

The figureSeries package*

Thomas Weise

Nature Inspired Computation and Applications Laboratory
USTC-Birmingham Joint Research Institute in Intelligent Computation and Its Applications
School of Computer Science and Technology (SCST)
University of Science and Technology of China (USTC)
West Campus, Science and Technology Building, West Wing, Room 601
Huangshan Road/Feixi Road, Hefei 230027, Anhui, China
tweise@ustc.edu.cn · twaise@gmx.de · <http://www.it-weise.de/>

July 11, 2015

Abstract

This package provides a first working version of a figure*-like construct, which can contain arbitrarily many sub-figures, (somewhat) float, and automatically break over multiple pages if necessary. The most current source code of this package can be found online at <http://github.com/thomasWeise/figureSeries>. Some discussions and additional information may be found at <http://www.it-weise.de>.

Contents

1	Introduction	2
1.1	Addressed Problem and Use Case	2
1.2	Provided Functionality	2
2	Usage	3
2.1	Loading the Package	3
2.2	Provided Macros	3
2.3	Examples	4
3	Implementation	22
3.1	Loading of Required Packages	22
3.2	User Interface Macros	23
3.3	Internal Utility Definitions and Macros	25
3.4	Errors, Tests, and Incompatibilities	31
3.5	Related Work	31
3.6	License	31

*This document corresponds to figureSeries v0.9.4, dated 2015/02/13.

1 Introduction

1.1 Addressed Problem and Use Case

L^AT_EX documents can contain floating objects such as figures or tables. “floating” here means that you insert a figure somewhere in your text and L^AT_EX will find the best place where to put it for you, a place where it fits nicely into the overall layout of the document. Sometimes it is desirable that a figure contains several sub-figures. Let’s say you want to group several diagrams which show related information and which all have the same structure.

In L^AT_EX, a floating object (composed of its contents and potentially a caption) can occupy at most one full page, meaning that it cannot contain any page break. This, in turn, means that the number of (or better the space for) sub-figures that can be hosted inside a figure is limited as well. But what if we have too many sub-figures? What if the sub-figures do not fit on one page, into one floating object, into one `\begin{figure}... \end{figure}`? (Again: The required space includes the space for the sub-figures themselves, their captions, the vertical space between rows of sub-figures, and the caption of the hosting figure.)

We can somewhat solve this issue by splitting the hosting figure into several separate figures, each containing a feasible amount of sub-figures. This needs to be done by hand, because we need to compile the document and fix and compile and fix and so on, since we usually not know beforehand how many sub-figures fit into one figure. This is not nice.

Problems may occur when our L^AT_EX document is created automatically and has an *a priori* unknown number of sub-figures in a figure. There is no way to automatically determine how many sub-figures we can place into a figure. Even if we would know the sub-figure sizes, we cannot compute the space occupied by their captions beforehand (well, without replicating L^AT_EX, that is...). Trust me, I have tried this (in the software project *TSPSuite* [1]). Similar use cases have been reported in [2, 3].

Also, L^AT_EX is limited in terms of the number of floating objects it can handle, I believe the limit is 18 and can be increased using the package `morefloats` [4], but this is a true problem when creating many sub-figures automatically. You may have so many sub-figures that splitting them into multiple (hosting) figures leads to too many (hosting) figures. Or you just end up with too many floating objects which are all laid out at once at the end of a chapter or something or which otherwise make your text layout look odd and unbalanced.

1.2 Provided Functionality

This package provides

1. a facility to include an arbitrary number of (potentially differently-sized) sub-figures into a figure*-like construct,
2. the ability to make this figure*-like construct look as if it was a floating object, which

3. works well in both one-column and two-column documents.

WARNING: This is just a first, hacked-together version of this package. I am not an expert in \LaTeX , and I am pretty sure I managed to incorporate a few “that cannot be made to work reliable”s into the code. Plus, it contains some code taken from the internet, which I myself only partially understand (see Section 3.3.5). This package here is just there for discussion. Use it at your own risk.

2 Usage

2.1 Loading the Package

Load this package using

```
\RequirePackage{figureSeries}
```

This will automatically load the packages `caption` [5], `subcaption` [6], and `afterpage` [7].

2.2 Provided Macros

Here we discuss the macros that can directly be accessed by the user to make use of the package’s functionality. The implementation of these macros is given in Section 3.2 and several examples can be found in Section 2.3.

`\figureSeriesElement`

The macro `\figureSeriesElement{<caption>}{<contents>}` inserts an element of the figure series, i.e., one sub-figure. Its first argument is the caption of the element and may contain a `\label`. The second argument is the graphic to print. It could, e.g., be a call to `\includegraphic` from the `graphicx` package [8].

`\figureSeriesRow`

`\figureSeriesElement{<contents>}` inserts a new row of elements (sub-figures) into the figure series. Its single argument should thus contain a sequence of `\figureSeriesElements`. As a consequence of this architecture, each sub-figure belongs to one row and no sub-figure can span multiple rows.

`\figureSeriesHere`

The macro `\figureSeriesHere{<caption>}{<contents>}` tries to insert a (non-floating) figure series at the current position in the document. This means that it may begin wherever, well, it is used, e.g., in the middle of the page.

The macro has two mandatory parameters, the caption and the contents. The caption will be put at the beginning of the figure series, which is different from the normal behavior of captions in `\begin{figure*}... \end{figure*}` or `\begin{figure}... \end{figure}`. The reason is that a figure series may span over multiple pages and having the caption at the end may be awkward and confusing. The caption text may contain a `\label`.

The contents of a figure series should be a sequence of `\figureSeriesRow` calls.

Since figure series are page-wide elements, starting them in the middle of the page only works in one-column documents. In two-column documents, any figure series will behave as specified in macro `\figureSeriesFloat` below.

