }, CUSUM = { y },
      LogWaldInf = { m } }+
      [\begin{enumerate}
      \label{logWald_{Time}_EvalAt{ \Scale } = \Scale\\Process_{Time} - \Constant{ \Scale } = \Constant{ \Scale} - \Con
      \frac{1}{2}\Scale^2\Time$;
                  \Lambda _{\Time}\ \Scale \ = \inf_{ 0\le s \le \Time}
      }\CUSUM_{s} \EvalAt{ \Scale }$.}
      \item{$\CUSUM_{\Time}\EvalAt{ \Scale } = \LogWaldInf_{\Time}\EvalAt{
      Scale } - LogWald_{Time}\EvalAt{ Scale }\ge0$, which is the CUSUM
      statistic process.}
```

\item{\$\CUSUMst \EvalAt{ \Scale, \LogWaldInf } = \inf\left[ \Time \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:  $\lambda, y$ )

## Listing 20. Listing 19 read from file

%

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

## Part III

## Other

## 1 Acknowledgment

This work has benefited from Q&A's from the LATEX community[6][10]. Specific attributions are made throughout this document.

## 2 Genealogy

In v2.0, the abstract began with

ccool stands for Custom COntent Oriented for LATEX, that is "give commands the ability to contain the mathematical meaning while retaining the typesetting versatility", a concept pioneered by cool[1]

But through forum interactions[10], the author realized it was confusing and the scope of ccool is in fact more generic.

## 3 Install

- 1) Compile ccool.dtx (under Unix, \$tex ccool.dtx)
- 2) Put the generated ccool.sty in the search path of the LATeX engine

### 4 Issue

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

## 5 Support

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

## 6 Testing

## 6.1 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 3 on one's own machine, against that of the repository for the same version.

#### 6.2 Platform

i) Linux laptop 4.15.0-20-generic #21-Ubuntu SMP Tue Apr 24  $\rightarrow$  06:16:15 UTC 2018 x86\_64 x86\_64 x86\_64 GNU/Linux

### 6.3 Engine

- a) pdfTeX 3.14159265-2.6-1.40.20 (TeX Live 2019)
- b) pdfTeX 3.14159265-2.6-1.40.21 (TeX Live 2020)
- c) LuaHBTeX, Version 1.12.0 (TeX Live 2020)
- d) XeTeX 3.14159265-2.6-0.999992 (TeX Live 2020)

#### 6.4 Results

- 1) ccool v1.8 satisfactory on platform i) and engine a)
- 2) ccool v1.8 satisfactory on platform i) and engine b)
- 3) ccool v1.9 satisfactory on platform i) and engines b) and c)
- 4) ccool v2.0 satisfactory on platform i) and engines b), c), and d)
- 5) ccool v2.1 satisfactory on platform i) and engines b), c), and d)

#### 6.5 Other

Check [5] for testing cool 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/13packages/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] 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] @joseph-wright's answer to "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
- [10] \Ccool, extension à IATEXà vocation mathématique, http://forum.mathematex.net/latex-f6/ccool-extension-latex-a-vocation-mathematique-t17314.html#p158966

# Change History

v1.0	Added: \OopsHook 15	
General: Initial version 15		
v1.1	sake, but)	
General: Added: Save 15	Added: Listing 1., 2., and 3 15	
Added: Listing 1., 2., 3., 4., 6., and	Added:optional +to \OopsNew to	
9		
Added:\OopsRestore 15	local group	
Added:\OopsTest 15	Deleted: Listing 1., and 2 15	
Deleted: Listing 1-5 from v1.0 15	Replaced: $s\{\{\langle tl_3\rangle\}\{\langle tl_4\rangle\}\{\langle tl_5\rangle\}\}$	
Fixed: apparent anomaly in v1.0's	by	
Listing 4, see Listing 11 15	$\mathfrak{s}\{\{\langle tl_3 \rangle\} \{\langle tl_3 \rangle\}\{\langle tl_4 \rangle\} \{\langle tl_3 \rangle\}\{\langle tl_4 \rangle\}\{\langle tl_5 \rangle\}\}$	}
Replaced:		
\OopsOptions by \OopsOption 15	v1.5	
Replaced:	General: Added: File 15	
$\{\langle kvl_2\rangle\}$ by $\langle kvl_2\rangle$ given that	Deleted: dependence on datetime $15$	
option type G not recommended[4] 15	v1.6	
Replaced: GenericObject by Name 15	General: Added: Listing 1 (preamble) 15	
Replaced: Separators by Separ 15	Renamed: \OopsClear to	
Revamped: much of the	\CcoolClear 15	
implementation $\dots \dots 15$	Renamed: \OopsDebug to	
v1.2	\CcoolDebug 15	
General: Added: optional *to	Renamed: \OopsHook to	
\OopsNew as instruction to expand	\CcoolHook 15	
$kvl_1$	Renamed: \OopsOption to	
Deleted: \OopsTest 15	\CcoolOption $\dots \dots 15$	
Deleted: $\langle kvl_2 \rangle$ and $\langle code_2 \rangle$ 15	Renamed: \OopsRead to	
Deleted: Listing 2-3 from v1.1 15	\CcoolRead	
Replaced: $\texttt{OopsClear}\{\langle tl_2\rangle\}$ by	Renamed: \Oops to \Ccool 15	
$\label{loss_clear} $$ \log Clear[\langle keyval \ list \rangle]  \ldots  15$$	Renamed: oops to cool (better	
Replaced: \Restore by \Read 15	describes the purpose) $\dots \dots 15$	
Replaced: \Save by \Write 15	v1.7	
v1.3	General: Added: Legends to listings $.$ 15	
General: Replaced: \OopsNew by \Oops 15	Added: Listing 19 (CUSUM) 15	
Replaced: $\{\langle tl_2 \rangle\}$ and $[\langle tl_2 \rangle]$ by	Deleted: \CcoolDebug 15	
$\langle \langle tl_2 \rangle \rangle$	Deleted: Listing 5 from v1.6 $\dots$ 15	
v1.4	v1.8	
General: Added: section 4 15	General: Added: \CcoolVers 15	
Added: \OopsDebug 15	Added: \CcoolLambda 15	

Added: Listing 17, Listing 18 15	Listing 8, and Listing 9 (tutorial) . 15
Added: Listing 2	Replaced: \CcoolLambda's optional
v1.9	integer argument (number of m's)
General: Added: support for LuaTeX 15	by a standard argument list 15
Moved: from Part I to Part IV,	Replaced: $\langle tl_2 \rangle$ 's position within
what is now that part's section 12 15	\Ccool's argument list, from first
v2.0	to second. Greater versatility $\dots$ 15
General: Added: support for $X_{\overline{1}}T_{\overline{1}}X$ . 15	Replaced: Name by Param 15
Deleted: File's dependency on	Replaced: as the de-
texosquery and $\pdfcreationdate$ 15	${ m fault\ of\ Param,\ Math\ by\ Default}$ . $15$
$\operatorname{Updated}$ : \RequirePackage,	v2.2
$\NeedsTeXFormat$ 's second	General: Removed: $\%$ from listings 15
argument / TeX Live $2020 \dots 15$	Replaced: part of the abstract's
v2.1	with more straighforward
General: Added: Listing 3, Listing 4,	descriptions based on input from
Listing 5, Listing 6, Listing 6,	forum participtants 15

# 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.

Symbols * (option)	\begingroup
+ (option)	\bool_gset_false:N 98
$\langle key_i \rangle$	\bool_gset_true:N 105
$\langle code_1 \rangle$ (option)	\bool_if:nTF 120, 174, 197, 414
$\langle code_2 \rangle$ (option)	\bool_set_false:N $\dots 94$
$\langle kvl_1 \rangle$ (option)	\BooleanFalse 419, 420
$\langle tl_1 \rangle$ (option)	
$\langle t1_2 \rangle$ (option)	C
$\langle t1_3 \rangle$ (option)	\Ccool 1, 2, 4, 5, 7, 15, 26, 26, 179, 186, 207 ccool internal commands:
$\langle t1_5 \rangle$ (option)	\ccool_aux_inner:n 5, 6, 37
$\langle t1_6 \rangle$ (option)	\ccool_aux_inner_set:n 3, 164
Expans (option)	\ccool_aux_key:N <u>16</u> , 168
File (option)	$\c \sum_{\text{ccool}} aux_{\text{key:n}} \dots \frac{12}{12}, 19$
Inner (option) 27	$\c \sum_{\text{ccool}} aux_{\text{key}} : w \dots \underbrace{8}, 14$
Outer (option) 28	\gccool_aux_key_seq
Param (option) 27	10, 18, 149, 169, 270, 271
Separ (option)	\gccool_aux_keyval_seq 165, 166, 168
Write (option)	\ccool_aux_outer:n 23, 151
	\ccool_aux_outer_set:n 21, 150
	\gccool_aux_prop 25, 28, 45, 167
\	\_ccool_aux_prop:N
A	\ccool_aux_prop:n
AfterEndEnvironment	\_ccool_aux_prop:w 31, 41
\AtEndDocument	\_ccool_aux_separ:n 75, 302
(Accidocument	\_ccool_aux_separ:n <u>52</u> , 77
В	\_ccool_aux_val:Nn
\begin 294	\gccool_aux_val_seq 81, 82, 155

$\c \sum_{\text{ccool\_erw\_seq\_use:Nn}} 294$	\ccool_sys_time: $\dots \dots 315, 324$
\ccool_lambda:nn <u>84</u> , 339	$\cdots$ _ccool_sys_time_hex: $\underline{323}$ , $\underline{329}$
$\cdots$ _ccool_lambda_expression 87, 90	\CcoolClear
\ccool_log_close: $\dots \dots \underline{92}, 416$	\CcoolHook
\ccool_log_entry 179, 180	\CcoolLambda
\gccool_log_file_tl . 100, 103, 353	\CcoolOption
\gccool_log_iow	\CcoolRead
$\dots \dots 92, 93, 97, 104, 122, 125$	\CcoolVers
\ccool_log_open: <u>100</u> , 415	cs commands:
\gccool_log_open_bool	\cs_generate_variant:Nn
94, 98, 105, 120, 174	$\dots$ 6, 30, 74, 112, 127, 137, 142, 259
\ccool_log_read: <u>113</u> , 427	\cs_gset:Npn 5, 23, 245
$\cdots$ _ccool_log_read:n $\underline{107}$ , $115$ , $426$	\cs_new:Nn $52, 75, 225,$
\gccool_log_to_tl 103, 104, 115, 117	273, 277, 298, 304, 313, 315, 323, 325
\ccool_log_write:n <u>117</u> , 176	\cs_new:Npn 281
\ccool_make_ccool:nnnn	$\c$ new_protected:Nn 3, 12, 16,
$$ $\underline{184}$ , 360, 374, 388, 402	21, 26, 39, 43, 79, 95, 101, 107, 113,
$\cdots$ _ccool_make_ccool_exp:nnn $\underline{147}$ , 195	118, 128, 138, 143, 147, 159, 171,
$\cdots$ _ccool_make_ccool_key:nnn $\underline{159}$ , $173$	217, 235, 243, 260, 264, 268, 282, 286
\ccool_make_ccool_sideeffect:nnn	\cs_new_protected:Npn
171, 192, 201	
\ccool_make_key:N <u>143</u> , 169	\cs_set:Nn 253
\ccool_make_key:n <u>138</u> , 145	\cs_set_protected:Nn 288
\ccool_make_key:Nn <u>128</u> , 140	D
\g_ccool_option_expans_tl 31, 35, 349	D
\ccool_option_inner:n $\underline{217}$ , 358	\DeclareDocumentCommand 87, 186
	\def 179
\gccool_option_inner_tl	
\gccool_option_inner_t1 219, 223, 363, 377, 391, 405	
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n 235, 386	${f E}$
219, 223, 363, 377, 391, 405 \ccool_option_outer:n 235, 386 \gccool_option_outer_tl	E \end 297
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n 235, 386	E \end
219, 223, 363, 377, 391, 405 \ccool_option_outer:n 235, 386 \gccool_option_outer_tl	E \end
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n 235, 386 \g_ccool_option_outer_tl 237, 241, 365, 379, 393, 407 \_ccool_option_param:n	E \end
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n	E \end
	E \end
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n 235, 386 \g_ccool_option_outer_tl	E \end 297 \endgroup 180 \ensuremath 397, 398 exp commands: \exp_args:NNf 153 \exp_args:NNx 86, 130, 186 \exp_args:No 164, 266
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n 235, 386 \g_ccool_option_outer_tl	E \end 297 \endgroup 180 \ensuremath 397, 398 exp commands: \exp_args:NNf 153 \exp_args:NNx 86, 130, 186 \exp_args:No 164, 266 \exp_args:Nx 34
	E \end
	E \end 297 \endgroup 180 \ensuremath 397, 398 exp commands: \exp_args:NNf 153 \exp_args:NNx 86, 130, 186 \exp_args:No 164, 266 \exp_args:Nx 34 \exp_last_unbraced:Nf 359, 373, 387, 401
	E \end
219, 223, 363, 377, 391, 405 \ccool_option_outer:n 235, 386 \gccool_option_outer_tl	E \end
	E \end
219, 223, 363, 377, 391, 405 \_ccool_option_outer:n 235, 386 \g_ccool_option_outer_tl 237, 241, 365, 379, 393, 407 \_ccool_option_param:n 225, 372 \g_ccool_option_param_tl 132, 227, 233, 332, 362, 376, 390, 404 \_ccool_option_separ:n 243, 400 \g_ccool_option_separ_tl 245, 249, 364, 378, 392, 406 \_ccool_prop_append:Nn 251, 262 \_ccool_prop_append:Nn 251, 262 \_ccool_prop_append:nn 253, 257 \_ccool_prop_clear_new:n 264, 271 \_ccool_prop_clear_new_map:n 268, 334	E \end
	E \end

\IfValueT 191, 205	P
\IfValueTF 425	prg commands:
int commands:	\prg_replicate:nn 57
\int_case:nnTF 54	prop commands:
\int_eval:n 306, 317	\prop_clear_new:N 266
\int_to_hex:n 314, 324	\prop_gclear_new:N 45
iow commands:	\prop_gput:Nnn 28, 255
$\iow_close: N \dots 93, 97$	\prop_if_exist:NTF 275
\iow_new:N 92	\prop_item:Nn 255, 279, 290
\iow_now:Nn 122	\prop_map_function:NN 257
$\verb \iow_open:Nn  \dots \dots$	\prop_new:N
\item 295, 296	\ProvideDocumentCommand 131, 337
K	
keys commands:	${f Q}$
\lambda_keys_choice_tl 349	quark commands:
\keys_define:nn 346	\q_stop 8, 14, 32, 41
\keys_set:nn 344	${f S}$
	seq commands:
M	\seq_gclear_new:N 18, 81
\meta 295, 296	\seq_gput_right:Nn 10, 290
msg commands:	\seq_if_empty:NTF 46
\msg_error:nnn 125, 231	\seq_map_function:NN
\msg_error:nnnn68	
\msg_new:nnn 211, 212, 213, 214, 215, 216	\seq_set_from_clist:Nn 165, 270
\msg_warning:nnn 223, 241, 249	\seq_use:Nnnn 301
N	sys commands:
\NewDocumentCommand	\c_sys_day_int 310
1, 2, 7, 331, 336, 341, 422, 429	\c_sys_hour_int 319
	\c_sys_jobname_str 327
O	\c_sys_minute_int 320
options:	\c_sys_month_int 309
* 5	\c_sys_year_int 308
+ 5	<u>_</u>
$\langle {\sf code}_1  angle$	${f T}$
$\langle code_2 \rangle$	tl commands:
$\langle \mathtt{kvl}_1 \rangle$	\c_empty_tl . 47, 66, 162, 182, 204, 336
$\langle tl_1 \rangle$	\tl_count:n
$\langle t1_2 \rangle$	\tl_gset:Nn 103, 219, 227, 237, 353
$\langle t1_3 \rangle$	\tl_gset_eq:NN 349
$\langle t1_4 \rangle$	\tl_log:n
$\langle t l_5 \rangle$	\tl_new:N
$\langle t1_6 \rangle$	\tl_trim_spaces:n 10, 36, 37
File	${f U}$
Inner	use commands:
Outer	\use:N
Param	\use_i:nn 60, 62
Separ	\use_ii:nn
Write 28	\usepackage //

## Part IV

# **Implementation**

## 1 Opening

1 (00=ccool)

