

# How to use Knowledge

Isa Vialard

Max Planck Institute for Software Systems

February 23, 2026

Logic and Mentoring Workshop @ CSL2026

What is knowledge?

What is knowledge.sty ?

# What is knowledge.sty ?

A LaTeX package by Thomas Colcombet



# What is knowledge.sty ?

A LaTeX package by Thomas Colcombet



Links a notion to its definition:

Reading a paper, you come across a **weird notion** defined a loooooong time ago. But when exactly ?

# What is knowledge.sty ?

A LaTeX package by Thomas Colcombet



Links a notion to its definition:

Reading a paper, you come across a **weird notion** defined a loooooong time ago. But when exactly ?



**Definition 1.** *Let us define **weird notion** as ...*

## How to read papers:

- Not linearly
- Sometimes diagonally
- Know what you are looking for

## How to read papers:

- Not linearly
- Sometimes diagonally
- Know what you are looking for

→ **When writing**, keep that in mind. That means:

- Self-contained theorems
- Clear structure
- Don't be afraid to be redundant

# How to use knowledge.sty?

```
\usepackage[composition,notion,quotation]{knowledge}
```

- Composition: when writing,  
Paper: when printing,  
Electronics: online version.
- Notion: Creates a directive called notion (will explain later)
- Quotation: to use ""..."" and "..."

# Introducing a new notion

In main.tex:

```
\begin{definition}  
Let us define "newnotion" as ...  
\end{definition}
```

Later we use "newnotions" again ...

# Introducing a new notion

In main.tex:

```
\begin{definition}  
Let us define "newnotion" as ...  
\end{definition}
```

Later we use "newnotions" again ...

**Definition 1.** *Let us define newnotion as ...*

Later we use newnotions again ...

# Introducing a new notion

Then, in preamble:

```
\knowledge{notion}  
| newnotion  
| newnotions
```

**Definition 1.** *Let us define `newnotion` as ...*

Later we use `newnotions` again ...

# Tip: look at main.diagnose



The screenshot shows a text editor window with the title bar '\*main.diagnose' and the path '~/Documents/Work/gap-order-constraints-system'. The editor contains the following text:

```
1 *****
2 * Summary *
3 *****
4
5 8 undefined knowledge(s).
6 0 autoreference(s) are introduced twice.
7 4 autoreference(s) are used but not introduced.
8
9 28 autoreference(s) are properly used.
10 6 autoreference(s) are defined but not used.
11
12
13 *****
14 * Help *
15 *****
16
17 \knowledgeconfigure{diagnose bar=false} deactivate `|'-notation in diagnose file.
18 \knowledgeconfigure{diagnose help=false} deactivate long help in the diagnose file.
19 \knowledgeconfigure{diagnose line=true} add line numbers to diagnose file.
20
21 *****
22 * Undefined knowledges *
23 *****
24
25 \knowledge{ignore}
26 | newnotion
27 | newnotions
28
29
```

At the bottom of the editor, the status bar shows: Plain Text ▾ Tab Width: 8 ▾ Ln 28, Col 1 ▾ INS

# Introducing a new macro

```
\newcommand{\newmacro}[2]{\mathcal{C}^{\#1}(\#2)}
```

```
\knowledgenewrobustcmd{\newmacro}[2]{\cmdkl{\mathcal{C}^{\#1}(\#2)}}
```

```
\begin{definition}
```

Let us define "newnotion" as ...

```
\end{definition}
```

Later we use "newnotions" again ...

```
\AP
```

Now let's define  $\$ \backslash \text{intro} * \backslash \text{newmacro} \{a\} \{b\} \$$  as ...

Later we use  $\$ \backslash \text{newmacro} \{c\} \{d\} \$$  ...

**Definition 1.** *Let us define newnotion as ...*

Later we use newnotions again ...

• Now let's define  $C^a(b)$  as ...

Later we use  $C^c(d)$  ...

# Tip: Anchor points

```
\begin{definition}
Let us define "newnotion" as ...
\end{definition}
```

Later we use "newnotions" again ...

**\AP**

Now let's define  $\$ \text{\intro*}\text{\newmacro{a}}\{\text{b}}\$$  as ...

Later we use  $\$ \text{\newmacro{c}}\{\text{d}}\$$  ...

**Definition 1.** *Let us define `newnotion` as ...*

Later we use `newnotions` again ...

• Now let's define  $C^a(b)$  as ...

Later we use  $C^c(d)$  ...

```

1 *****
2 * Summary *
3 *****
4
5 8 undefined knowledge(s).
6 0 autoreference(s) are introduced twice.
7 4 autoreference(s) are used but not introduced.
8
9 29 autoreference(s) are properly used.
10 6 autoreference(s) are defined but not used.
11
12
13 *****
14 * Help *
15 *****
16
17 \knowledgeconfigure{diagnose bar=false} deactivate `|'-notation in diagnose file.
18 \knowledgeconfigure{diagnose help=false} deactivate long help in the diagnose file.
19 \knowledgeconfigure{diagnose line=true} add line numbers to diagnose file.
20
21 *****
22 * Undefined knowledges *
23 *****
24
25 \knowledge{ignore}
26 | newnotion
27 | newnotions
28
29 *****
30 * Autoref used without intro *
31 *****
32
33 \nointro{default}{base}{\Exp }
34 \nointro{default}{base}{\Cmp }
35 \nointro{default}{base}{cost function}
36 \nointro{default}{base}{Two-Counter Machine}
37
38
39 *****
40 * Autoref introduced but not used *
41 *****
42
43 macros.tex:35: {\Cplus }{default}{base}
44 knowledge.tex:61: {gap-definable}{default}{base}
45 knowledge.tex:72: {game with guess actions}{default}{base}
46 knowledge.tex:99: {arena}{default}{base}

```

Main.diagnose can help you see:

- what you forgot to define
- what you defined twice

Main.diagnose can help you see:

- what you forgot to define
- what you defined twice
- what you defined but never used

Main.diagnose can help you see:

- what you forgot to define
- what you defined twice
- what you defined but never used
- how many different terms you used for the same thing

Main.diagnose can help you see:

- what you forgot to define
- what you defined twice
- what you defined but never used
- how many different terms you used for the same thing
- if you used "..." when you wanted to use "..."

Main.diagnose can help you see:

- what you forgot to define
- what you defined twice
- what you defined but never used
- how many different terms you used for the same thing
- if you used "..." when you wanted to use "..."

Advice: Add quotation during writing; Troubleshoot at the end.

- Choose your colors
- Scoping: some notion mean one thing in one section, and another meaning elsewhere
- Linking to urls
- etc...

Very good documentation (written using knowledge!!!)

```
\begin{scope}\label{scopename}
```

```
\knowledge{notion}  
  |notioninscope
```

```
  ...
```

```
\end{scope}
```

Then call it with "notioninscope@scopename" (even from out of scope).

Pros ✓

- + Reader-friendly
- + Helps you to write rigorously
- + Good distraction when you don't really want to write

Cons ✗

- You will get annoyed by papers without knowledge
- Your 

3
11 #

 Key will get used