Jane's LaTeX Tips, Tricks, and Personal Preferences

Jane Hoffswell | July 28, 2023

1. Get started early!

1. Get started early!

and

2. Write as you go!

3. Include well-formed citations from the start

Tips for a well-formed citation:

Standardize reference label format (i.e., author-shortname)

- Indicate when there are two authors (i.e., oneANDtwo-shortname)
- Indicate if a single author (i.e., authorSINGLE-shortname)

Use consistent names for conferences

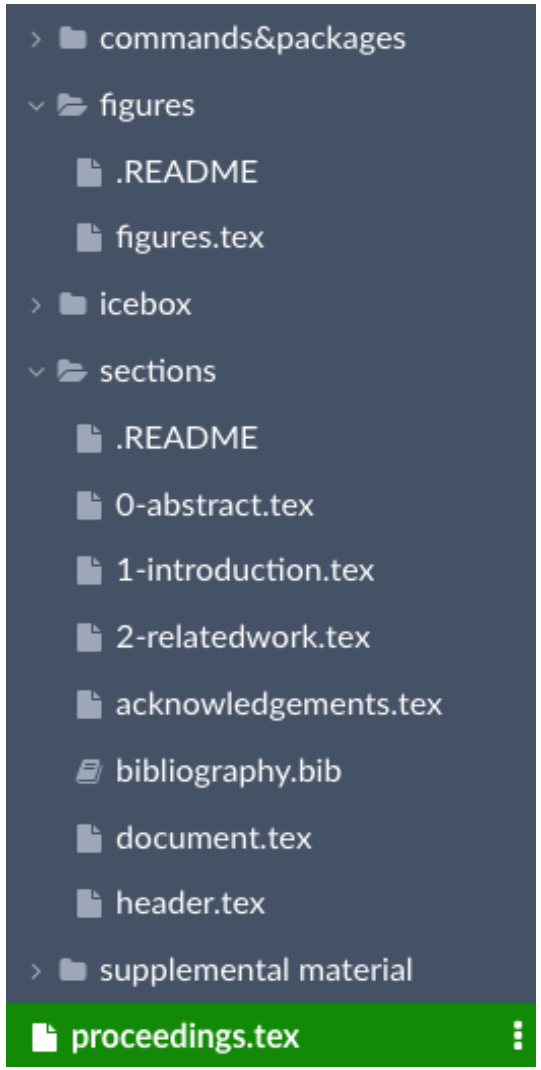
Given page constraints, remove unnecessary details

Add the DOI link and/or number

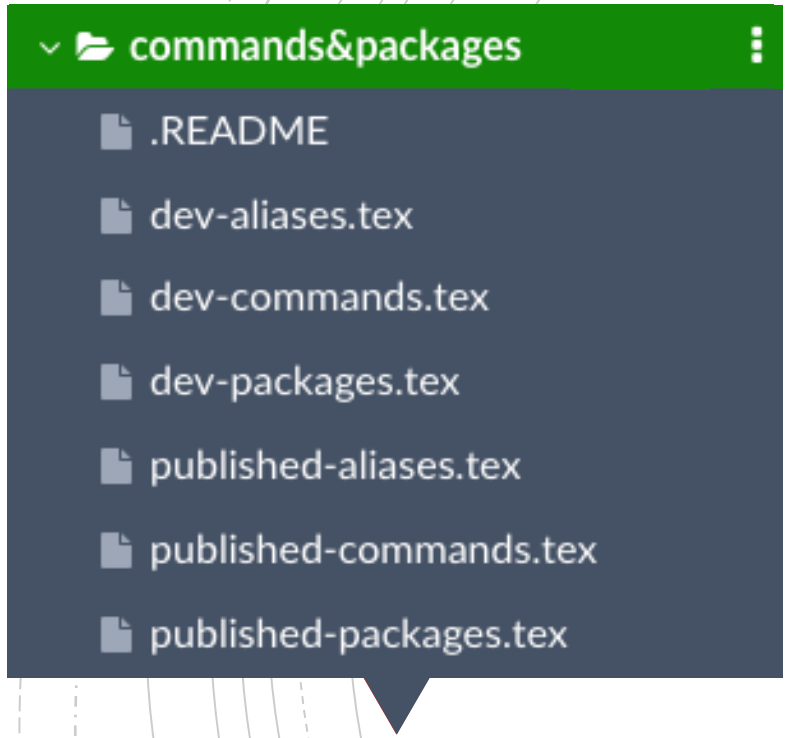
Same conference, different formats…

```
@inproceedings{hoffswellANDliu-tableRepair,
  title={Interactive repair of tables extracted from pdf documents on mobile devices},
  author={Hoffswell, Jane and Liu, Zhicheng},
  booktitle={Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems},
  pages={1--13},
  year={2019}
}

@inproceedings{hoffswell-responsiveTechniques,
  doi={https://doi.org/10.1145/3313831.3376777},
  title={Techniques for Flexible Responsive Visualization Design},
  author={Jane Hoffswell AND Wilmot Li AND Zhicheng Liu},
  booktitle={CHI Conference on Human Factors in Computing Systems},
  year={2020}
}
```