```
2 \ExplSyntaxOn
                                   aux
\__ccool_aux_inner_set:n #1: \langle code \rangle
                               3 \cs_new_protected:Nn \__ccool_aux_inner_set:n
                                   \cs_gset:Npn \__ccool_aux_inner:n ##1 {#1}
                                   \cs_generate_variant:Nn \__ccool_aux_inner:n { e }
                             (End\ definition\ for\ \verb|\_\_ccool\_aux\_inner\_set:n.|)
      \__ccool_aux_key:w #1: \langle key \rangle
                               8 \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} }
                             (End definition for \__ccool_aux_key:w.)
      \__ccool_aux_key:n #1: \langle key = value \rangle
                              12 \cs_new_protected:Nn \__ccool_aux_key:n
                                    \__ccool_aux_key:w #1 \q_stop
                             (End\ definition\ for\ \verb|\_\_ccool\_aux\_key:n.|)
      \__ccool_aux_key:N #1: \langle seq \rangle
                              16 \cs_new_protected:Nn \__ccool_aux_key:N
                                   \seq_gclear_new:N \g__ccool_aux_key_seq
                                   \seq_map_function:NN #1 \__ccool_aux_key:n
                             (End definition for \__ccool_aux_key:N.)
\__ccool_aux_outer_set:n #1: \langle inline \ code \rangle
                              21 \cs_new_protected:Nn \__ccool_aux_outer_set:n
                                   \cs_gset:Npn \__ccool_aux_outer:n ##1 {#1}
                              24 }
                             (End\ definition\ for\ \_\_ccool\_aux\_outer\_set:n.)
```

