Quick start for LaTeXing with IEEEtran.cls for IEEE Computer Society Conferences

First Author, Second Author
University of Examples, Germany
{lastname}@example.org

Third Author
School of Electrical and
Computer Examples
Georgia Institute of Examples
Atlanta, Georgia 30332–0250
http://www.example.org

Abstract—**Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.**

I. Introduction

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus enim. Vestibulum pellentesque felis eu massa.TODO!

The remainder of the paper starts with a presentation of related work (Section II). It is followed by a presentation of hints on LaTeX (Section III). Finally, a conclusion is drawn and outlook on future work is made (Section IV).

II. Related Work

Winery [1] is a graphical modeling tool. The whole idea of TOSCA is explained by Binz et al. [2].

III. LaTeX Hints

This section contains hints on writing LaTeX. It focuses on minimal examples, which can be directly adapted to the content

A. Handling of paragraphs

One sentence per line. This rule is important for the usage of version control systems. A new line is generated with a blank line. As you would do in Word: New paragraphs are generated by pressing enter. In LaTeX, this does not lead to a new paragraph as LaTeX joins subsequent lines. In case you want a new paragraph, just press enter twice! This leads to an empty line. In word, there is the functionality to press shift and enter. This leads to a hard line break. The text starts at the beginning of a new line. In LaTeX, you can do that by using two backslashes (\\). This is rarely used.

Please do *not* use two backslashes for new paragraphs. For instance, this sentence belongs to the same paragraph, whereas the last one started a new one. A long motivation for that is provided at http://loopspace.mathforge.org/HowDidIDoThat/TeX/VCS/#section.3.

```
511  One sentence per line.
512  This rule is important for the usage of version
     ↪  control systems.
513  A new line is generated with a blank line.
514  As you would do in Word:
515  New paragraphs are generated by pressing enter.
516  In LaTeX, this does not lead to a new paragraph
     ↪  as LaTeX joins subsequent lines.
517  In case you want a new paragraph, just press
     ↪  enter twice!
518  This leads to an empty line.
519  In word, there is the functionality to press
     ↪  shift and enter.
520  This leads to a hard line break.
521  The text starts at the beginning of a new line.
522  In LaTeX, you can do that by using two
     ↪  backslashes (\textbackslash\textbackslash).
523  \\
524  This is rarely used.
525
526  Please do \textit{not} use two backslashes for
     ↪  new paragraphs.
527  For instance, this sentence belongs to the same
     ↪  paragraph, whereas the last one started a
     ↪  new one.
528  A long motivation for that is provided at
     ↪  \url{http://loopspace.mathforge.org/ ⌋
     ↪  HowDidIDoThat/TeX/VCS/#section.3}.
```

B. Notes separated from the text

The package mindflow enables writing down notes and annotations in a way so that they are separated from the main text.

This is a small note.

```
536  \begin{mindflow}
537  This is a small note.
538  \end{mindflow}
```

C. Handling TODOs

Markierter Text.

```
544  \textmarker{Markierter Text.}
```

Bei \textmarker wird nur die Textfarbe geändert, da dies auch bei einigen Worten gut funktioniert.
Markierter Text.

```
550  \textcomment{Markierter Text.}{Kommentar dazu.}
```

Manuelle Markierung für Text, der seit der letzten Version geändert wurde.

```
554  \modified{Manuelle Markierung für Text, der
     ↪  seit der letzten Version geändert wurde.}
```

Das ist ein Text. Geänderter Text.

```
558  Das ist ein Text.
559  \change{FL1: Text angepasst}{Geänderter Text}.
```

Hier nur ein Kommentar.

```
563  Hier nur ein Kommentar\sidecomment{Kommentar}.
```

TODO!

```
567  \todo{Hier muss noch kräftig Text produziert
     ↪  werden}
```

D. Hyphenation

LATEX automatically hyphenates words. When using microtype, there should be fewer hyphenations than in other settings. It might be necessary to tweak the hyphenations nevertheless. Here are some hints:

In case you write "application-specific", then the word will only be hyphenated at the dash. You can also write `applica\allowbreak{}tion-specific` (result: applica tion-specific), but this is much more effort.

You can now write words containing hyphens which are hyphenated at other places in the word. For instance, `application"=specific` gets application"=specific. This is enabled by an additional configuration of the babel package.

E. Typesetting Units

Numbers can be written plain text (such as 100), by using the siunitx package as follows: $100\,\frac{\mathrm{km}}{\mathrm{h}}$, or by using plain LaTeX (and math mode): $100\frac{km}{h}$.