`\figureSeriesFloat`

The macro `\figureSeriesFloat{<caption>}{<contents>}` macro takes the

same parameters as `\figureSeriesHere`, but has a float-like behavior. By using the `\afterpage` command of `afterpage` package [7], we let it start at the following page. This is different from L^AT_EX’ normal floating behavior, but as good as we can get with page-breaking objects, I think.

2.3 Examples

Here we provide a set of examples for the use of the package. Each example demonstrates another facet of the package and, at the same time, serves as test case. Instead of using `\includegraphic`, we simply stretch single letters via `\resizebox` and use them “sub-figures”. This is good enough to see how the layout works and allows us to generate arbitrarily-sized placeholders for figures.

In order to create some placeholder text in the examples, we use the `lipsum` command from package `lipsum` [9], which prints pseudo-Latin text known as “Lorem Ipsum” (see http://en.wikipedia.org/wiki/Lorem_ipsum).

2.3.1 Non-Floating Figure Series in Single-Column Document

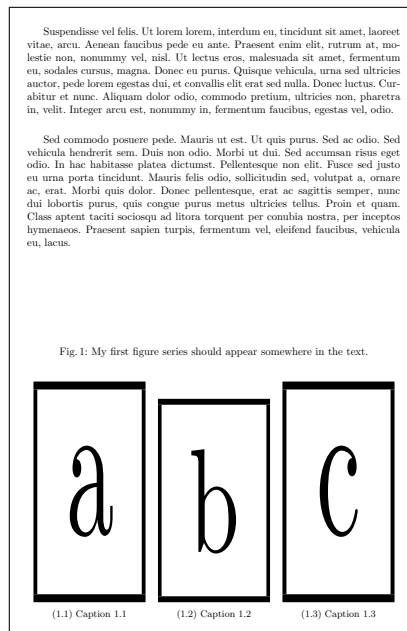
In Example 1 we place a non-floating figure series consisting of two rows of figures into a one-column document using Springer’s document class `llncs` [10]. The result can be seen in Figure 1 and compared with a floating version in Figure 7, which represents the floating version of this example (given in Example 7).

Example 1 An example using the one-column `llncs` class, rendered as Figure 1.

```
\documentclass{llncs}%
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum%
\figureSeriesHere{%
My first figure series should appear somewhere in the text.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 1.1}{\resizebox{0.3\linewidth}{5cm}{\fbox{a}}}%
\figureSeriesElement{Caption 1.2}{\resizebox{0.3\linewidth}{5cm}{\fbox{b}}}%
\figureSeriesElement{Caption 1.3}{\resizebox{0.3\linewidth}{5cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 1.4}{\resizebox{0.3\linewidth}{5cm}{\fbox{d}}}%
\figureSeriesElement{Caption 1.5}{\resizebox{0.3\linewidth}{5cm}{\fbox{e}}}%
\figureSeriesElement{Caption 1.6}{\resizebox{0.3\linewidth}{5cm}{\fbox{f}}}%
}%
\lipsum%
\end{document}%
```



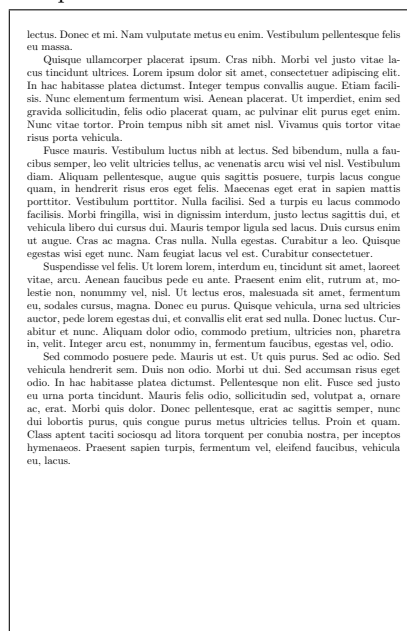
(1.1) Page 1 of the pdf compiled from Example 1.



(1.2) Page 2 of the pdf compiled from Example 1.



(1.3) Page 3 of the pdf compiled from Example 1.



(1.4) Page 4 of the pdf compiled from Example 1.

Figure 1: The rendered result of Example 1 (with trimmed page margins): A `figureSeries` starts at the bottom of a page and extends to the top of the next page.

2.3.2 Floating Figure Series in Two-Column Document

We now put a floating figure series into a two-column document using the `IEEEtran` [11] class in Example 2. This new figure series has five rows of sub-figures and should span over multiple pages. The two-column text continues directly after the figure series. The rendered results of this example are given in Figure 2.

2.3.3 Coalescing Figure Series in Two-Column Document

In Example 3, we put several floating figure series close to each other into a two-column document, again using the `IEEEtran` class [11]. The bodies of the figure series should coalesce without losing their captions, figure numbers, or identities. Since they coalesce, no empty pages are produced in between (see Section 3.3.4.F). The result is rendered as Figure 3.

Example 2 An example using the two-column IEEEtran class, rendered as Figure 2.