```
\__ccool_aux_prop:nn
                           25 \prop_new:N \g__ccool_aux_prop
                            {\tt ^{26}\ \backslash cs\_new\_protected:Nn\ \backslash\_ccool\_aux\_prop:nn}
                                 \prop_gput:Nnn \g__ccool_aux_prop{#1}{#2}
                           28
                           29 }
                            30 \cs_generate_variant:Nn \__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
                            31 \tl_new:N \g__ccool_option_expans_tl
                            32 \cs_new_protected:Npn \__ccool_aux_prop:w #1 = #2 \q_stop
                            33 {
                                 \exp_args:Nx
                            34
                                \use:c{__ccool_aux_prop:\g__ccool_option_expans_tl}
                                 { \tl_trim_spaces:n{#1} }
                                 { \__ccool_aux_inner:n{ \tl_trim_spaces:n{#2} } }
                            37
                            38 }
                           (End\ definition\ for\ \verb|\_\_ccool\_aux\_prop:w.|)
  \__ccool_aux_prop:n #1: \langle key = value \rangle
                           39 \cs_new_protected:Nn \__ccool_aux_prop:n
                                 \__ccool_aux_prop:w #1 \q_stop
                           42 }
                           (End definition for \__ccool_aux_prop:n.)
  \__ccool_aux_prop:N #1: \langle keyval \ list \rangle
                            43 \cs_new_protected:Nn \__ccool_aux_prop:N
                                 \prop_gclear_new:N \g__ccool_aux_prop
                                \seq_if_empty:NTF #1
                                 { \c_empty_tl }
                            48
                                   \seq_map_function:NN #1 \__ccool_aux_prop:n
                            49
                                 }
                           50
                           51 }
                           (End\ definition\ for\ \verb|\_\_ccool\_aux\_prop:N.)
\__ccool_aux_separ:nn #1: \langle int \rangle
                           #2: \langle tokens \rangle
                            52 \cs_new:Nn \__ccool_aux_separ:nn
                                 \int_case:nnTF {#1}
                            55
                                 {
                                   {1}
                            56
                                   { \prg_replicate:nn{ 3 }{#2} }
                            57
                                   {2}
                            58
                                   {
                            59
```

