The PracTEX Journal, 2007, No. 3
Article revision 2007/08/17

LATEXing with TextMate

Charilaos Skiadas and Thomas Kjosmoen

Abstract This article discusses the TextMate text editor and its many capabilities that make working with LATEX documents a lot easier. Some of its features include syntax highlighting, various methods for automatic insertion of text (such as the begin-end blocks in environments and automatic labels for section commands), lookup of labels and cite keys based on partial matches, as well as tools for dealing with large projects.

TextMate is designed with the user in mind, so it is easy to customize it to your needs. During its short lifetime (about two and a half years) it has gained many supporters and has become a very popular text editor for the Mac OS X platform, and especially among LATEX users, as can be seen from the exponential growth in the number of users, and the large number of LATEX related questions on the TextMate mailing list.

In addition to this article, the first author's weblog can be used as a starting point for learning more about using TextMate for LATEX: `http://skiadas.dcostanet.net/afterthought`

1 Introduction

We spend a large part of our lives working on LATEX documents. It could be our new hot paper to be published in a top scientific journal, our *magnum opus* that we have worked on for the last 10 years and is already more than 1000 pages long, or just a shopping list for tomorrow's trip to the grocery store.

But whatever the content of that LATEX document is, we are all faced with a common set of tasks that we need to perform:

1. Common text-editing tasks, such as adding new text, deleting existing text, or simply moving text around
2. Inserting various LATEX commands and environments, and references
3. Compiling and viewing our output, locating the parts of the code corresponding to a particular part of the output, and locating and fixing errors

Text Editors are particularly suited for editing text. After all that's what they were designed for. Many editors go further and offer a lot of extra functionality particularly targeted toward working with LaTeX documents, thus helping the user with the last two sets of tasks described above. This article is about one such editor, *TextMate*.

TextMate[1] first appeared in 2004, and it has in many ways changed the way we think about text editing. It has a clean, simple interface and it is very easy to customize, even for people with little or no programming skills. It draws most of its power from its modular approach, where a lot of the functionality is placed in various collections of preferences and commands called *bundles*. These bundles can be easily edited and configured to fit each user's particular needs. Most of TextMate's LaTeX-related abilities are drawn from the corresponding LaTeX bundle, which has been refined over the years by the contributions of many individuals, in order to make LaTeXing in TextMate as painless as possible.

One thing that should be mentioned here is that TextMate is available only on the Mac OS X operating system, as it uses a lot of features available only on that platform. However, recently a number of text editors for the Windows platform have been developed to draw from the power of TextMate's bundles mechanism.[2]

We will continue this article in Section 2 by looking closer at the core TextMate features that are useful in editing LaTeX documents. We will then in Section 3 move on to discuss some of the commands and tools that are included in the LaTeX bundle. As hopefully will become apparent, TextMate makes writing LaTeX almost a pleasure.

This article is not an attempt to offer a complete account of all that TextMate has to offer to LaTeX-editing. If you want to find out more, you can see a number of TextMate's features in action via the numerous screencasts available[3], including two specifically for the LaTeX bundle[4]. You can also learn a lot more from the recent book on TextMate ([2]), as well as the first author's weblog[5], which contains a lot of articles related to TextMate in general, and LaTeX in TextMate in

1. `http://www.macromates.com`
2. `http://www.e-texteditor.com` and `http://intype.info/home/index.php`
3. `http://macromates.com/screencasts`
4. `http://macromates.com/screencast/latex_part_1.mov` and `http://macromates.com/screencast/latex_part_2.mov`
5. `http://skiadas.dcostanet.net/afterthought`.

particular[6].