$5\,\%$ of $10\,\mathrm{kg}$

Numbers are automatically grouped: 123 456.

F. Surrounding Text by Quotes

Please use the "enquote command" to quote something. Quoting with "quote" or "quote" also works.
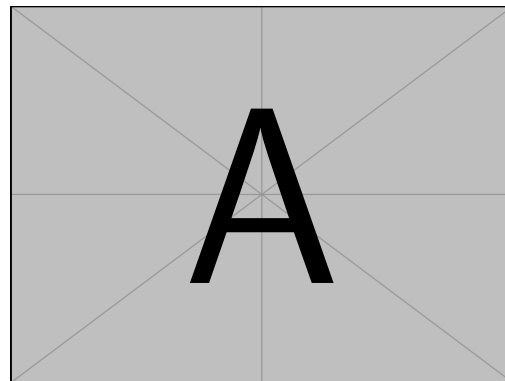
Figure 1: Example figure for cref demo

Heading1	Heading2
One	Two
Thee	Four

Figure 2: Example table for cref demo

G. Cleveref examples

Cleveref demonstration: Cref at beginning of sentence, cref in all other cases.

Figure 1 shows a simple fact, although Figure 1 could also show something else.

Figure 2 shows a simple fact, although Figure 2 could also show something else.

Section III-G shows a simple fact, although Section III-G could also show something else.

H. Figures

Figure 3 shows something interesting.
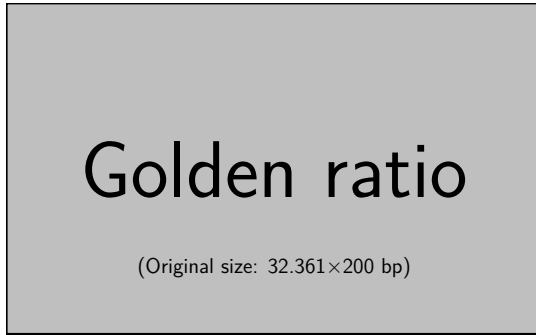
Figure 3: Simple Figure. Based on Scharrer [3].

```
648  \Cref{fig:label} shows something interesting.
649
650  \begin{figure}
651    \centering
652    \includegraphics[width=.8\linewidth]{
       ↪ example-image-golden}
653    \caption[Simple Figure]{
654      Simple Figure.
655      Based on \citet{mwe}.
656    }
657    \label{fig:label}
658  \end{figure}
```

One can span a figure across multiple columns by using `\begin{figure*}`. See Figure 4 as an example.

```
665  \begin{figure*}
666    \centering
667    % note that \textwidth is used instead of
       ↪ \linewidth
668    % This ensures that the graphics width is 60%
       ↪ of the "page" (text block), and not just
       ↪ 60% of the current text column
669    % See https://tex.stackexchange.com/a/17085/
       ↪ 9075 for details
670    \includegraphics[width=.6\textwidth]{
       ↪ example-image-16x9}
671    \caption{16x9 Figure}
672    \label{fig:16x9}
673  \end{figure*}
```

I. Sub Figures

An example of two sub figures is shown in Figure 5.

```
682  \begin{figure*}[!b]
683    \centering
684    \subfloat[Case I]{\includegraphics[width=.4
       ↪ \linewidth]{example-image-a}%
685      \label{fig:first_case}}
686    \hfil
687    \subfloat[Case II]{\includegraphics[width=.4
       ↪ \linewidth]{example-image-b}%
688      \label{fig:second_case}}
689    \caption{Example figure with two sub
       ↪ figures.}
690    \label{fig:two_sub_figures}
691  \end{figure*}
```

Note that often IEEE papers with subfigures do not employ subfigure captions (using the optional argument to `\subfloat[]`), but instead will reference/describe all of them (a), (b), etc., within the main caption. Be aware that for subfig.sty to generate the (a), (b), etc., subfigure labels, the optional argument to `\subfloat` must be present. If a subcaption is not desired, just leave its contents blank, e.g., `\subfloat[]`. An example is shown in Figure 6.

```
704  \begin{figure*}[!b]
705    \centering
706    \subfloat[]{\includegraphics[width=.4
       ↪ \linewidth]{example-image-a}%
707      \label{fig:first_case_ieee}}
708    \hfil
709    \subfloat[]{\includegraphics[width=.4
       ↪ \linewidth]{example-image-b}%
710      \label{fig:second_case_ieee}}
711    \caption{Example figure with two sub figures.
       ↪ IEEE style. (a) The first case. (b) The
       ↪ second case.}
712    \label{fig:two_sub_figures_ieee}
713  \end{figure*}
```