```
{ \use_i:nn #2 }
                                  { \use_ii:nn #2 }
                         61
                                  { \use_i:nn #2 }
                         62
                         63
                                {3}{#2}
                         64
                             }
                         65
                              { \c_empty_tl }
                         66
                         67
                                \msg_error:nnnn { __ccool }
                                { separ }
                                { \exp_not:N \__ccool_aux_separ:nn }
                         72
                         73 }
                         74 \cs_generate_variant:Nn \__ccool_aux_separ:nn { e }
                        (End definition for \__ccool_aux_separ:nn.)
\__ccool_aux_separ:n #1: \langle tokens \rangle
                         75 \cs_new:Nn \__ccool_aux_separ:n
                              \__ccool_aux_separ:en{ \tl_count:n{#1} }{#1}
                         78 }
                        (End\ definition\ for\ \verb|\__ccool_aux_separ:n.|)
 \__ccool_aux_val:Nn #1: \langle seq \rangle
                        #2: \langle tl \ var \ name \rangle
                         79 \cs_new_protected:Nn \__ccool_aux_val:Nn
                         80 {
                              \seq_gclear_new:N \g__ccool_aux_val_seq
                         81
                              \label{lem:cool_seq_from_prop:NNn } $$ \ccool_aux_val_seq #1 { \ccool_prop_name:n{#2} } $$
                         82
                         83 }
                        (End\ definition\ for\ \verb|\__ccool_aux_val:Nn.|)
                        3
                              lambda
  \__ccool_lambda:nn
                       [7]
                         84 \cs_new_protected:Npn \__ccool_lambda:nn #1 #2
                         85 {
                              \exp_args:NNx
                              \DeclareDocumentCommand \__ccool_lambda_expression
                         87
                              {#2}
                              91 }
                        (End\ definition\ for\ \_\_ccool\_lambda:nn.)
```

## 4 log

