The Black Magic of Floats in LaTeX

Raphael Frey

https://github.com/alpenwasser/TeX/

March 29, 2017

Abstract

The behavior of floats can often be confusing for the uninitiated, and yield unexpected results. This document gives a brief overview on the subject, primarily based on Leslie Lamport's *LaTeX – A Document Preparation System* [1].

This document does not cover every possible edge case, but presents some usage examples and common problem one tends to run into while working with floats. It also presents some alternatives for when floats may not necessarily be the most sensible solution.

Contents

List of Listings

List of Figures

List of Tables

1   Incomplete Summary for the Impatient

Default Float Environments in LaTeX

```
\begin{figure}[loc]    body    \end{figure}
\begin{figure*}[loc]   body    \end{figure*}
\begin{table}[loc]     body    \end{figure}
\begin{table*}[loc]    body    \end{figure*}
```

The starred versions produce double-column floats in two-column documents. They are identical to the non-starred versions in single-column documents. See Section 4 on page 7 for the meaning of `loc`.

Example: Table, Centered, Labeled and Captioned

```
\begin{table}
    \centering
    \caption{A table of curiosities}
    \label{tab:cur-table}
    \begin{tabular}{ll}
        \scshape Observation              & \scshape Explanation      \\
        Sun rose in the west this morning & universe is broken, or:   \\
                                          & too many drugs            \\
        rain is falling upwards           & definitely too many drugs \\
    \end{tabular}
\end{table}
```

Note that the `\label` must come either inside or after the `\caption` command. For `figure`s, the caption and label are usually below the content, for `table`s above.

Print Leftover Floats

```
\clearpage
\cleardoublepage
```

The `\clearpage` command prints any leftover floats, putting them on separate pages with no text. It is useful for ensuring that no floats from one chapter are printed on the first page(s) of a new chapter.
In `twoside` documents, the `\cleardoblepage` command can be used to ensure that a new section is only started on a right-hand page.
Multiple commands will not produce multiple empty pages.

Some packages which might be of interest when working with floats, for example for customizing captions, creating new captions and creating new float environments, can be found the Bibliography on page 14. Specifically the packages `caption` [3], `floatrow` [8], `float` [9], `capt-of` [10], and `captdef` [11].

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.

Figure 1: A `figure` environment with placeholder text

Table 1: Results for an experiment

Experiment Input	Experiment Output
interesting thing	interesting result!
boring thing	mildly surprising result
weird thing	very unexpected result
fascinating thing	machine broke
Xenomorph XX121	dead scientists

2 What Are Floats, Anyway?

Normal text is broken by TeX across lines and pages automatically. Some content, such as images, are not well-suited to being split into pieces. That is what floating environments are for: To provide a way to put such content in a place where it does not need to be broken; a mechanism for it to *float* to a suitable location for an optimal overall result.

Two such floating environments are provided by LaTeX by default: The `figure` and the `table`[1] environment. There are packages which define more floating environments (for example, the `listings` packages can let its code listings float, if so desired), or the user may define their own floating environments, if they so wish.

Fundamentally, the only important difference between these environments is how they are captioned and numbered: `figure` environments get a different caption and number than `table` environments. However, one may in principle put pretty much anything one desires into either environment. As long as the code itself is valid, LaTeX will not complain. Figure 1 demonstrates this by placing some Lorem Ipsum text inside its environments.

The floating behavior can be demonstrated by the fact that, in the source code of this document, Figure 1 and Table 1 are placed almost right after this sentence. In the resulting document, they may be placed wherever LaTeX deems most suitable[2].

3 Basic Usage

Listing 1[3] shows the basic code for including an external graphics file inside a `figure` environment and providing a caption and label to go along with it.

---

[1] Because confusion is fun: The `table` environment does not actually typeset a table. That is what `tabular`, `tabularx` and similar environments are for. See [2] for an overview. The `table` environment is merely a floating container intended for containing tabular content.

[2] LaTeX usually tries to place floats either at the top of a page, the bottom of a page, or on a separate page. See Section 4 for more information.

[3] Incidentally, Listing 1 is one of those cases where a new type of floating environment has been provided; in this case by the `minted` package.