2 TextMate

Some of TextMate's features will be familiar to people used to working with an advanced text editor, for instance the ability to select whole words with a simple keystroke, to navigate quickly through the text, and to cut and paste whole chunks of text. TextMate also has a powerful "Search and Replace" mechanism that allows the use of regular expressions, which for instance makes it possible to easily search for text of the form: `\verb!anything here!`, and to automatically convert them to:

```
\begin{verbatim}
  anything here
\end{verbatim}
```

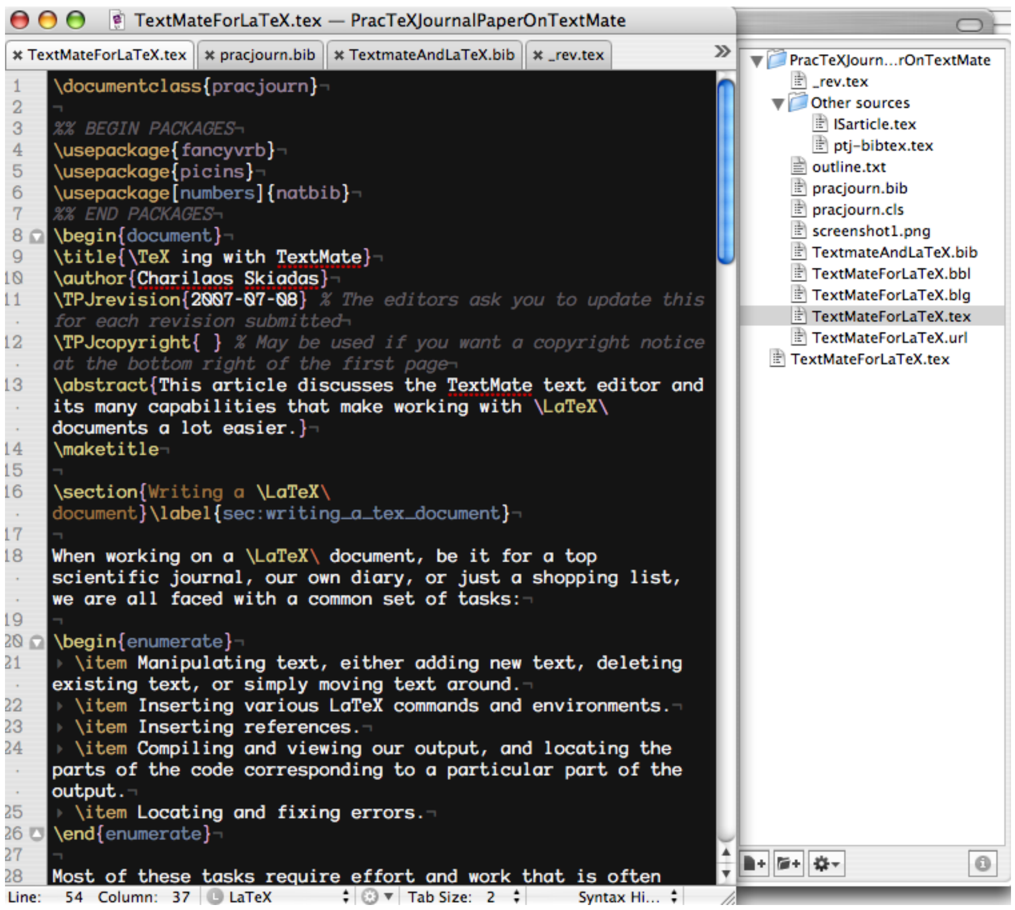
Often, we must work on several related files. To handle this, TextMate has a *Project* mode[7], which makes managing multiple files very easy. Specifically, for each project, we can set up a master LaTeX document that pulls in the other documents using `\include` or `\input` commands. TextMate is then able to view the project as a whole and handle the compilation correctly, and pull in cross references from any of the LaTeX documents. You can navigate to the various files via the *Project Drawer*, or you can move around much more quickly via the very elegant "Go to File" menu[8]. All that is required is, that you know a couple of letters from the name of the file you want to open.