J. Tables

Note that IEEE does not support `\begin{table}`, one has to use `\begin{figure}`.

```
721  \begin{figure}
722    \caption{Simple Table}
723    \label{tab:simple}
724    \centering
725    \begin{tabular}{ll}
726      \toprule
727      Heading1 & Heading2 \\
728      \midrule
729      One     & Two      \\
730      Thee    & Four     \\
731      \bottomrule
732    \end{tabular}
733  \end{figure}
```
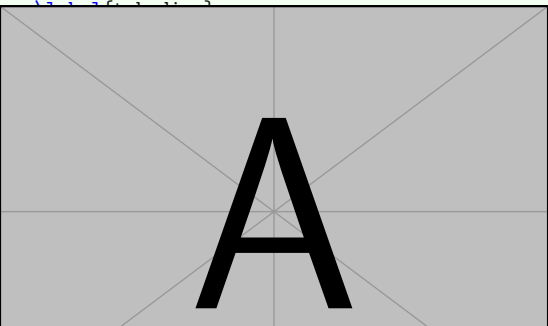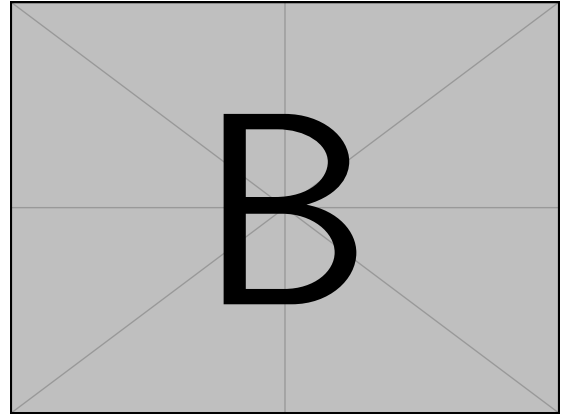
Figure 4: 16x9 Figure

(a) Case I
(b) Case II

Figure 5: Example figure with two sub figures.
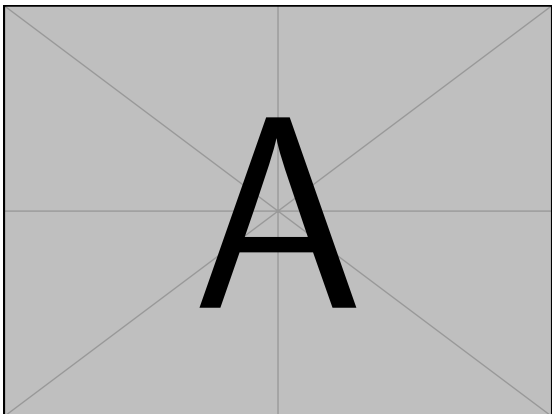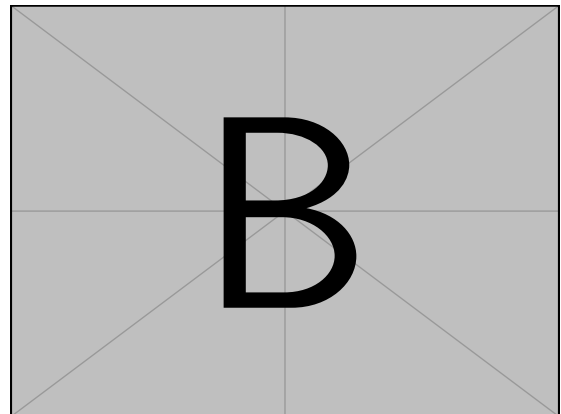
(a)
(b)

Figure 6: Example figure with two sub figures. IEEE style. (a) The first case. (b) The second case.

Figure 7: Simple Table

Heading1	Heading2
One	Two
Thee	Four

Figure 8: Table with diagonal line

Diag Column Head II	Second	Third
Diag Column Head I		
	foo	bar

K. Source Code

minte is a sophisticated package to enable properly highlighted listings. It uses the pygments library, which in turn requires Python.