Figure Environment with External Picture

```latex
\begin{figure}
    \includegraphics[height=3cm,width=4.5cm]{images/grid8cm.png}
    \caption{This is a distorted grid.}
    \label{fig:distorted-grid}
\end{figure}
```

Listing 1: Code block for including a graphics in a figure and adding a caption and a label so that we can refer to the figure elsewhere in the text. Note that the `\label` command must come either inside or after the `\caption` command.

It is often desirable to center a table or a picture, in which case we add a `\centering` directive into the environment, as done in Listing 2[4].

Tabular Inside Table, Centered

```latex
\begin{table}
    \centering
    \caption{Results for an experiment}
    \label{tab:experiment}
    \begin{tabular}{ll}
        \toprule
        \scshape Experiment Input & \scshape Experiment Output \\
        \midrule
        interesting thing         & interesting result!       \\
        boring thing              & mildly surprising result  \\
        weird thing               & very unexpected result    \\
        fascinating thing         & machine broke             \\
        Xenomorph XX121           & dead scientists           \\
        \bottomrule
    \end{tabular}
\end{table}
```

Listing 2: Centering a `tabular` environment inside a `table` floating environment. This is actually the code for Table 1.

You may have noticed that the `\caption` is placed above the content for tables and below the picture in a `figure` environment. This is not prescribed by LATEX, obviously, and will depend on the style guide you're following or your own preferences. All I will say on the subject is that most tables I've seen had their caption above the table and most images had them below the picture.

What does matter, however, is where the `\label` is put! In order to pick up the correct number, it must always come either inside the `\caption` command to which is is supposed to be connected, or after it. If you put it before the `\caption` command, the `\label` will pick up whichever counter was the last active one before the `\caption`, which can be anything (another picture or table or float of some sort, but also a chapter, section or similar). This is a mistake which is easily made and often hard to detect.

Also note the optional argument to the `\caption` command. If it is given, it will be the text for the entry in the `listof...` command. See Listing 3 and the *List of Listings* on page 2.

---

[4]There exists also a `\begin{center} ...\end{center}` environment. For the curious, some information on the differences between that and `\centering` can be found at [5, 6]

Optional Argument for \caption Command

```
\caption[This is the text for the List of <something> entry]{
    This is the  text which goes below/above the  float. It can be
    rather  long, depending  on how  much explanation  the content
    of  the  float requires  (remember:  because  a float  is  not
    necessarily right  where you  write about  its content  in the
    main  body of  text,  it might  be useful  for  the reader  to
    understand its  content without having  to go and  dig through
    the main text),  and in such cases, it does  not make sense to
    have the entire text of the caption in the List of entry.}
```

Listing 3: Optional Argument to the `\caption` command for the `List of ...` entry. You can see what happens in the *List of Listings* when we do not use this optional argument with the entry for Listing 1. Compare to the *LoL* entry for this listing on page 2, which is short, despite this rather lengty caption text.

56	57	58	59	60	61	62	63
48	49	50	51	52	53	54	55
40	41	42	43	44	45	46	47
32	33	34	35	36	37	38	39
24	25	26	27	28	29	30	31
16	17	18	19	20	21	22	23
8	9	10	11	12	13	14	15
0	1	2	3	4	5	6	7

Figure 2: A classic use case of the `figure` environment: including a graphics file.
You can also have linebreaks inside a `caption`, like here, but it requires the `\protect` command, and there must be no empty line after the \\. See Listing 4 for the code.

Lastly: Because captions are a moving argument (Section 2.2.3, page 22, and Section 3.5.1, page 58 in [1]), fragile commands such as linebreaks inside them must be preceded by `\protect`, as shown in the caption of Figure 2 and the code in Listing 4.

The `\caption` command can only be used inside a floating environment by default (that's how `\caption` knows whether it's supposed to say *Figure n* or *Table m*, for example). If you require captions for non-floating content, there are packages which provide such facilities, as well as more caption customisation options; see [3, 4] and Chapter 7 of this document.

Linebreaks in Captions

```
\begin{figure}
    \includegraphics[height=3cm,width=4.5cm]{images/grid8cm.png}
    \caption{%
        Here we have a sentence.\protect\\
        And this sentence will be on a new line.}
    \label{fig:distorted-grid}
\end{figure}
```

Listing 4: Code block for including a forced linebreak in a caption with a `\protect` command

4 Placement Options

Placement options allow you to influence LATEX's placement behavior of floats, with more or less vehemence.