2.1 Syntax Highlighting and Scopes

TextMate truly excels when it comes to highlighting a document. This is accomplished via certain parts of the bundles called *Language Grammars*. A language grammar tells TextMate to locate particular patterns in a text, and assign to each

6. See: http://skiadas.dcostanet.net/afterthought/list-of-my-textmate-pages.
7. http://www.macromates.com/textmate/manual/working_with_multiple_files
8. http://www.macromates.com/textmate/manual/working_with_multiple_files#moving_between_files_with_grace

of them a *scope*. Then, the chosen *Color Theme* assigns formatting to each scope, such as foreground and background colors, typeface, and underlining. Figure 1 shows how an early version of this particular article looks in TextMate. The formatting can be customized by the user very easily, and there are available many very nice themes for all tastes.



```
\documentclass{pracjourn}

%% BEGIN PACKAGES
\usepackage{fancyvrb}
\usepackage{picins}
\usepackage[numbers]{natbib}
%% END PACKAGES
\begin{document}
\title{\TeX ing with TextMate}
\author{Charilaos Skiadas}
\TPJrevision{2007-07-08} % The editors ask you to update this for each revision submitted
\TPJcopyright{ } % May be used if you want a copyright notice at the bottom right of the first page
\abstract{This article discusses the TextMate text editor and its many capabilities that make working with \LaTeX\ documents a lot easier.}
\maketitle

\section{Writing a \LaTeX\ document}\label{sec:writing_a_tex_document}

When working on a \LaTeX\ document, be it for a top scientific journal, our own diary, or just a shopping list, we are all faced with a common set of tasks:

\begin{enumerate}
	\item Manipulating text, either adding new text, deleting existing text, or simply moving text around.
	\item Inserting various LaTeX commands and environments.
	\item Inserting references.
	\item Compiling and viewing our output, and locating the parts of the code corresponding to a particular part of the output.
	\item Locating and fixing errors.
\end{enumerate}

Most of these tasks require effort and work that is often
```

Figure 1: TextMate's Project Drawer, and Syntax Highlighting

In addition to providing visual formatting, scopes enable other important and powerful features; we can make TextMate behave differently depending on context. For example, by default, pressing the escape key offers auto-completion based on all words in the current document. If, on the other hand, the caret[9]

9. The location where the text will appear when you type, as opposed to the *cursor*, which is the

is inside a `\ref` command, then the auto-completion takes into account only the words that appear in a `\label` command. This way, you can keep your labels very long and descriptive, relying on the fact that knowing the first couple of letters is enough to insert the whole label quickly. Likewise, if the caret is inside a `\cite` command, TextMate looks for corresponding cite keys in the associated BIBTEX files or in `\bibitem` entries in the text.

In other words, each component of your text is not just a set of characters, but has a particular meaning to TextMate, corresponding to its particular meaning to LATEX. For instance a LATEX label is really perceived by TextMate as a label, a section or subsection is understood as such, each environment is a unit of its own. Every part of the document comes alive and interacts with other parts in an intelligent way. A user can address these parts and call them by their "name", and they will reply!

Another advantage of the scope mechanism is that it allows for the insertion of one grammar in another. Thus, if you have an `lstlisting` environment containing Python code like the following:

```
\begin{lstlisting}[label=lst:vertex,float=htbp] % Python
class Vertex:
    def __init__(self,num):
        self.id = num
        self.adj = []
        self.color = 'white'
        self.dist = sys.maxint

    def addNeighbor(self,nbr,cost=0):
        self.adj.append(nbr)
        self.cost[nbr] = cost
\end{lstlisting}
```

then the Python code part will be colored according to the rules for the Python grammar, *i.e.* it will be colored as what it really is. This, as far as we know, is a unique feature of TextMate.

mouse pointer on the screen.

2.2 Snippets

One of the features of TextMate that was instantly loved by its users was *snippets*[10]. In its simplest form, a snippet is a piece of text that can be inserted into the document. Often when writing code or documents, we rewrite a lot of text. With TextMate, we can save those chunks of text as snippets, and have them inserted by using a shortcut. Think of them, if you like, as your e-mail signature. However, this would be like saying that LaTeX is just a software for creating shopping lists.