Listing 1 shows source code written in XML. line 2 contains a comment.

```
1 <listing name="example">
2   <!-- comment -->
3   <content>not interesting</content>
4 </listing>
```

List. 1: Example XML listing using minted

```
Corresponding LATEX code of ./paper-minted.tex

760  \Cref{lst:XML} shows source code written in
       ↪ XML.
761  \refline{line:comment} contains a comment.
762
763  \begin{listing}[htbp]
764    \begin{↓
       ↪ minted}[linenos=true,escapeinside=||]{↓
       ↪ xml}
765  <listing name="example">
766    <!-- comment --> |\labelline{line:comment}|
767    <content>not interesting</content>
768  </listing>
769  \end{minted}
770    \caption{Example XML listing using minted}
771    \label{lst:XML}
772  \end{listing}
```

One can also typeset JSON as shown in Listing 2.

```
1 {
2   key: "value"
3 }
```

List. 2: Example JSON listing using minted

```
Corresponding LATEX code of ./paper-minted.tex

778  \begin{listing}[htbp]
779    \begin{↓
       ↪ minted}[linenos=true,escapeinside=||]{↓
       ↪ json}
780  {
781    key: "value"
782  }
783  \end{minted}
784    \caption{Example JSON listing using minted}
785    \label{lst:flJSON}
786  \end{listing}
```

Java is also possible as shown in Listing 3.

```
1 public class Hello {
2     public static void main (String[] args) {
3         System.out.println("Hello World!");
4     }
5 }
```

List. 3: Java code rendered using minted

```
Corresponding LATEX code of ./paper-minted.tex

792  \begin{listing}[htbp]
793    \begin{↓
       ↪ minted}[linenos=true,escapeinside=||]{↓
       ↪ java}
794  public class Hello {
795      public static void main (String[] args) {
796          System.out.println("Hello World!");
797      }
798  }
799  \end{minted}
800    \caption{Java code rendered using minted}
801    \label{lst:flJava}
802  \end{listing}
```

L. Itemization

One can list items as follows:

- Item One
- Item Two

```
Corresponding LATEX code of ./paper-minted.tex

810  \begin{itemize}
811    \item Item One
812    \item Item Two
813  \end{itemize}
```

With the package paralist, one can create itemizations with lesser spacing:

- Item One
- Item Two

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 819  \begin{compactitem}
> 820    \item Item One
> 821    \item Item Two
> 822  \end{compactitem}
> ```

One can enumerate items as follows:

1) Item One
2) Item Two

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 828  \begin{enumerate}
> 829    \item Item One
> 830    \item Item Two
> 831  \end{enumerate}
> ```

With the package paralist, one can create enumerations with lesser spacing:

1) Item One
2) Item Two

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 837  \begin{compactenum}
> 838    \item Item One
> 839    \item Item Two
> 840  \end{compactenum}
> ```

With paralist, one can even have all items typeset after each other and have them clean in the TeX document:

1) All these items... 2) ...appear in one line 3) This is enabled by the paralist package.

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 846  \begin{inparaenum}
> 847    \item All these items...
> 848    \item ...appear in one line
> 849    \item This is enabled by the paralist
>      ↪  package.
> 850  \end{inparaenum}
> ```

M. Other Features

The words "workflow" and "dwarflike" can be copied from the PDF and pasted to a text file.

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 856  The words \enquote{workflow} and
>      ↪  \enquote{dwarflike} can be copied from the
>      ↪  PDF and pasted to a text file.
> ```

The symbol for powerset is now correct: $\wp$ and not a Weierstrass p ($\wp$).

$\wp(1, 2, 3)$

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 860  The symbol for powerset is now correct:
>      ↪  $\powerset$ and not a Weierstrass p
>      ↪  ($\wp$).
> 861
> 862  $\powerset({1,2,3})$
> ```

Brackets work as designed: <test> One can also input backticks in verbatim text: `` `test` ``.

> **Corresponding LaTeX code of `./paper-minted.tex`**
>
> ```
> 866  Brackets work as designed:
> 867  <test>
> 868  One can also input backticks in verbatim text:
>      ↪  \verb|`test`|.
> ```

IV. Conclusion and Outlook

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

References

[1] O. Kopp *et al.*, "Winery – A Modeling Tool for TOSCA-based Cloud Applications," in *Proceedings of 11th International Conference on Service-Oriented Computing (ICSOC'13)*, ser. LNCS, vol. 8274. Springer Berlin Heidelberg, 2013, pp. 700–704.

[2] T. Binz, G. Breiter, F. Leymann, and T. Spatzier, "Portable Cloud Services Using TOSCA," *IEEE Internet Computing*, vol. 16, no. 03, pp. 80–85, May 2012.

[3] M. Scharrer, *The mwe Package*, 2017. [Online]. Available: http://texdoc.net/mwe

[4] B. Veytsman, "Latex class for the association for computing machinery – acknowledgement information," Aug. 2021. [Online]. Available: https://github.com/borisveytsman/acmart/blob/1704c8bf7eee92a1515ff755f5118b6a22bb1f8e/samples/samples.dtx#L709

All links were last followed on October 5, 2020.