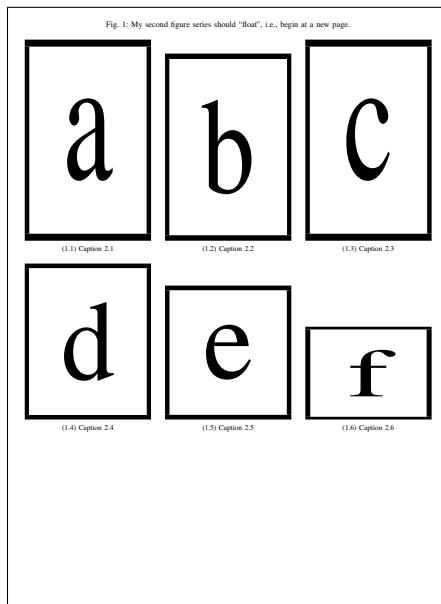
```

\documentclass{IEEEtran}%
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum%
\figureSeriesFloat{%
My second figure series should ‘float’, i.e., begin at a new page.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 2.1}{\resizebox{0.3\linewidth}{6cm}{\fbox{a}}}%
\figureSeriesElement{Caption 2.2}{\resizebox{0.3\linewidth}{6cm}{\fbox{b}}}%
\figureSeriesElement{Caption 2.3}{\resizebox{0.3\linewidth}{6cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 2.4}{\resizebox{0.3\linewidth}{5cm}{\fbox{d}}}%
\figureSeriesElement{Caption 2.5}{\resizebox{0.3\linewidth}{4cm}{\fbox{e}}}%
\figureSeriesElement{Caption 2.6}{\resizebox{0.3\linewidth}{3cm}{\fbox{f}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 2.7}{\resizebox{0.3\linewidth}{!}{\fbox{g}}}%
\figureSeriesElement{Caption 2.8}{\resizebox{0.3\linewidth}{!}{\fbox{h}}}%
\figureSeriesElement{Caption 2.9}{\resizebox{0.3\linewidth}{!}{\fbox{i}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 2.10}{\resizebox{0.3\linewidth}{3cm}{\fbox{j}}}%
\figureSeriesElement{Caption 2.11}{\resizebox{0.3\linewidth}{2cm}{\fbox{k}}}%
\figureSeriesElement{Caption 2.12}{\resizebox{0.3\linewidth}{2.5cm}{\fbox{l}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 2.13}{\resizebox{0.3\linewidth}{!}{\fbox{m}}}%
\figureSeriesElement{Caption 2.14}{\resizebox{0.3\linewidth}{6cm}{\fbox{n}}}%
\figureSeriesElement{Caption 2.15}{\resizebox{0.3\linewidth}{4cm}{\fbox{o}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 2.16}{\resizebox{0.3\linewidth}{3cm}{\fbox{p}}}%
\figureSeriesElement{Caption 2.17}{\resizebox{0.3\linewidth}{3cm}{\fbox{q}}}%
\figureSeriesElement{Caption 2.18}{\resizebox{0.3\linewidth}{3cm}{\fbox{r}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 2.19}{\resizebox{0.3\linewidth}{3cm}{\fbox{s}}}%
\figureSeriesElement{Caption 2.20}{\resizebox{0.3\linewidth}{3cm}{\fbox{t}}}%
\figureSeriesElement{Caption 2.21}{\resizebox{0.3\linewidth}{3cm}{\fbox{u}}}%
}%
\lipsum[1-20]%
\end{document}%

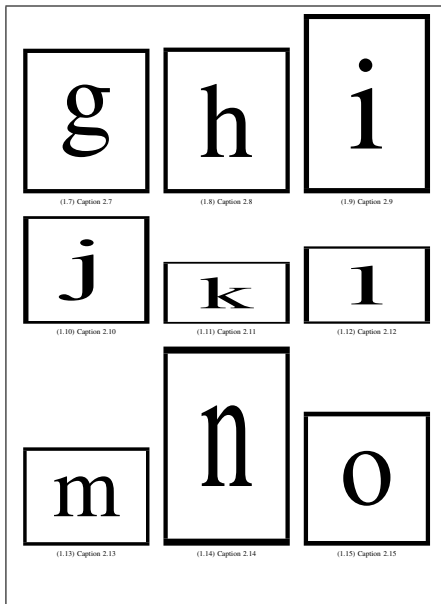
```



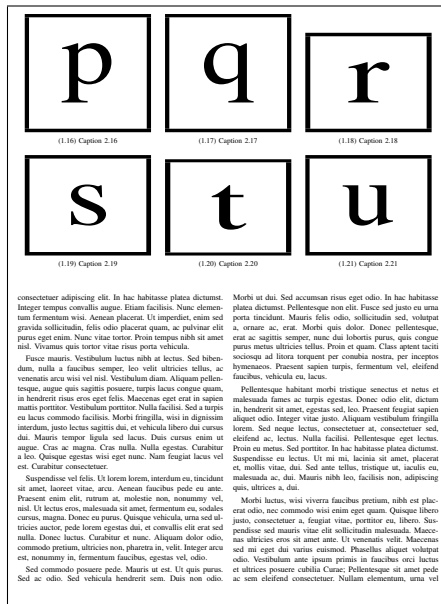
(2.1) Page 1 of the pdf compiled from Example 2.



(2.2) Page 2 of the pdf compiled from Example 2.



(2.3) Page 3 of the pdf compiled from Example 2.



(2.4) Page 4 of the pdf compiled from Example 2.

Figure 2: The rendered result of Example 2 (with trimmed page margins): A floating `figureSeries` in two-column mode.

Example 3 An example using the two-column `IEEEtran` class and two coalescing figure series, rendered as Figure 3.