Whatever your preferences, only use placement options once your text is (almost) complete! Otherwise you will end up needing to change them again and again and again, causing a lot more work. Also, it is quite easily possible to overlook a bad placement option from an earlier version of a document which makes you jump through hoops trying to get the best result even though LATEX would actually do the right thing if you would just let it[5].

> **Placement Options**
>
> `\begin{float type}[loc] body \end{float type}`

Listing 5: Placement options for floats

Listing 5 contains the basic syntax of a float with placement options. The `loc` argument can be a sequence of one to four letters and an exclamation mark. It specifies where LATEX is allowed to put the float. The meaning of these options is as follows (paraphrased from [1]):

h *Here:* at the location in the text where the environment is in the source code. Does not work for double-column floats in two-column documents.

t *Top:* at the top of a text page.

b *Bottom:* at the bottom of a text page. Does not work for double-column floats in two-column documents.

p *Page of floats:* on a separate page containing only floats, but no text.

! *Try harder:* tells LATEX to try harder to place the float at the earliest possible place in the document allowed by the rest of the argument. The meaning of *try harder* is elaborated in Section 6.

This also overrides a potential `\suppressfloats` command (see below) for the float to which the `loc` argument belongs.

The default value for `loc`, if it is not specified manually, is `tpb`, meaning LATEX may put the float at the top of a text page, the bottom of a text page, or on a separate page containing only floats.

When an optional `loc` argument is given, make sure to specify enough options to allow LATEX sufficient flexibility with placing the floats. Otherwise a float and all subsequent floats can end up being saved until the end of a chapter or document, potentially causing TEX to run out of memory or producing a result which does not make sense from a document design point of view.

Additionally, there exists the `\supressfloats[loc]` command. This tells LATEX not to put any additional floats on the current page. In this case, `loc` can be:

t No more figures at the top of the current page.

b No more figures at the bottom of the current page.

---

[5]I am obviously *not* speaking from personal experience here. I am smarter than that, I assure you.

5   Help, My Floats Are Jinxed!

When your floats are just not quite doing what you want them to do, it might be time to have a look at LaTeX's rules which govern the behavior of floats. There are six such rules, backed by fifteen parameters. Because rules are rules, and phrasing matters, I will quote [1] verbatim for these:

> **LaTeX Float Rules from Lamport**
>
> Here are the rules that determine where a figure or table is put:
>
> - It is printed at the earliest place that does not violate subsequent rules, except that an `h` (here) position takes precedence over a `t` (top) position.
> - It will not be printed on an earlier page than the place in the text where the figure or table environment appears.
> - A figure will not be printed before an earlier figure, and a table will not be printed before an earlier table.[a]
> - It may appear only at a position allowed by the `loc` argument, or, if that argument is missing, by the default `tbp` specifier.
> - Placement of the figure or table cannot produce an overfull page.
> - The page constraints determined by the formatting parameters described below are not violated. However, if a `!` appears in the optional argument, then the constraints for text pages are ignored, and only the ones for float pages (expressed by `\floatpagefraction` and `\dblfloatpagefraction`) apply.
>
> The last three rules are suspended when a `\clearpage`, `\cleardoublepage`, or `\end{document}` command occurs, all unprocessed figures and tables being allowed a `p` option and printed at that point.
>
> ---
> [a]However, in a two-column page style, a single-column figure can come before an earlier double-column figure, and vice versa.

Some issues which have caused me the occasional headache over the years are:

- *Many floats, not a lot of text.* There is not really much one can do in this case (I would not advise writing more text just for the sake of padding your document's layout). But it can produce odd results, and some experimentation with the placement of floats in the source code, placement options and `\clearpage` may be needed.

  Float pages tend to be the most sensible option in this case, at least in my humble opinion. Make sure to allow LaTeX to place floats on float pages in this case with the `p` placement option.

- *All floats move to the end of a chapter or the document.* As mentioned above, this tends to come about when not enough placement options are specified for a float (yes, a single one for one float suffices to move all subsequent floats). Usually, I would recommended to at least specify `ht`, see [7].