In fact, snippets are so much more than just boilerplate; they can be *dynamic*. In other words, they can include code that gets executed when the snippet is inserted. This way, for instance, you could have a snippet that inserts the current date and time, the amount of free space available in your computer, a full listing of all files in hard drive, a stock price downloaded from the internet that very moment, or anything else you can possibly imagine. More importantly, you can insert *placeholders* and *tab stops*, that you would use to type in text in various parts of the snippet. For example, this allows you to create a little snippet that, at the press of a key, will insert the text:

```
\begin{foo}
  bar
\end{foo}
```

and highlight the word `foo` in the `\begin` block. Whatever you type in there will be automatically *mirrored* on the `foo` in the `\end` block. When you are done with that, pressing tab would automatically select the word `bar` for you. This saves us a lot of keystrokes and makes sure we never forget to close an environment; perhaps the most common mistake we make, not to mention difficult to track down.[11] Taking this concept one step further, we can for instance create a snippet that automatically creates a label based on the section name.[12] Another option, a powerful and versatile one, is to use commands to insert snippets. We will discuss commands in some detail in the next section.

10. `http://www.macromates.com/textmate/manual/snippets`

11. If you worry about how you would create this particular snippet, don't: It's already built into the LaTeX bundle. But you can take our word for it, that it is indeed very easy to create.

12. Before your rush to your computers and try to create this snippet, keep in mind that this is already built into the LaTeX bundle as well.

The icing on the proverbial cake is that the snippets are easy for non-technical users to create. One quick read through the snippet section in the manual is all it takes to get you started on creating your very own snippets!

2.3 Commands

Snippets are easy to create, but they are somewhat limited. A more powerful feature, one that has almost unlimited capabilities, are *commands*[13]. They do, however, require a bit more work and some programming skills. Simply put, a command is a script which can process either the whole document, or part of it, and make changes to it, create a new document, or even generate an HTML window. This HTML window can even contain links back to specific parts of the document. You can write commands in any scripting language, like Bash, Ruby, Perl, Python, etc. In fact, all of the functionality discussed in the next section is provided by TextMate commands, so there will be a lot of examples of commands there. Here we will briefly discuss the various ways in which commands can be customized.

First of all, we have a number of options about the input text that the command will accept. The input could be the entire document, the selected text, or simply nothing. Secondly, we have many options for the output; it could replace the selected text, it could be used to create a new document, used to create an HTML window, or simply discarded.

Finally, and this is something that can be done for snippets as well, you can specify a scope for the command. Thus, the command can only be executed when the caret is in that particular scope, *e.g.* we can have commands that only work when inside a math environment or if the document is a Beamer document.

Let us consider a specific example of what commands can do. You could create a search command that will take into account the current selection, or the current word if you haven't selected anything, and will find all appearances of the word in the text. The command could then open up an HTML window listing all the locations of the search hits, allowing you to click in the HTML window and be taken to those locations in the text.

A specific type of command, a *Drag Command*, allows you to specify to TextMate what you want it to do when files of particular type are dropped into the

13. `http://www.macromates.com/textmate/manual/commands`

text. For instance, a command in the LaTeX bundle tells TextMate to automatically create a figure environment when an image file is dropped into a LaTeX document.

We have now covered the basics of what commands can do. The details are beyond the scope of this article, and we encourage you to take a look at the TextMate manual and explore the existing bundles to find out more.[14]

3 TextMate's LaTeX Bundle

Let us now move on to explore the LaTeX bundle in some detail. The LaTeX bundle is a set of *TextMate commands*[15], snippets and *macros*, that collectively enhance TextMate's LaTeX editing abilities. Roughly speaking, they fall under the following broad categories:

- Project manipulation; compilation, navigation, outline
- Automatic text insertion; environments, wrapping, labels, and references

3.1 Project Manipulation

Compilation The most essential command in the bundle is the *Typeset & View* command, which typesets the document and opens the resulting output file (usually PDF) in the appropriate viewer program. The command is smart in the sense that if you are working with a project involving a master document, then the master document will be compiled, regardless of what document the command was issued from. It is also smart about looking at the packages used, and deciding based on those what compilation steps to take.[16]