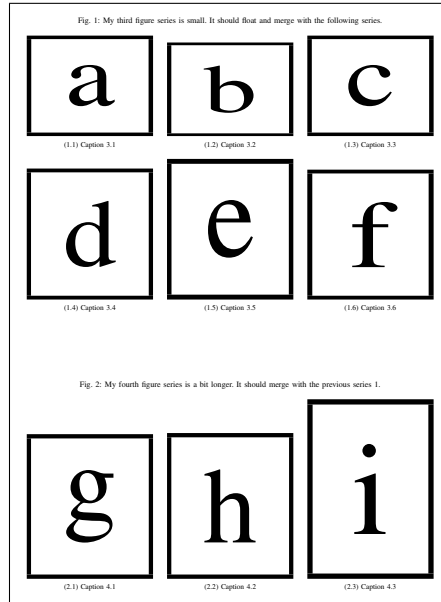
```

\documentclass{IEEEtran}%
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum%
\figureSeriesFloat{%
My third figure series is small. It should float and merge with the following
series.\label{ser:3}%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 3.1}{\resizebox{0.3\linewidth}{3cm}{\fbox{a}}}%
\figureSeriesElement{Caption 3.2}{\resizebox{0.3\linewidth}{3cm}{\fbox{b}}}%
\figureSeriesElement{Caption 3.3}{\resizebox{0.3\linewidth}{3cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 3.4}{\resizebox{0.3\linewidth}{4.2cm}{\fbox{d}}}%
\figureSeriesElement{Caption 3.5}{\resizebox{0.3\linewidth}{4.2cm}{\fbox{e}}}%
\figureSeriesElement{Caption 3.6}{\resizebox{0.3\linewidth}{4.2cm}{\fbox{f}}}%
}%
This is a short text inbetween the two series~\ref{ser:3} and~\ref{ser:4},
which should not prevent them from merging.
\figureSeriesFloat{%
My fourth figure series is a bit longer. It should merge with the previous
series~\ref{ser:3}.\label{ser:4}%
}{
\figureSeriesRow{%
\figureSeriesElement{Caption 4.1}{\resizebox{0.3\linewidth}{!}{\fbox{g}}}%
\figureSeriesElement{Caption 4.2}{\resizebox{0.3\linewidth}{!}{\fbox{h}}}%
\figureSeriesElement{Caption 4.3}{\resizebox{0.3\linewidth}{!}{\fbox{i}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 4.4 is a very long caption. It will probably %
break into multiple lines. Does this matter? Let's find out.}%
\resizebox{0.3\linewidth}{3cm}{\fbox{j}}}%
\figureSeriesElement{Caption 4.5}{\resizebox{0.3\linewidth}{2cm}{\fbox{k}}}%
\figureSeriesElement{Caption 4.6}{\resizebox{0.3\linewidth}{2.5cm}{\fbox{l}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 4.7}{\resizebox{0.3\linewidth}{!}{\fbox{m}}}%
\figureSeriesElement{Caption 4.8}{\resizebox{0.3\linewidth}{6cm}{\fbox{n}}}%
\figureSeriesElement{Caption 4.9}{\resizebox{0.3\linewidth}{4cm}{\fbox{o}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 4.10}{\resizebox{0.3\linewidth}{3cm}{\fbox{p}}}%
\figureSeriesElement{Caption 4.11}{\resizebox{0.3\linewidth}{3cm}{\fbox{q}}}%
\figureSeriesElement{Caption 4.12}{\resizebox{0.3\linewidth}{3cm}{\fbox{r}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 4.13}{\resizebox{0.3\linewidth}{3cm}{\fbox{s}}}%
\figureSeriesElement{Caption 4.14}{\resizebox{0.3\linewidth}{3cm}{\fbox{t}}}%
\figureSeriesElement{Caption 4.15}{\resizebox{0.3\linewidth}{3cm}{\fbox{u}}}%
}%
\lipsum[1-20]%
\end{document}%

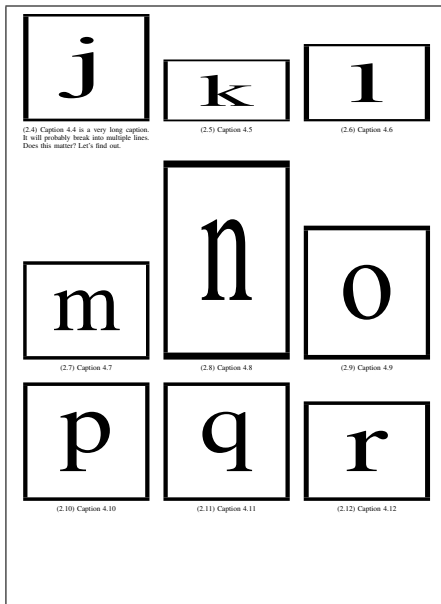
```



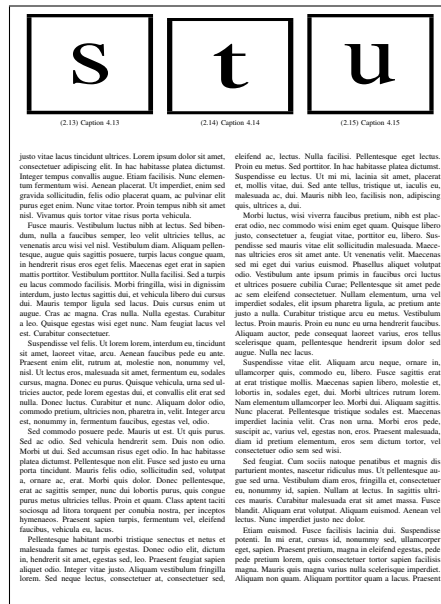
(3.1) Page 1 of the pdf compiled from Example 3.



(3.2) Page 2 of the pdf compiled from Example 3.



(3.3) Page 3 of the pdf compiled from Example 3.



(3.4) Page 4 of the pdf compiled from Example 3.

Figure 3: The rendered result of Example 3 (with trimmed page margins): Two floating `figureSeries` in two-column mode are coalesced, without using their caption and identities.

2.3.4 Two-Column Document with sig-alternate

In the following Example 4, we test the `figureSeries` together for documents using ACM's `sig-alternate` [12] document class.

2.3.5 Many Small Sub-Figures in Two-Column Document

In the following Example 5 (again based on ACM's `sig-alternate` [12] document class), we put many small sub-figures into a figure. Also, the last paragraph of the text in the example is a reference to one of the sub-figures. The results are rendered as Figure 5.

2.3.6 Two figureSeries Separated by Text in Two-Column Document

In the following Example 6 (again based on ACM's `sig-alternate` [12] document class), we put two `figureSeries` which are separated by text. The results are rendered as Figure 6.

2.3.7 One Floating figureSeries in a Single-Column Document

In the following Example 7 (based on Springer's `llncs` [10] document class), we let the figure series from Example 1 float. You can compare the rendered in Figure 7 with those in Figure 1.