```
\__ccool_log_close:
                                                         92 \iow_new:N \g__ccool_log_iow
                                                         93 \AtEndDocument{\iow_close:N \g__ccool_log_iow}
                                                          94 \bool_set_false:N \g__ccool_log_open_bool
                                                          95 \cs_new_protected:Nn \__ccool_log_close:
                                                                     \in \g_ccool_log_iow
                                                          97
                                                                     \bool_gset_false:N \g__ccool_log_open_bool
                                                          98
                                                          99 }
                                                       (End definition for \__ccool_log_close:.)
    \__ccool_log_open:
                                                        100 \tl_new:N \g__ccool_log_file_tl
                                                        101 \cs_new_protected:Nn \__ccool_log_open:
                                                        102
                                                                     \tl_gset:Nx \g_ccool_log_to_tl{\g_ccool_log_file_tl}
                                                        103
                                                                    \iow_open:Nn \g__ccool_log_iow {\g__ccool_log_to_tl}
                                                         104
                                                                    \bool_gset_true:N \g__ccool_log_open_bool
                                                        105
                                                        106 }
                                                       (End\ definition\ for\ \_\_ccool\_log\_open:.)
  \__ccool_log_read:n #1: \langle path \rangle
                                                        \label{loss_new_protected:Nn \label} $$ \cs_new_protected:Nn \label{loss_new_protected:new} $$ \cs_new_protected:Nn \label{loss_new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protected:new_protecte
                                                        108 {
                                                                     \file_input:n{#1}
                                                                    \tl_log:n{read~from~#1}
                                                        110
                                                        111 }
                                                        112 \cs_generate_variant:Nn \__ccool_log_read:n { e }
                                                        (End\ definition\ for\ \_\_ccool\_log\_read:n.)
    \__ccool_log_read:
                                                        113 \cs_new_protected:Nn \__ccool_log_read:
                                                                     \__ccool_log_read:e{\g__ccool_log_to_tl}
                                                        115
                                                        116 }
                                                       (End\ definition\ for\ \verb|\_\_ccool\_log\_read:.)
\__ccool_log_write:n
                                                        118 \cs_new_protected:Nn \__ccool_log_write:n
                                                        119 {
                                                                     \bool_if:nTF{ \g__ccool_log_open_bool }
                                                        120
                                                        121
                                                                          \iow_now:Nn \g__ccool_log_iow {#1}
                                                         122
                                                                          \tl_log:n{ write~to~#1 }
                                                        123
                                                        125
                                                                     126 }
                                                        127 \cs_generate_variant:Nn \__ccool_log_write:n { e }
                                                       (End definition for \__ccool_log_write:n.)
```

```
5 make_key
```

```
\__ccool_make_key:Nn #1: \langle token \rangle
                                #2: \langle key \rangle
                                128 \cs_new_protected:Nn \__ccool_make_key:Nn
                                129 {
                                      \exp_args:NNx
                                130
                                     \ProvideDocumentCommand{#1}
                                131
                                     { D<>{\g_ccool_option_param_tl} }
                                        134
                                     }
                                135
                                136 }
                                137 \cs_generate_variant:Nn \__ccool_make_key:Nn {c}
                                (End definition for \__ccool_make_key:Nn.)
         \__ccool_make_key:n #1: \langle key \rangle
                                138 \cs_new_protected:Nn \__ccool_make_key:n
                                      \cline{1}{make_key:cn{#1}{#1}}
                                141 }
                                142 \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
                                143 \cs_new_protected:Nn \__ccool_make_key:N
                                144
                                      \seq_map_function:NN #1 \__ccool_make_key:e
                                145
                                146 }
                                (End\ definition\ for\ \_\_ccool\_make\_key:N.)
                                6
                                      make_ccool
\__ccool_make_ccool_exp:nnn
                                147 \cs_new_protected:Nn \__ccool_make_ccool_exp:nnn
                                148 {
                                      \__ccool_aux_val:Nn \g__ccool_aux_key_seq {#1}
                                149
                                      \__ccool_aux_outer_set:n{#3}
                                150
                                      \__ccool_aux_outer:n
                                151
                                152
                                        \exp_args:NNf
                                153
                                        \__ccool_seq_use:Nn
                                154
                                        \g__ccool_aux_val_seq
                                155
                                        {#2}
                                156
                                     }
                                157
                                158 }
                                (End\ definition\ for\ \_\_ccool\_make\_ccool\_exp:nnn.)
```