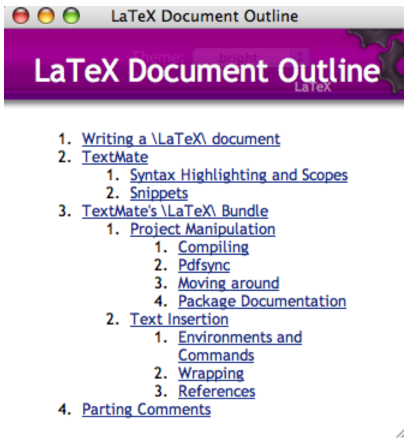
When the command is executed, the compilation log is presented in a separate window with all warnings and errors hyperlinked to their corresponding positions in the document. This is, of course, a feature that most LaTeX-editors have.

14. When TextMate runs a command, it makes many useful environment variables available to the command. These environment variables provide information about the current document. For more details, see `http://www.macromates.com/textmate/manual/environment_variables`.

15. `http://www.macromates.com/textmate/manual/commands`

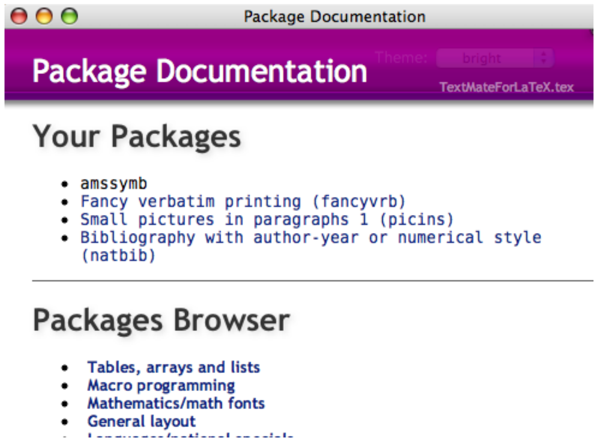
16. For instance if it detects the `pstricks` package, it will use latex+dvi2ps+ps2pdf instead of pdflatex.

Pdfsync If you are using a previewer that utilizes the `pdfsync` package[17], then you can easily go back and forth between the TEX source and the previewed PDF file. Namely, a viewer command can take you from a location in the previewed PDF file to the corresponding location in the LATEX document. Conversely there is a command in TextMate that will take you to the previewer at the area corresponding to the current caret location in the document.[18]



1. Writing a \LaTeX\ document
2. TextMate
    1. Syntax Highlighting and Scopes
    2. Snippets
3. TextMate's \LaTeX\ Bundle
    1. Project Manipulation
        1. Compiling
        2. Pdfsync
        3. Moving around
        4. Package Documentation
    2. Text Insertion
        1. Environments and Commands
        2. Wrapping
        3. References
4. Parting Comments

(a) The "Show Outline" command provides an outline of the entire project for easy navigation.



Your Packages

- amssymb
- Fancy verbatim printing (fancyvrb)
- Small pictures in paragraphs 1 (picins)
- Bibliography with author-year or numerical style (natbib)

Packages Browser

- Tables, arrays and lists
- Macro programming
- Mathematics/math fonts
- General layout

(b) The "Package Browser" window allows navigating through the documentation for the various LATEX packages.

Figure 2: Document and Package Navigation is made easier by two TextMate commands.

Navigation TextMate has a built-in command for navigating among the various 'symbols' of the document, which in the case of LATEX would be the various sections, subsections etc. This command shares the elegant interface of the "Go to File" command[19]. There is also the "Show Outline" command, which produces an outline of the entire project (Figure 2a). Clicking on any item in the outline

17. 'Skim' (http://skim-app.sourceforge.net) is a very nice such viewer.
18. The precision of this back and forth is of course limited by the information stored by `pdfsync`. Another popular LATEX editor, TeXShop, utilizes an additional Mac OS X technology, Spotlight, to offer more precise syncing by using text comparisons.
19. http://www.macromates.com/textmate/manual/navigation_overview#function_pop-up

takes you to the corresponding position in the project, regardless of which file that position is in. Finally, we have a command that take us from an `\include` line to the actual file being included.