2.3.8 Mixed figureSeries, figure, and figure*

In Example 8, we mix `figureSeries` with `figure` and `figure*` environments. Something like this would cause ! `LaTeX Error: Float(s) lost.` errors in version 0.9.2 of our package. Due to some modifications, this particular document now compiles, but the errors still appear in other documents.

We again use the `IEEEtran` class [11]. The result is rendered as Figure 8. As you can see, the ordering of the figures in the current version of our package (0.9.4) is far from perfect, but at least the document compiles. In a productive environment, you would have more text between your figures, so hopefully, such layout problems would not occur.

Example 4 An example featuring the two-column `sig-alternate` class. The results are rendered as Figure 4.

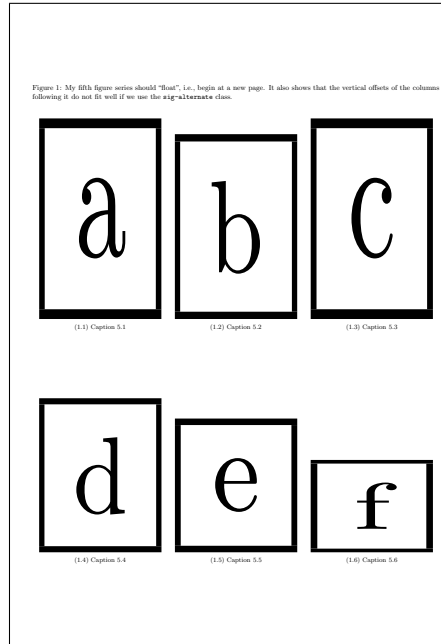
```

\documentclass{sig-alternate}
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum[1]%
\figureSeriesFloat{%
My fifth figure series should ‘float’, i.e., begin at a new page. It
also shows that the vertical offsets of the columns following it do not fit
well if we use the \texttt{sig-alternate} class.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 5.1}{\resizebox{0.3\linewidth}{6cm}{\fbox{a}}}%
\figureSeriesElement{Caption 5.2}{\resizebox{0.3\linewidth}{6cm}{\fbox{b}}}%
\figureSeriesElement{Caption 5.3}{\resizebox{0.3\linewidth}{6cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 5.4}{\resizebox{0.3\linewidth}{5cm}{\fbox{d}}}%
\figureSeriesElement{Caption 5.5}{\resizebox{0.3\linewidth}{4cm}{\fbox{e}}}%
\figureSeriesElement{Caption 5.6}{\resizebox{0.3\linewidth}{3cm}{\fbox{f}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 5.7}{\resizebox{0.3\linewidth}{!}{\fbox{g}}}%
\figureSeriesElement{Caption 5.8}{\resizebox{0.3\linewidth}{!}{\fbox{h}}}%
\figureSeriesElement{Caption 5.9}{\resizebox{0.3\linewidth}{!}{\fbox{i}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 5.10}{\resizebox{0.3\linewidth}{3cm}{\fbox{j}}}%
\figureSeriesElement{Caption 5.11}{\resizebox{0.3\linewidth}{2cm}{\fbox{k}}}%
\figureSeriesElement{Caption 5.12}{\resizebox{0.3\linewidth}{5.5cm}{\fbox{l}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 5.13}{\resizebox{0.3\linewidth}{!}{\fbox{m}}}%
\figureSeriesElement{Caption 5.14}{\resizebox{0.3\linewidth}{6cm}{\fbox{n}}}%
\figureSeriesElement{Caption 5.15}{\resizebox{0.3\linewidth}{4cm}{\fbox{o}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 5.16}{\resizebox{0.3\linewidth}{3cm}{\fbox{p}}}%
\figureSeriesElement{Caption 5.17}{\resizebox{0.3\linewidth}{3cm}{\fbox{q}}}%
\figureSeriesElement{Caption 5.18}{\resizebox{0.3\linewidth}{3cm}{\fbox{r}}}%
}%
\lipsum[1-20]%
\end{document}%

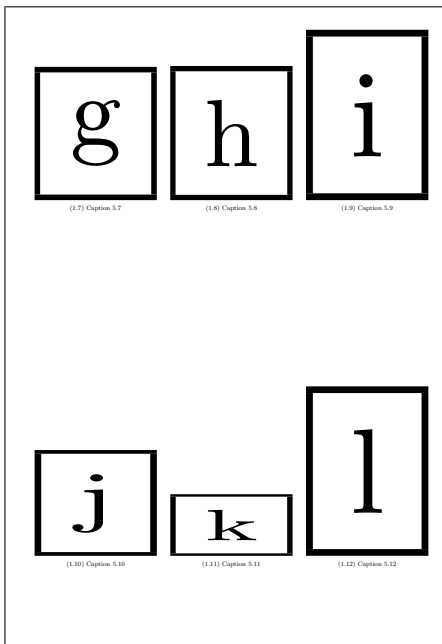
```



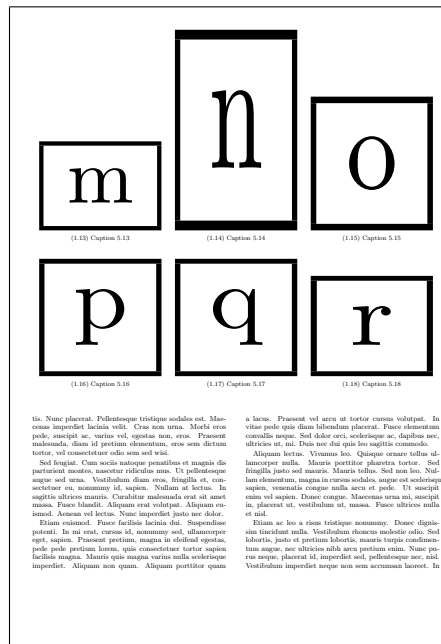
(4.1) Page 1 of the pdf compiled from Example 4.