```
\__ccool_make_ccool_key:nnn
                               159 \cs_new_protected:Nn \__ccool_make_ccool_key:nnn
                               160 {
                                     \__ccool_prop_if_exist:nTF{#1}
                               161
                                    { \c_empty_tl }
                               162
                                    { \__ccool_prop_new:n{#1} }
                               163
                                    \exp_args:No \__ccool_aux_inner_set:n{#2}
                               164
                                    \seq_set_from_clist:Nn \g__ccool_aux_keyval_seq {#3}
                               165
                                     \__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
                               169
                               170 }
                               (End definition for \__ccool_make_ccool_key:nnn.)
                               [9]
     \__ccool_make_ccool_sideeffect:nnn
                               172 {
                                     \cline{1}{make_ccool_key:nnn{#1}{#2}{#3}}
                               173
                                     \bool_if:nTF{ \g__ccool_log_open_bool }
                               174
                               175
                                       \__ccool_log_write:n
                               176
                               177
                                         \begingroup
                                         \label{locality} $$ \left( \cool<\#1>i\{\#2\}\{\#3\} \right) \exp and after $$
                               179
                               180
                                         \endgroup \__ccool_log_entry
                                    }{\c_empty_t1}
                               182
                               183 }
                               (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
                               \cs_new_protected:Npn \__ccool_make_ccool:nnnn #1 #2 #3 #4
                               185 {
                                     \exp_args:NNx \DeclareDocumentCommand \Ccool
                                     {%^^A
                                                              4 5 6 7 8
                                              2
                                                   3
                               187
                                       +o D<>{#1} E{ i }{{#2}} m t+ s E{ s o }{{#3}{#4}} +o
                               188
                                    }
                               189
                                     {
                               190
                                       \IfValueT{##1}{##1}
                               191
                                       \__ccool_make_ccool_sideeffect:nnn{##2}{##3}{##4}
                               192
                                       \IfBooleanT{##6}
                                         \__ccool_make_ccool_exp:nnn{##2}{##7}{##8}
                                       }
                               196
                                       \bool_if:nTF{##5}
                               197
                               198
                                         \gappto{\CcoolHook}
                               199
                               200
```

```
202
                                       }
                               203
                                       {\c_empty_tl}
                               204
                                       \IfValueT{##9}
                               205
                               206
                                          \exp_not:n{ \Ccool[##9] }
                               207
                                       }
                                    }
                               210 }
                              (End definition for \__ccool_make_ccool:nnnn.)
                              7
                                     msg
                               211 \mbox{ } \mbox{msg_new:nnn } \{\_\mbox{ccool}\} \{ \mbox{ generic } \} \{ \#1 \}
                               \label{local_local} $$\min_{n\in\mathbb{N}} {\max_{n\in\mathbb{N}} {\text{iow }}{\#1}^is^closed^can't^write}$$}
                               \label{local_local} $$\max_{new:nnn {\_ccool}{ keyonly }{\#1$-does$-not$-take$-values;$-keyval$-is$-$\#2}$}
                               214 \msg_new:nnn {__ccool}{ keywrong }{#1~does~not~recognize~key~#2}
                               \label{local_local} $$\max_{new:nnn {\_ccool}{ separ }{\#1\sim expects\sim 1\sim to\sim 3\sim items,\sim \#2}$}
                               216 \msg_new:nnn {__ccool}{ unset }{#1~unset}
                              8
                                     option
\__ccool_option_inner:n #1: \langle code \rangle
                               217 \cs_new_protected:Nn \__ccool_option_inner:n
                               218 {
                               219
                                     \tl_gset:Nn \g__ccool_option_inner_tl {#1}
                               220 }
                               221 \cool_option_inner:n
                               222 {
                                     \msg_warning:nnn{ __ccool }{ unset }{ \exp_not:N \g__ccool_option_inner_tl }
                               223
                               224 }
                              (End\ definition\ for\ \verb|\_\_ccool\_option\_inner:n.|)
\__ccool_option_param:n
                              #1: \langle token \ list \rangle
                               225 \cs_new:Nn \__ccool_option_param:n
                                     \tl_gset:Nn \g__ccool_option_param_tl{#1}
                               227
                               228 }
                               230 {
                                    \msg_error:nnx{ __ccool }
                               231
                                    { generic }
                                     { \exp_not:N\g_ccool_option_param_tl~undefined }
                               234 }
                              (End definition for \__ccool_option_param:n.)
\__ccool_option_outer:n #1: \langle inline code \rangle
                               235 \cs_new_protected:Nn \__ccool_option_outer:n
                               236 {
```

 $\cline{1.8} \cline{1.8} \cli$ 

201

```
\tl_gset:Nn \g__ccool_option_outer_tl {#1}
                                238 }
                                     __ccool_option_outer:n
                                239
                                240 {
                                      \msg_warning:nnn{ __ccool }{ unset }{ \exp_not:N \g__ccool_option_outer_tl }
                                241
                                242 }
                               (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\}
                                243 \cs_new_protected:Nn \__ccool_option_separ:n
                                      \cs_gset:Npn \g__ccool_option_separ_tl {#1}
                                246 }
                                     __ccool_option_separ:n
                                248
                                      \msg_warning:nnn{ __ccool }{ unset }{ \exp_not:N \g__ccool_option_separ_tl }
                                249
                                250 }
                               (End\ definition\ for\ \verb|\_\_ccool\_option\_separ:n.)
                               9
                                      prop
  \__ccool_prop_append:NN
                               #1: \langle prop_1 \rangle
                               #2: \langle prop_2 \rangle
                                251 \cs_new_protected:Npn \__ccool_prop_append:NN #1 #2
                                252 {
                                      \cs_set:Nn \__ccool_prop_append:nn
                                253
                                        \prop_gput:Nnx #1 {##1}{ \prop_item:Nn #2{##1} }
                                      \prop_map_function:NN #2 \__ccool_prop_append:nn
                                257
                                258 }
                                259 \cs_generate_variant:Nn \__ccool_prop_append:NN { cN }
                               (End\ definition\ for\ \_\_ccool\_prop\_append:NN.)
  \cdots ccool_prop_append:Nn #1: \langle prop \rangle
                               #2: \langle tl \ var \ name \rangle
                                260 \cs_new_protected:Nn \__ccool_prop_append:Nn
                                261 {
                                      \__ccool_prop_append:cN{ \__ccool_prop_name:n {#2} } #1
                                262
                                263 }
                               (End\ definition\ for\ \verb|\__ccool\_prop\_append:Nn.)
\__ccool_prop_clear_new:n #1: \langle tl var name \rangle
                                264 \cs_new_protected:Nn \__ccool_prop_clear_new:n
                                      \exp_args:No \prop_clear_new:c{ \__ccool_prop_name:n {#1} }
                                266
                                267 }
                               (End definition for \__ccool_prop_clear_new:n.)
```