Package Documentation For those of you constantly referring to package documentation, the "Documentation for Package" command offers easy access to the documentation of all packages via a "Package Browser" window. You can then navigate through the various packages, and have the corresponding documentation files open up in your favorite PDF viewer.

3.2 Automatic Text Insertion

Environments and Commands Adding text is probably what we do most of the time when working with a LaTeX document. TextMate has a number of commands to help us do this. Two of the most often used ones are the commands for creating environments and LaTeX commands based on an abbreviation. For instance, typing `it` and pressing the key combination that triggers the "Insert Environment" command produces:

```
\begin{itemize}
  \item
\end{itemize}
```

with the caret right at the end of the `\item` line. Pressing the Enter key at that point will add a new line and automatically insert the `\item` part. The "Insert Environment" command responds to many other abbreviations, and you may add your own abbreviations as well.

Wrapping Two other commands allow you to wrap the currently selected text in a LaTeX environment or command. Suppose that you want to make `two words` bold. This is simply done by selecting the `two words`, and using the appropriate shortcut.[20] The command will then wrap the selection like `\emph{two words}` and automatically select `emph`. You can now enter the wanted command, in this case

20. It should be noted that using these shortcuts without first selecting any text will simply wrap the word which is in contact with the caret.

`textbf`[21], and finish by pressing tab to place the caret after the newly wrapped words. Similarly, we have a shortcut for wrapping a selection in an environment, again giving you the environment argument selected for entry. For example, if you want to center the selection, enter `center` in the argument placeholder and press tab to finish:

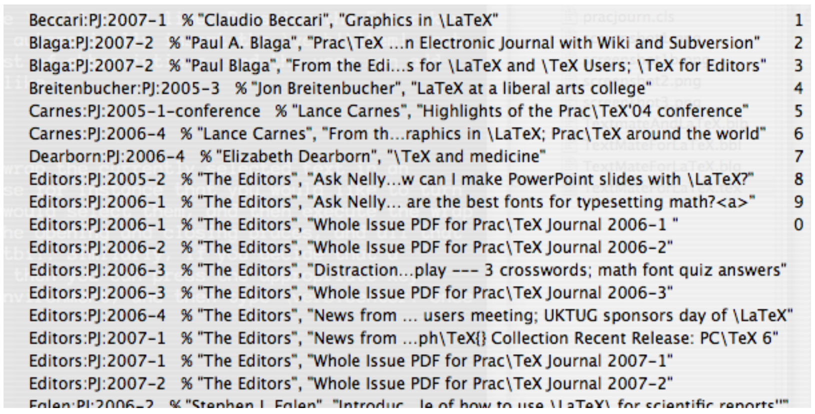
```
\begin{center}
  two words
\end{center}
```

There is even a command that will allow you to change the name of an existing environment, and another one that will toggle between a starred and a non-starred environment. Both can be executed if the caret is inside the environment (*i.e.* without having to select the entire environment).

References Perhaps the most tedious task in LaTeX is inserting references, both in the case of the `\cite` and the `\ref` commands. Suppose you want to insert a citation from the PracTeX journal bibliography[22], namely an article on BibTeX ([1]). You are guessing it has "BibTeX" in the title, but can't remember anything else about it, and you certainly can't remember the exact cite key for it. In TextMate, all you have to do is type: `\cite{Bib}` and press the key for completion, and it will become `\cite{Fenn:PJ:2006-4}`. (In fact this is exactly how we created this entry just now.) If, on the other hand, you type `\cite{TeX}` and then press the completion key, then you get a menu to choose from, with all the items that match the word TeX (Figure 3).

A similar principle extends to label referencing. Let us assume you used the common convention that the labels for all equations start with `eqn:`, those for sections with `sec:`, and other labels correspondingly, and that you wanted to add a reference to a particular equation number. Then, you could simply type: `\eqref{eqn}` and press the completion key (the same key as discussed above, but since the scope is different now, TextMate will change behaviour accordingly). TextMate will reply with offering you a list of all the labels in your project, *i.e.* from all the files included in the master document, that start with the word 'eqn'.