(4.2) Page 2 of the pdf compiled from Example 4.



(4.3) Page 3 of the pdf compiled from Example 4.



(4.4) Page 4 of the pdf compiled from Example 4.

Figure 4: The rendered result of Example 4 (with trimmed page margins): It works for sig-alternate too.

Example 5 An example featuring the two-column `sig-alternate` class with a reference to sub-figure. The results are rendered as Figure 5.

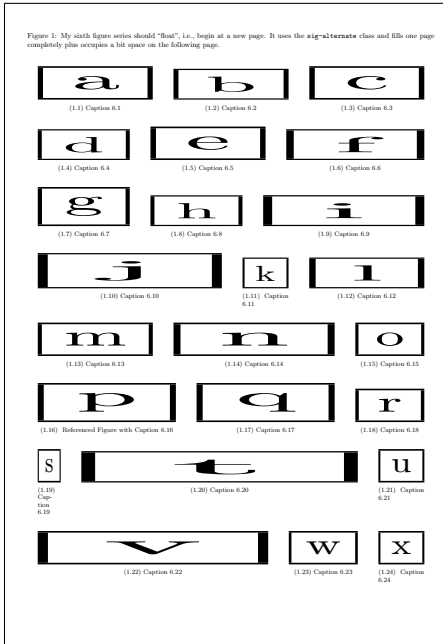
```

\documentclass{sig-alternate}
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum[1]%
\figureSeriesFloat{%
My sixth figure series should ‘float’, i.e., begin at a new page.
It uses the \texttt{sig-alternate} class and fills one page completely
plus occupies a bit space on the following page.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 6.1}{\resizebox{5cm}{1cm}{\fbox{a}}}%
\figureSeriesElement{Caption 6.2}{\resizebox{5cm}{1cm}{\fbox{b}}}%
\figureSeriesElement{Caption 6.3}{\resizebox{5cm}{1cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.4}{\resizebox{4cm}{1cm}{\fbox{d}}}%
\figureSeriesElement{Caption 6.5}{\resizebox{5cm}{1cm}{\fbox{e}}}%
\figureSeriesElement{Caption 6.6}{\resizebox{6cm}{1cm}{\fbox{f}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.7}{\resizebox{4cm}{1cm}{\fbox{g}}}%
\figureSeriesElement{Caption 6.8}{\resizebox{4cm}{1cm}{\fbox{h}}}%
\figureSeriesElement{Caption 6.9}{\resizebox{7cm}{1cm}{\fbox{i}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.10}{\resizebox{8cm}{1cm}{\fbox{j}}}%
\figureSeriesElement{Caption 6.11}{\resizebox{2cm}{1cm}{\fbox{k}}}%
\figureSeriesElement{Caption 6.12}{\resizebox{5cm}{1cm}{\fbox{l}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.13}{\resizebox{5cm}{1cm}{\fbox{m}}}%
\figureSeriesElement{Caption 6.14}{\resizebox{7cm}{1cm}{\fbox{n}}}%
\figureSeriesElement{Caption 6.15}{\resizebox{3cm}{1cm}{\fbox{o}}}%
}\figureSeriesRow{%
\figureSeriesElement{\label{checkThis}Referenced Figure with Caption 6.16}{%
\resizebox{6cm}{1cm}{\fbox{p}}}%
\figureSeriesElement{Caption 6.17}{\resizebox{6cm}{1cm}{\fbox{q}}}%
\figureSeriesElement{Caption 6.18}{\resizebox{3cm}{1cm}{\fbox{r}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.19}{\resizebox{1cm}{1cm}{\fbox{s}}}%
\figureSeriesElement{Caption 6.20}{\resizebox{12cm}{1cm}{\fbox{t}}}%
\figureSeriesElement{Caption 6.21}{\resizebox{2cm}{1cm}{\fbox{u}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.22}{\resizebox{10cm}{1cm}{\fbox{v}}}%
\figureSeriesElement{Caption 6.23}{\resizebox{3cm}{1cm}{\fbox{w}}}%
\figureSeriesElement{Caption 6.24}{\resizebox{2cm}{1cm}{\fbox{x}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 6.25}{\resizebox{7.5cm}{1cm}{\fbox{y}}}%
\figureSeriesElement{Caption 6.26}{\resizebox{7.5cm}{1cm}{\fbox{z}}}%
}%
\lipsum[1-20]\par%
\huge{Dude, check Figure~\ref{checkThis}.}%
\end{document}%

```



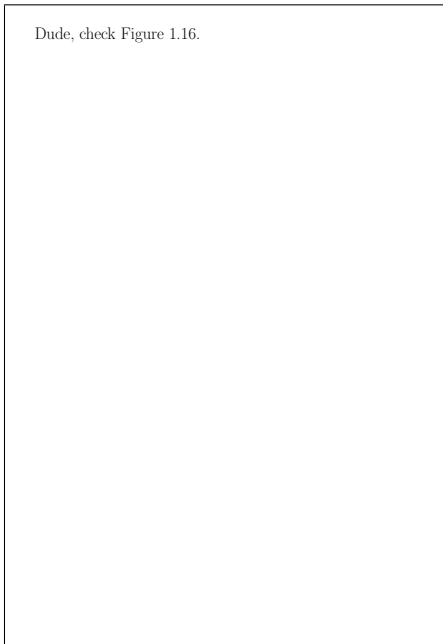
(5.1) Page 1 of the pdf compiled from Example 5.



(5.2) Page 2 of the pdf compiled from Example 5.



(5.3) Page 3 of the pdf compiled from Example 5.



(5.4) Page 4 of the pdf compiled from Example 5.