4. Keep organized with separate files and folder structure




```latex
\documentclass{article}
\usepackage[utf8]{inputenc}

\title{Jane's Sample LaTeX Template}
\author{Jane Hoffswell}

\input{sections/header}             % Header content to add before the document (packages/commands/figures)

\begin{document}

\maketitle

\input{sections/0-abstract}         % Abstract (in whatever position required by the conference template)
\input{sections/document}           % Text content to add within the document (individual paper sections)

\section*{Acknowledgements}
\input{sections/acknowledgements}   % Acknowledgements (positioned according to conference guidelines)

\bibliography{sections/bibliography}
\nocite{*}                          % Show all citations (even those that are currently not cited)

\input{icebox/icebox}               % Archived content to be removed from final paper

\end{document}
```



5. Keep all custom commands in one place under commands&packages

Use the dev- files for paper-writing (draft) support only (e.g., custom LaTeX comments)

Use the published- files for adding custom support (e.g., new style packages) that are approved by the conference venue

```
%% Note: Relevant packages added for the contents of this file
%% Please include a comment describing the purpose of each new package
\usepackage{xargs}      % Used for new commands with optional arguments
\usepackage{soul}       % Used for custom comments
\usepackage{color}      % Used for custom colors in comments
\usepackage{xspace}     % Used for abbreviation spacing
\usepackage{xpunctuate} % Used for abbreviation spacing
```

6. Always add a comment when including new packages to explain **WHY**

*Aliases are new commands with no arguments.

```
10  %% Note: New command for referencing (and easily changing) the system name
11  \newcommand{\sys}{StillUndecided}
12
13  %% Note: Commands for spacing Latin letters/abbreviations
14  \newcommand{\ie}{{i.e.,}\xspace}
15  \newcommand{\eg}{{e.g.,}\xspace}
16  \newcommand{\etal}{{et~al\xperiod}\xspace}
17  \newcommand{\aka}{{a.k.a.}\xspace}
18  \newcommand{\etc}{{etc\xperiod}\xspace}
```

```
11  %% Note: Preprint notice to add to title when sharing early version of paper
12  \newcommand{\preprintNotice}{
13      \textcolor{red}{
14          Preprint: Currently under review. Please do not redistribute.}}
```