```
\_ccool_prop_clear_new_map:n #1: \langle keyval \ list \rangle
                                  268 \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
                                 271
                                 272 }
                                 (End\ definition\ for\ \verb|\__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
                                 273 \cs_new:Nn \__ccool_prop_if_exist:nTF
                                        \prop_if_exist:cTF{ \qrderight \qrderight = ccool_prop_name:n $$\{\#1\} $$} $$
                                 276 }
                                 (End definition for \__ccool_prop_if_exist:nTF.)
      \__ccool_prop_item:nn #1: \langle tl var name \rangle
                                 #2: \langle key \rangle
                                  277 \cs_new:Nn \__ccool_prop_item:nn
                                       \prop_item:cn { \qrupe -ccool_prop_name:n {#1} } {#2}
                                  279
                                 280 }
                                 (End\ definition\ for\ \verb|\_\_ccool\_prop\_item:nn.)
       \__ccool_prop_name:n #1: \langle tl var name \rangle
                                  281 \cs_new:Npn \__ccool_prop_name:n #1{ __ccool_#1 }
                                 (End\ definition\ for\ \verb|\_\_ccool\_prop\_name:n.)
        \__ccool_prop_new:n #1: \langle tl var name \rangle
                                  282 \cs_new_protected:Nn \__ccool_prop_new:n
                                        \prop_new:c{ \__ccool_prop_name:n {#1} }
                                  284
                                  285 }
                                 (End\ definition\ for\ \verb|\_\_ccool\_prop\_new:n.|)
                                 10
                                          seq
                                 #1: \langle seq_1 \rangle
\__ccool_seq_from_prop:NNn
                                 #2: \langle seq_2 \rangle (keys)
                                 \#3: \langle prop \rangle
                                  286 \cs_new_protected:Nn \__ccool_seq_from_prop:NNn
                                        \cs_set_protected:Nn \__ccool_seq_from_prop:n
                                  289
                                       {
                                          290
                                  291
                                       \seq_map_function:NN #2 \__ccool_seq_from_prop:n
                                  292
                                  293 }
```

```
(End\ definition\ for\ \verb|\_\_ccool\_seq\_from\_prop:NNn.)
\__ccool_erw_seq_use:Nn
                             294 %
                                        \begin{arguments}
                             295 %
                                        \item \meta{ seq }
                                        \item \meta{ tokens }
                             296 %
                                        \verb|\end{arguments}|
                             297 %
                             ^{298} \cs_new:\n \__ccool_seq_use:\n
                             299 {
                                   \exp_last_unbraced:NNf
                             300
                                   \seq_use:Nnnn #1
                             301
                                   \__ccool_aux_separ:n{#2}
                             302
                             303 }
                             (End\ definition\ for\ \verb|\__ccool_erw_seq_use:Nn.|)
                             11
                                     sys
     \__ccool_sys_date:
                             304 \cs_new:Nn \__ccool_sys_date:
                             305 {
                                  \int_eval:n
                             306
                                  {
                             307
                                     \c_sys_year_int * 10000
                                     +\c_sys_month_int * 100
                                     +\c_sys_day_int * 1
                             310
                                  }
                             311
                             312 }
                             (End\ definition\ for\ \verb|\_\_ccool\_sys\_date:.)
 \__ccool_sys_date_hex:
                             313 \cs_new:Nn \__ccool_sys_date_hex:
                             314 {\int_to_hex:n{\__ccool_sys_date:}}
                             (End definition for \__ccool_sys_date_hex:.)
     \__ccool_sys_time:
                             315 \cs_new:Nn \__ccool_sys_time:
                             316 {
                                   \int_eval:n
                             317
                             318
                                     \c_sys_hour_int * 100
                             319
                                     +\c_sys_minute_int * 1
                             321
                             322 }
                             (End definition for \__ccool_sys_time:.)
 \__ccool_sys_time_hex:
                             323 \cs_new:Nn\__ccool_sys_time_hex:
                             324 {\int_to_hex:n{\__ccool_sys_time:}}
                             (End definition for \__ccool_sys_time_hex:.)
```