- *Floats are at an invoncenient place.* Despite its best efforts, sometimes LaTeX's algorithm will simply not produce an optimal result. For example, the text which talks about the content of a float and the corresponding float are inconveniently located. Maybe the reader has to keep flipping a page to read the text for a float and look at the float itself.

  In such cases, I would advise one or more of a few things:

  - Move the descriptive text to the float's caption (with appropriate rephrasing, if needed).

- Try to relocate the float via placement options.
- Try to relocate the float by moving it in the source code.
- Split the float into several pieces of content and have the corresponding text between them. For example, if you have a float with several plots.
- Alternatively, maybe you have several plots which can all be combined into fewer or even a single one.
- Some other genius idea which has not yet occurred to me.

6   LaTeX's Innards

You probably do not have to read this section. But for the curious, or the desperate, these are the fifteen style parameters mentioned above. Note that, according to [1], if you're having trouble, it tends to be caused by one of the first seven more likely than not. Again, I shall quote [1] verbatim:

LaTeX Float Style Parameters from Lamport

`topnumber` A counter whose value is the maximum number of floats allowed at the top of a text page.

`\topfraction` The maximum fraction of the page that can be occupied by floats at the top of the page. Thus, the value `.25` specifies that as much as the top quarter of the page may be devoted to floats. It is changed with `\renewcommand`.

`bottomnumber` Same as `topnumber` except for the bottom of the page.

`\bottomfraction` Same as `\topfraction` except for the bottom of the page.

`totalnumber` A counter whose value is the maximum number of floats that can appear on a single text page, irrespective of their positions.

`\textfraction` The minimum fraction of a text page that must be devoted to text. The other 1- `\textfraction` fraction may be occupied by floats. It is changed with `\renewcommand`.

`\floatpagefraction` The minimum fraction of a float page that must be occu- pied by floats, limiting the amount of blank space allowed on a float page. It is changed with `\renewcommand`.

`dbltopnumber` The analog of topnumber for double-column floats on a two- column page.

`\dbltopfraction` The analog of `\topfraction` for double-column floats on a two-column page.

`\dblfloatpagefraction` The analog of `\floatpagefraction` for a float page of double-column floats.

`\floatsep` The vertical space added between floats that appear at the top or bottom of a text page. It is a rubber length.

`\textfloatsep` The vertical space added between the floats appearing at the top or bottom of a page and the text on that page. It is a rubber length.

`\intextsep` The vertical space placed above and below a float that is put in the middle of the text with the h location option. It is a rubber length.

`\dblfloatsep` The analog of `\floatsep` for double-width floats on a two-col- umn page. It is a rubber length.

`\dbltextfloatsep` The analog of `\textfloatsep` for double-width floats on a two-column page. It is a rubber length.

Changes made to these parameters in the preamble will take effect on the first page. Changes made inside `\begin{document}` ... `\end{document}` will take effect on the following page.

7 Alternatives to Using Floats

It is not difficult when perusing the World Wide Web to find questions, discussions and answers about how to make floats behave in a certain way; particularly placing them *"Right here where I want it!"* seems to be a rather popular wish.

For those who feel so inclined, there exist various solutions and packages, usually via some form of an additional placement option H. See for example the `floatrow` [8] and `float` packages [9] (though these packages offer more than just that).

Personally, I'm not a huge fan of forcing floats to behave like non-floats. In such situations, I'd rather not use a float at all.

As the phrasing implies, this is my personal preference, not official gospel. You may obviously disagree, in which case I would indeed suggest having a look at those packages or other solutions.

7.1 Direct Placement of Content

The most straightforward path to avoid floats is to simply not create a `figure` or `table` environment and put a picture or table right where you want it in the text:

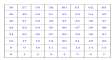| 56	57	58	59	60	61	62	63
48	49	50	51	52	53	54	55
40	41	42	43	44	45	46	47
32	33	34	35	36	37	38	39
24	25	26	27	28	29	30	31
16	17	18	19	20	21	22	23
8	9	10	11	12	13	14	15
0	1	2	3	4	5	6	7

The code for this is:

Placing a Picture at an Arbitrary Location

```
The  most straightforward  one  is to simply not  create  a \verb|figure|  or
\verb|table| environment and  put a picture or table  right where you want it
in the text:

\begin{center}
    \includegraphics[height=3cm,width=4.5cm]{images/grid8cm.png}
\end{center}

the code for this is:
```

The `\begin{center}...\end{center}` is of course optional.