7. Use simple aliases to enforce consistency or combat uncertainty

8. Use custom commands for more complex (or variable) styles

New commands are also useful for maintaining consistency or combatting uncertainty, while providing additional support for inputs with one or more arguments.

```
%% Note: Commands for custom styling of bold, inline paragraph headings
\newcommand{\parHeading}[1]{\vspace{8px}\noindent{\textbf{#1}}}
```

```
%% Note: Commands for custom styling of participant quotes
\newcommand{\myquote}[1]{\emph{``#1''}}                    % Italic text in quotation marks
\newcommand{\myquoteparticipant}[2]{\emph{``#1''}~(#2)}    % Italic text in quotation marks AND P# with non-breaking space
\newcommand{\mylongquote}[1]{\begin{quote}\emph{#1}\end{quote}} % Block quote in italics, no quotation marks or P#
```

```
\definecolor{lightgreen}{RGB}{121,210,121}
\newcommandx{\jane}[2][1=]
    {\setulcolor{lightgreen}{\ul{#1}} \textcolor{lightgreen}   %% Usage: \jane[(optionally) underline text]{With a comment.}
    {[\textbf{Jane:} #2]}}
```

1 INTRODUCTION

UNDER REVISION

Once upon a time, there was a paper... [**Jane:** Does this beginning make sense?] There have been lots of papers since then, but in the beginning, there was one, and it had to start somewhere. So,

2 RELATED WORK

INCOMPLETE

[TODO] *Add related work text here*

9. Use custom status badges to facilitate collaboration.

```
To add a figure to the paper, simply refer to the
corresponding command in the document:
\figureSample
```

10. Use new commands in **figures.tex** to make figure placement easier

```
%%
%% NOTE: For each figure in the paper, create a \newcommand here (as shown below)
%% that defines the figure; then, simply call this new command from the desired
%% place in the paper. This structure makes editing easier by putting all of the
%% figure descriptions in one place. Moving figures in the paper also becomes
%% easier because only one line (the place where the new command is called) needs
%% to be changed any time the figure needs to be moved.
%%
%% All figure image files and intermediate files used to generate the image should
%% also be stored in the "figures/" folder to support collaboration among authors.
%%

\newcommand{\figureSample}{
\begin{figure}[t]
    \centering
    \includegraphics[width=0.5\textwidth]{figures/sample.pdf}
    \caption{}
    \Description{}
    \label{fig:sample}
\end{figure}
}
```

11. Archive old content to remove (but save, just in case) in icebox.tex

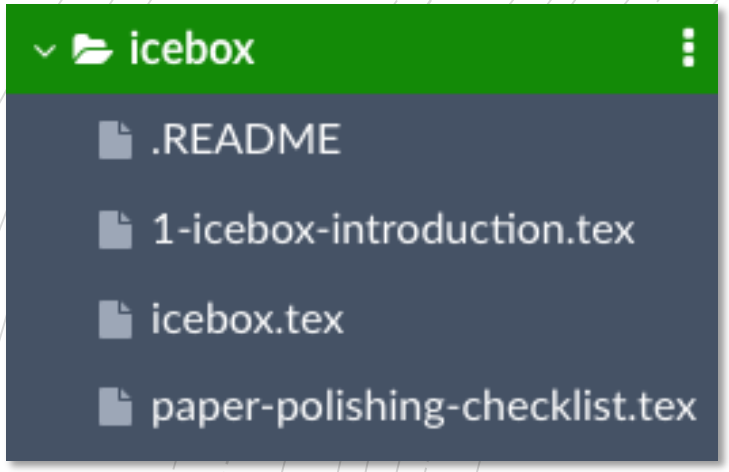
Related work section you don't need anymore?