21. Actually for this particular case there is another command that will just wrap the text in a textbf block, without you having to type anything else.
22. http://www.tug.org/pracjourn/pracjourn.bib



```
Beccari:PJ:2007-1  % "Claudio Beccari", "Graphics in \LaTeX"                                  1
Blaga:PJ:2007-2  % "Paul A. Blaga", "Prac\TeX ...n Electronic Journal with Wiki and Subversion"  2
Blaga:PJ:2007-2  % "Paul Blaga", "From the Edi...s for \LaTeX and \TeX Users; \TeX for Editors"  3
Breitenbucher:PJ:2005-3  % "Jon Breitenbucher", "LaTeX at a liberal arts college"              4
Carnes:PJ:2005-1-conference  % "Lance Carnes", "Highlights of the Prac\TeX'04 conference"      5
Carnes:PJ:2006-4  % "Lance Carnes", "From th...raphics in \LaTeX; Prac\TeX around the world"    6
Dearborn:PJ:2006-4  % "Elizabeth Dearborn", "\TeX and medicine"                                7
Editors:PJ:2005-2  % "The Editors", "Ask Nelly...w can I make PowerPoint slides with \LaTeX?"   8
Editors:PJ:2006-1  % "The Editors", "Ask Nelly... are the best fonts for typesetting math?<a>"  9
Editors:PJ:2006-1  % "The Editors", "Whole Issue PDF for Prac\TeX Journal 2006-1 "             0
Editors:PJ:2006-2  % "The Editors", "Whole Issue PDF for Prac\TeX Journal 2006-2"
Editors:PJ:2006-3  % "The Editors", "Distraction...play --- 3 crosswords; math font quiz answers"
Editors:PJ:2006-3  % "The Editors", "Whole Issue PDF for Prac\TeX Journal 2006-3"
Editors:PJ:2006-4  % "The Editors", "News from ... users meeting; UKTUG sponsors day of \LaTeX"
Editors:PJ:2007-1  % "The Editors", "News from ...ph\TeX{} Collection Recent Release: PC\TeX 6"
Editors:PJ:2007-1  % "The Editors", "Whole Issue PDF for Prac\TeX Journal 2007-1"
Editors:PJ:2007-2  % "The Editors", "Whole Issue PDF for Prac\TeX Journal 2007-2"
Eglen:PJ:2006-2  % "Stephen J. Eglen", "Introduc...le of how to use \LaTeX\ for scientific reports'"
```

Figure 3: The menu provided by the Bibliography Completion command offers summary information for each matching bibliographic entry.

You can then pick the preferred one. Similarly, in the current document there is a label `sec:text_insertion`. If you simply type: `\ref{ins}` and press the completion key, it will be converted to `\ref{sec:text_insertion}`.

4 Parting Comments

This article has just scratched the surface of what TextMate can do to help you with your LaTeXing. There are commands for creating new tables and for converting text selection to a LaTeX table, for converting a text selection to an itemized list, and much more. Not to mention the support TextMate offers for working with Sweave documents.[23] The screencasts referred to in the introduction can illustrate what TextMate can do much better than any words could, and the LaTeX bundle help file has more up-to-date information, along with the corresponding shortcuts for all the commands.

Though TextMate is not free[24], it comes with a 30 day all-features-included trial version that is more than enough to get you hooked on it. The authors are two of those happy TextMate-addicts, with the first author having used it for more than two years, and the second author having used it for less than a year.

23. http://www.stat.umn.edu/~charlie/Sweave

24. The core TextMate application is not free, but the bundles are an open-source project and more than 100 people contribute to them.

References

[1] J. Fenn. Managing citations and your bibliography with bibtex. *The PracTEX Journal*, (4), 2006.

[2] James Edward Gray II. *TextMate: Power Editing for the Mac*. Pragmatic Bookshelf, 2007.