You can even place a picture in the middle of text like it were text itself, as demonstrated here. Though this is admittedly rarely a sensible solution.

> **Placing a Picture inside Text**
>
> ```
> You can even place a picture in the middle of text like it were text itself,
> \includegraphics[height=2em,width=4em]{images/grid8cm.png}  as  demonstrated
> here. Though this is admittedly rarely a sensible solution.
> ```

The same goes for tabular environments:

a	a
a	a

> **Placing a Tabular at an Arbitrary Location**
>
> ```
> The same goes for tabular environments:
>
> \begin{center}
>     \begin{tabular}{ll}
>         \toprule
>         a & a \\
>         a & a \\
>         \bottomrule
>     \end{tabular}
> \end{center}
> ```

If we want to get very silly indeed, we can even do the same trick with tables and put them somewhere in the middle of our text: | a a | a a | . Though really, do not do this.

> **Placing a Tabular inside Text**
>
> ```
> If  we want to  get very  silly indeed,  we can  even do the  same trick with
> tables and put them somewhere in the  middle of our text: \begin{tabular}{ll}
> \toprule a & a \\ a & a\\\bottomrule\end{tabular}.  Though  really, do not do
> this.
> ```

As you have undoubtedly noticed, there are no captions for these pictures and tables. Often, in such cases, they are indeed not needed. Float numbers allow an author to refer to a float and a reader to find it. Captions allow to understand the float without having to have the corresponding main body of text right besides it. So when we insert a picture or a table and place it right where we want it, it does not necessarily make sense to give it a number and a caption, because the information needed to understand the content of the inserted element is right there. Or at least it should be. Referring to the picture, table or other content via a caption and numberic index would be odd and cumbersome in this situation.

7.2   Captions Outside Floats

With that said, we still sometimes want to have numbers and captions. For example, I recently wrote a report where the maximum number of pages was limited. In order to tweak my document

to utilize that limited number of pages to their optimum, LaTeX's floating mechanisms just were not suitable. However, I still wanted to have the basic look and feel of a document which was using floats; the reader was not supposed to notice any difference. I carefully sized and placed my pseudo-floats at the appropriate locations to best utilize the limited page space.

Doing this requires solving two problems:

- How do we create captions outside of floating environments?
- How do we keep the captions and the content to which they refer associated so that they are not split over page breaks?

As always in LaTeX, there are many ways to achieve this. I will present four answers to the first question, and one solution for the second problem.

Having captions oustide of floats can be achieved, among other things, wich the `caption`, `capt-of` and `captdef` packages. The packages `caption` and `capt-of` provide a command `\captionof`, whereas you can define your own commands for each type of float in the `captdef` package.

Non-Float Captions with the `capt-of` or `caption` Packages

```
\includegraphics[width=0.5\textwidth]{images/bestPictureEver.jpeg}
\captionof{figure}{This is a figure caption outside a float.}
```

Non-Float Captions with the `captdef` Package

```
% Syntax: \DeclareCaption{command}{counter}:
\DeclareCaption{\figcaption}{figure}
\DeclareCaption{\tabcaption}{table}
\includegraphics[width=0.5\textwidth]{images/bestPictureEver.jpeg}
\figcaption{This is a figure caption outside a float.}
```

If you are using the `memoir` document class (such as this document), you have a similar mechanism available as offered by the `captdef` package, with additional formatting options:

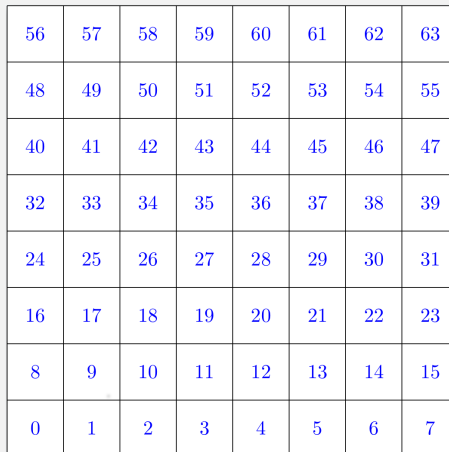**Defining a Non-Float Caption in `memoir`**

```
% Preamble
\newfixedcaption{\figcaption}{figure}
\newfixedcaption{\tabcaption}{table}
\captiontitlefont{\small}
\captionnamefont{\bfseries\small}
\captiondelim{: }
...
\begin{document}
...
\includegraphics[width=0.5\textwidth]{images/bestPictureEver.jpeg}
\figcaption{This is a figure caption outside a float.}
...
```

For more information, consult the documentation of the respective packages.

For creating an unbreakable unit of caption and content, I usually go with a minipage. There are quite a lot of other types of boxes in TeX and LaTeX, but in my experience, minipages have proven most robust for this purpose.

If one were to feel so inclined, one could define a new environment with this or similar code, of course.

7.3   A Tiny Issue

This section has been inspired by the manuals for the `captdef` and `capt-of` packages [10, 11] and appears practically identically in those.

There is one potential problem with using captions in a non-floating context: The commands which are used for the standalone captions increase the respective content type's counter (figure, table, etc.). When mixing floats and non-floats of the same type, for example like so:

If the `figure` does not fit anywhere between where it is in the text and where the non-floating image is included, it can float after the second image, resulting in the figure numbers getting out of order:

```
Mixing Floats and Non-Floats, Figure Numbers Out of Order

<text>
<Figure [n + 1]> % A nice picture!
<more text>
<Figure [n]>      % A magnificent picture!
<yet more text>
```

When using only floats (or only non-floats) this cannot happen, because as per the rules laid out in Section , a float which appears before another float in the source code can never appear after that float in the result. This is also one potential advantage of sticking to floats and using an H parameter, for example via the `floatrow` package (at least as far as I'm aware).

The bottom line is this: Once you take control over where content is placed, beware of dragons.

8 References

[1] Leslie Lamport, Digital Equipment Corporation. "*LᴬTᴇX – A Document Preparation System*", 2nd Edition, 1994, Addison-Wesley Publishing Company.

[2] Lapo Mori, "*Tables in LᴬTᴇX 2ε: Packages and Methods*", The PracTᴇX Journal, 2007-FEB-20. [Online], https://www.tug.org/pracjourn/2007-1/mori/mori.pdf, [Accessed: 2017-MAR-27].

[3] Axel Sommerfeldt. "*Package caption – Customising captions in floating environments*", Version 2016-05-22, 2016-MAY-22. [Online], http://ctan.org/pkg/caption, [Accessed: 2017-MAR-26].

[4] Comprehensive TᴇX Archive Network. "*Topic caption*". [Online], http://ctan.org/topic/caption, [Accessed: 2017-MAR-26].

[5] Enrico Gregorio, "*When should we use \begin{center} instead of \centering?*", [Online], http://tex.stackexchange.com/a/23653, [Accessed: 2017-MAR-26].

[6] stefan, "*TᴇXBlog – center vs. centering*". [Online], http://texblog.net/latex-archive/floats/center-centering/, [Accessed: 2017-MAR-27].

[7] Stefan Kottwitz, "*h float specifier changed to ht warning when not attempting to specify a float*". [Online], http://tex.stackexchange.com/a/1527, [Accessed: 2017-MAR-27].

[8] Olga Lapko. "*floatrow – Modifying the layout of floats*", Version 0.3b, 2009-AUG-02. [Online], http://ctan.org/pkg/float, [Accessed: 2017-MAR-27].

[9] Anselm Lingnau. "*float – Improved interface for floating objects*", Version 1.3d, 2001-NOV-08. [Online], http://ctan.org/pkg/float, [Accessed: 2017-MAR-27].

[10] Robin Fairbairns. "*Capt-of – Captions on more than floats*", 2010-JAN-22. [Online], https://www.ctan.org/pkg/capt-of, [Accessed: 2017-MAR-28].

[11] Robin Fairbairns. "*Captdef – Declare free-standing \caption commands*", 2010-MAR-06. [Online], https://www.ctan.org/pkg/captdef, [Accessed: 2017-MAR-28].