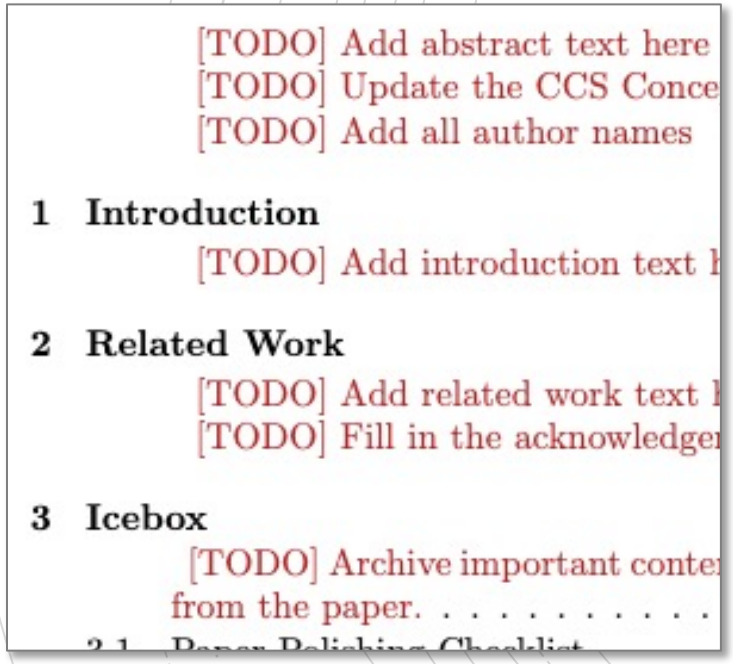
Figure that is too big for the final version?

Intro paragraph you really like, but doesn't fit the story?

Add them to the icebox!

11. Archive old content to remove (but save, just in case) in icebox.tex






```
%!TEX root = proceedings.tex

%%
%% NOTE: Move any content that should be removed from the paper to this section
%% (to keep individual section files clean). This section can act as an archive
%% from which to restore content that was previously cut from the working paper.
%% This file can easily be excluded from the final submission by commenting out
%% or deleting the \input{icebox} line from the end of proceedings.tex.
%%

\section{Icebox}

\todo{Archive important content here that should be excluded from the paper.}

\subsection{Paper Polishing Checklist}
\input{icebox/paper-polishing-checklist}

%%
%% NOTE: The table of contents is included here for reference while revising the
%% paper to provide a quick overview of the flow of the paper content, as well as
%% the list of todos to provide a quick overview of the remaining items that need
%% to be addressed for the final paper submission (organized by section).
%%
\tableofcontents

\input{icebox/1-icebox-introduction}
```

Bonus:

Jane's Paper Polishing Checklist

- ❑ Citations should have a non-breaking space ("~")
- ❑ Participant numbers should have a non-breaking space
- ❑ Figure/Section references should have a non-breaking space
- ❑ Every figure should be referenced in the text
- ❑ There should be paragraph text under every section heading
- ❑ **Remove typographic "widows" and "orphans"**
- ❑ **Figure captions should describe the image and takeaways**
- ❑ **Check for and remove dangling "this"**

and abstracting visualizations of the data flow graph to be useful to developers.

I'd count this too

The current Vega workflow consists of first writing a specification, and then handing it off to the Vega library to parse and render the final visualization. This deferred evaluation creates a lag between authoring a visualization and debugging the resultant output, making it hard to pinpoint the source of errors. A visualization of the data flow graph can tighten this feedback loop, enabling a more iterative design process. Brushing and linking allows developers to jointly inspect the specification, visualization and data flow graph, but challenges such as the behavior of interactive visualizations and hierarchical specifications makes identifying correlated elements difficult. Additional techniques, such as transient overlaid guides [SH14], will be necessary to account for these complications.

widow

3. Conclusion

The proposed techniques can help bridge the gulf of evaluation that is introduced by decoupling specification and execution. By presenting and augmenting the data flow graph, developers can jointly inspect the specification, visualization, data flow graph, and streaming data to iteratively develop visualizations. In general, visual representations of program states can provide developers with the context necessary to better interpret and interact with their code. Surfacing the program state allows developers to adjust their mental model alongside the program execution and can therefore enable more effective debugging practices by limiting this separation.

orphan

Two benefits of removing widows and orphans:

1. Makes text clearer and more concise by removing unnecessary words
2. Saves a noticeable amount of space to provide more room for content

Captions should be standalone and just as informative as the text.