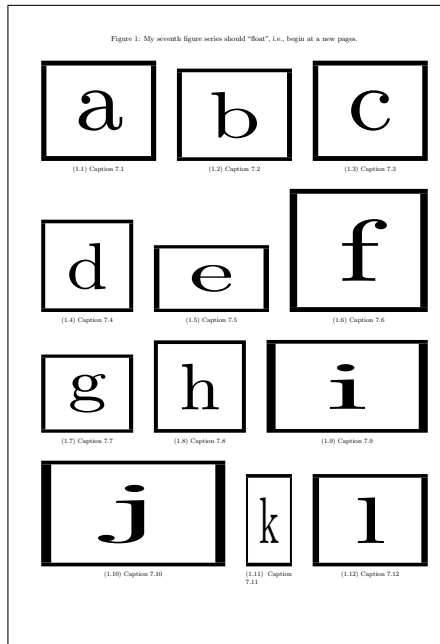
Figure 5: The rendered result of Example 5 (with trimmed page margins): Many small sub-figures nicely fill a page.

Example 6 An example featuring two figure series separated by text. The results are rendered as Figure 6.

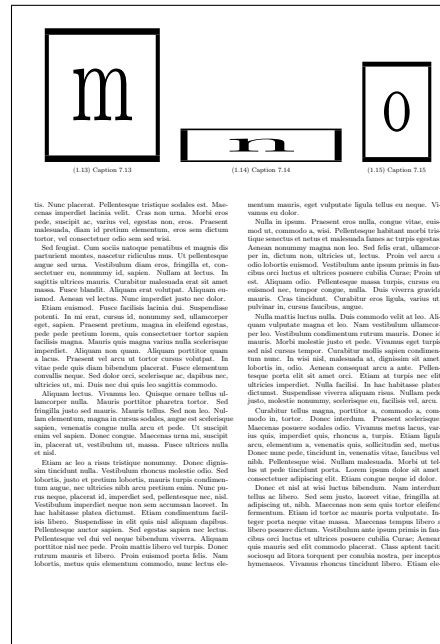
```

\documentclass{sig-alternate}
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum[1]%
\figureSeriesFloat{%
My seventh figure series should ‘‘float’’, i.e., begin at a new pages.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 7.1}{\resizebox{5cm}{3cm}{\fbox{a}}}%
\figureSeriesElement{Caption 7.2}{\resizebox{5cm}{3cm}{\fbox{b}}}%
\figureSeriesElement{Caption 7.3}{\resizebox{5cm}{3cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 7.4}{\resizebox{4cm}{3cm}{\fbox{d}}}%
\figureSeriesElement{Caption 7.5}{\resizebox{5cm}{2cm}{\fbox{e}}}%
\figureSeriesElement{Caption 7.6}{\resizebox{6cm}{4cm}{\fbox{f}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 7.7}{\resizebox{4cm}{2cm}{\fbox{g}}}%
\figureSeriesElement{Caption 7.8}{\resizebox{4cm}{3cm}{\fbox{h}}}%
\figureSeriesElement{Caption 7.9}{\resizebox{7cm}{3cm}{\fbox{i}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 7.10}{\resizebox{8cm}{3cm}{\fbox{j}}}%
\figureSeriesElement{Caption 7.11}{\resizebox{2cm}{3cm}{\fbox{k}}}%
\figureSeriesElement{Caption 7.12}{\resizebox{5cm}{3cm}{\fbox{l}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 7.13}{\resizebox{5cm}{4cm}{\fbox{m}}}%
\figureSeriesElement{Caption 7.14}{\resizebox{7cm}{1cm}{\fbox{n}}}%
\figureSeriesElement{Caption 7.15}{\resizebox{3cm}{3cm}{\fbox{o}}}%
}%
\lipsum[1-30]%
\figureSeriesFloat{%
My eighth figure series also ‘‘floats’’. It comes after the seventh.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 8.1}{\resizebox{0.3\linewidth}{6cm}{\fbox{a}}}%
\figureSeriesElement{Caption 8.2}{\resizebox{0.3\linewidth}{6cm}{\fbox{b}}}%
\figureSeriesElement{Caption 8.3}{\resizebox{0.3\linewidth}{6cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 8.4}{\resizebox{0.3\linewidth}{5cm}{\fbox{d}}}%
\figureSeriesElement{Caption 8.5}{\resizebox{0.3\linewidth}{4cm}{\fbox{e}}}%
\figureSeriesElement{Caption 8.6}{\resizebox{0.3\linewidth}{3cm}{\fbox{f}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 8.7}{\resizebox{0.3\linewidth}{!}{\fbox{g}}}%
\figureSeriesElement{Caption 8.8}{\resizebox{0.3\linewidth}{!}{\fbox{h}}}%
\figureSeriesElement{Caption 8.9}{\resizebox{0.3\linewidth}{!}{\fbox{i}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 8.10}{\resizebox{0.3\linewidth}{3cm}{\fbox{j}}}%
\figureSeriesElement{Caption 8.11}{\resizebox{0.3\linewidth}{2cm}{\fbox{k}}}%
\figureSeriesElement{Caption 8.12}{\resizebox{0.3\linewidth}{8.5cm}{\fbox{l}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 8.13}{\resizebox{0.3\linewidth}{!}{\fbox{m}}}%
\figureSeriesElement{Caption 8.14}{\resizebox{0.3\linewidth}{6cm}{\fbox{n}}}%
\figureSeriesElement{Caption 8.15}{\resizebox{0.3\linewidth}{4cm}{\fbox{o}}}%
}%
\lipsum[1-30]%
\end{document}%

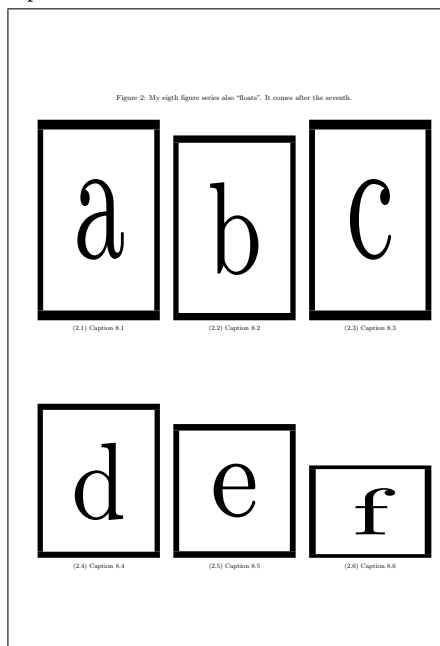
```