```
326 {
                     \c_sys_jobname_str--
                327
                     \__ccool_sys_date_hex:--
                328
                     \__ccool_sys_time_hex:
                330 }
                (End\ definition\ for\ \verb|\_\_ccool\_sys\_filename:.)
                12
                       Front-end
 \CcoolClear
               #1: \langle token \ list \rangle
                Semantics Clears any data created by \cool{decool}{decool}
                331 \NewDocumentCommand{ \CcoolClear }
                332 { D<>{\g_ccool_option_param_tl} }
                      \__ccool_prop_clear_new_map:n{#1}
                335 }
  \CcoolHook
                Example \AfterEndEnvironment{theorem}{\CcoolHook}
                336 \NewDocumentCommand{\CcoolHook}{}{\c_empty_tl}
\CcoolLambda
                #1: \langle arg \ spec \rangle
                #2: \langle code \rangle
                Example \Ccool{ EvalAt = \CcoolLambda{(#1)} }
                Semantics Creates a lambda expression with \langle integer \rangle arguments for \langle code \rangle
                337 \ProvideDocumentCommand \CcoolLambda { O{m} m }
                      \__ccool_lambda:nn { #1 } { #2 }
                340 }
```

325 \cs\_new:Nn\\_\_ccool\_sys\_filename:

\\_\_ccool\_sys\_filename:

```
\CcoolOption
               #1: \langle keyval \ list \rangle
               341 \NewDocumentCommand{ \CcoolOption }
                343 {
                     \keys_set:nn{ __ccool }{#1}
               344
               345 }
               346 \keys_define:nn { __ccool }
               347 {
      Expans
               Value eo|ee|ex|xo|xe|xx
               348 Expans .multichoices:nn = { eo, ee, ex, xo, xe, xx }
                349 { \tl_gset_eq:NN \g__ccool_option_expans_tl \l_keys_choice_tl },
                350 Expans .default:n = { xo },
                351 Expans .initial:n = { xo },
        File
               Value \( path \)
                352 File .code:n = {
               \tl_gset:Nx \g__ccool_log_file_tl{#1}
               354 },
               355 File .default:n = { \__ccool_sys_filename: },
               356 File .initial:n = { \__ccool_sys_filename: },
       Inner
               Value \langle code \rangle, with ####1 as the argument to be replaced
                357 Inner .code:n={
                    \__ccool_option_inner:n{#1}
                     \exp_last_unbraced:Nf
                350
                     \__ccool_make_ccool:nnnn
                360
                361
                       { \g_ccool_option_param_tl }
                362
                       { \g_ccool_option_inner_tl }
                363
                       { \g_ccool_option_separ_tl }
                       { \g_ccool_option_outer_tl }
               367 },
               368 Inner .value_required:n = false,
               369 Inner .default:n = {###1},
               370 Inner .initial:n = {####1},
       Param
               Value \(\langle token \ list \rangle \)
               371 Param .code:n={
                   \__ccool_option_param:n{#1}
                    \exp_last_unbraced:Nf
                    \__ccool_make_ccool:nnnn
               374
                    {
               375
```

```
{ \g_ccool_option_inner_tl }
        377
               { \g_ccool_option_separ_tl }
        378
               { \g_ccool_option_outer_tl }
        379
        380
        381 },
        382 Param .value_required:n = false,
        383 Param .default:n = { Default },
        384 Param .initial:n = { Default },
Outer
        Value \langle code \rangle, with ####1 as the argument to be replaced
        385 Outer .code:n={
             \__ccool_option_outer:n{#1}
             \exp_last_unbraced:Nf
             \__ccool_make_ccool:nnnn
        388
             {
        389
               { \g_ccool_option_param_tl }
        390
               { \g_ccool_option_inner_tl }
        391
               { \g_ccool_option_separ_tl }
                { \g_ccool_option_outer_tl }
        395 },
        396 Outer .value_required:n = false,
        397 Outer .default:n = { \left\{ \text{ensuremath} \right\} \},
        398 Outer .initial:n = { \ensuremath{####1} },
Separ
        Value That of 'separators' in [2, Section 8 of I3seq]
        399 Separ .code:n={
             \__ccool_option_separ:n{#1}
        400
             \exp_last_unbraced:Nf
        401
             \__ccool_make_ccool:nnnn
        403
               { \g_ccool_option_param_tl }
               { \left\{ \ \ \ \ \ \ \right. }
               { \g_ccool_option_separ_tl }
        406
               { \g_ccool_option_outer_tl }
        407
             }
        408
        409 },
        410 Separ .value_required:n = false,
        411 Separ .default:n = { {\ }and{\ } } { ,{\ } } { ,{\ }and{\ } },
        412 Separ .initial:n = { {\ }and{\ } } { ,{\ } } { ,{\ }and{\ } },
Write
        Value (boolean)
        ^{413} Write .code:n = {
        414 \bool_if:nTF{#1}
             {\__ccool_log_open:}
             {\__ccool_log_close:}
        417 },
```

{ \g\_ccool\_option\_param\_tl }

```
418 Write .value_required:n = false,
419 Write .default:n = \BooleanFalse,
420 Write .initial:n = \BooleanFalse
421 }
```

## $\CcoolRead$

#1:  $\langle path \rangle$ 

#### Semantics

- 1. Reads the definitions in  $\langle path \rangle$ .
- 2. Writes to ccool.log: 'read from \langle path\rangle'

```
422 \NewDocumentCommand{\CcoolRead}
423 {0}
424 {
425 \IfValueTF{#1}
426 {\__ccool_log_read:e{#1}}
427 {\__ccool_log_read:}
428 }
```

#### \CcoolVers

Semantics Expands to the package's version

```
429 \NewDocumentCommand{\CcoolVers}
430 {}
431 {\use:c{ver@ccool.sty}}
```

## 13 Closing

432 **\ExplSyntaxOff**