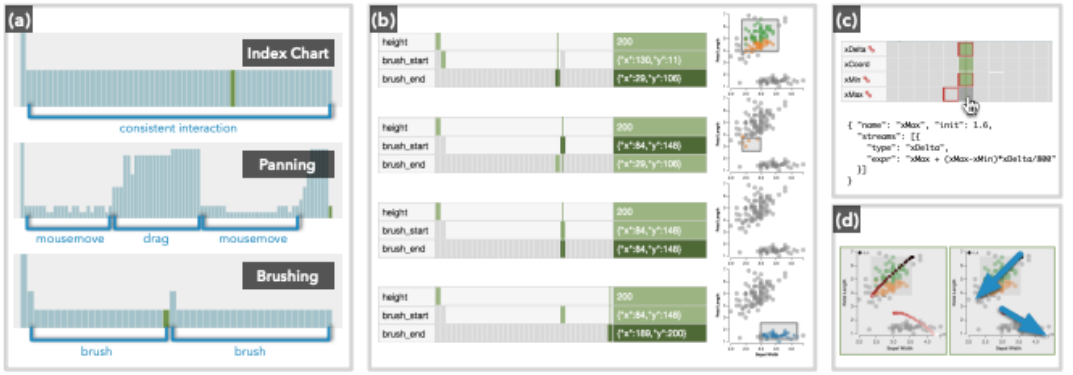


Figure 3: *The (a) overview, (b) timeline, and (c) signal annotations after performing interactions.*

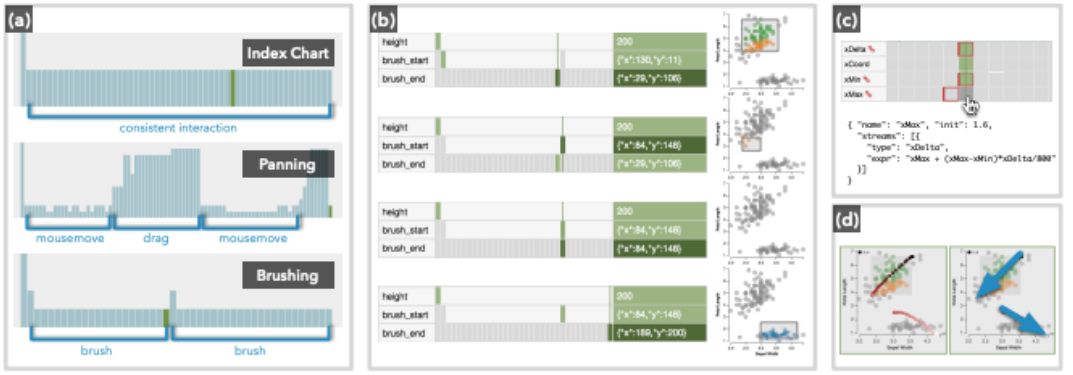


Figure 3: *The overview, timeline, and signal annotations after performing interactions. (a) The overview provides insight into different interaction patterns. (b) Stepping within a pulse allows users to see intermediate states of an interaction. The second scatterplot shows a brush representing the* new `brush_start` *and* old `brush_end`*. (c) Dependencies are shown as red outlines on hover. (d) Signal annotations overlay the visualization, with fill color encoding temporality: from darkest (past), through red (current), to lightest (future).*

“This is possible because signals express the bulk of an interaction technique, abstracting away the particular input events that trigger interactive behavior.”

simplification?

algorithm?

finding?

system?

technique?

"This _____ is possible because signals express the bulk of an interaction technique, abstracting away the particular input events that trigger interactive behavior."

fill in the blank

visualization?

approach?

model?

result?

Sometimes, it is impossible to make everything work…

so try your best and be open to compromises.

Try the template…

By Jane Hoffswell