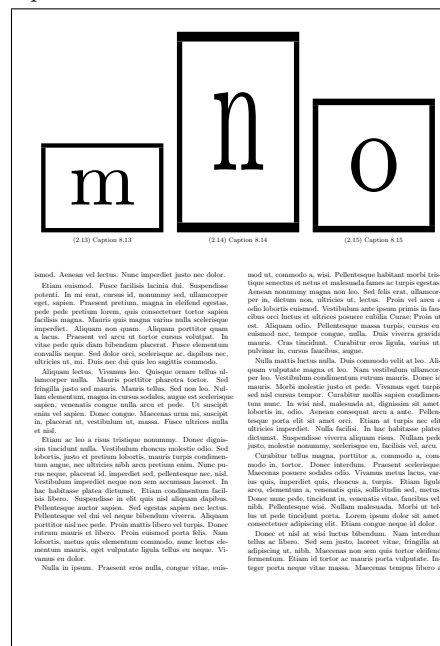
(6.1) Page 2 of the pdf compiled from Example 6.



(6.2) Page 3 of the pdf compiled from Example 6.



(6.3) Page 6 of the pdf compiled from Example 6.



(6.4) Page 8 of the pdf compiled from Example 6.

Figure 6: The rendered result of Example 6 (with trimmed page margins): Two figureSeries are separated by text.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur 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 eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultrices tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

(7.1) Page 1 of the pdf compiled from Example 7.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

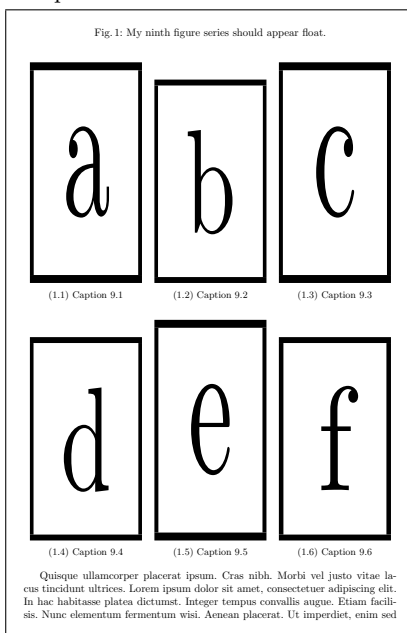
Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur 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 eu enim. Vestibulum pellentesque felis eu massa.

(7.2) Page 2 of the pdf compiled from Example 7.



(7.3) Page 3 of the pdf compiled from Example 7.

gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultrices tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

(7.4) Page 4 of the pdf compiled from Example 7.

Figure 7: The rendered result of Example 7 (with trimmed page margins): A `figureSeries` floating in one-column mode (see Example 1 for the non-floating version).

Thomas Weise

e

Fig. 3: Sodales condimentum nascetur tempus

in hendecris risus eros eget felix. Maecenas eget erat in sapien-
tiam mattis portitor. Vestibulum portitor. Nulla facilisi. Sed a turpi-
eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim
interdum, justo lectus sagittis dui, et vehicula libero dui cursus
dui. Mauris tempor ligula sed lacus. Duis cursus enim u-
augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur
a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus ve-
est. Curabitur conseteuener.

Suspensioe vellet. ar. Aorem lozem, interdum ex, tuncdum tunc
sit amet, laoreet vitae, arcu. Lorem faucibus pede ex ante
Praesent enim elit, rutrum at, molestie non, nonummy vel
nisl. Ut lectus eros, malesuada sit amet, fermentum ex, sodales
curcuis, magna. Donec ex purus. Quisque vehicula, urna sed
tricies auctor, pede lorem egestas dui, et convallis elit erat
nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio
commodo pretium, ultricies non, pharetra in, vellet. Integer arcu
est, nonummy in, fermentum faucibus, egestas vel, odio.

Se commodo posuere pede. Mauris ut est. Ut quis purus
Sed ac odio. Sed vehicula hendrerit sem. Duis non odio
Morbi ut dictum. Sed accumsan risus eget odio. In hac habitabit
platea dictumst. Pellentesque non elit. Fusce sed justo eu urna
porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat
a, ornare ac, erat. Mauris quis dolor. Donec pellentesque
erat ac sagittis semper, nunc dicti lobortis purus, quis congue
purus metus ultricies tellus. Proin et quam. Class aptent tacit
sociosqu ad litora torquent per conubia nostra, per inceptos
hymenaeos. Praesent sapien turpis, fermentum vel, eleifend
faucibus, vehicula eu, lacus.

Ehim ac le a rissu tristiqe nonumy. Donec digni
 sint tuncid nullu. Vestibulum rhoncus molestie odio. Seco
 lobortis, justo et pretium lobortis, mauris turpis condimentu
 augue, nec ultricies nibb arcu pretium enim. Nunc purus
 neque, placerat id, imperdiet sed, pellentesque nec, nisl
 Vestibulum imperdiet neque non sem accumsan laoreet. In hac
 habetasse platea dictumst. Ehim condimentum facilis libero
 Suspendisse in elit quis nisl aliquam dapibus. Pellentesque
 auctor sapien. Sed egestas sapien nec lectus. Pellentesque ve
 dui vel neque bibendum viverra. Aliquam portitor nisl neq

a
b

Fig. 1: Interdum primis ultrices augue

f

Fig. 4: Soluta autem nascetur litora

(8.2) Pa
ample 8.

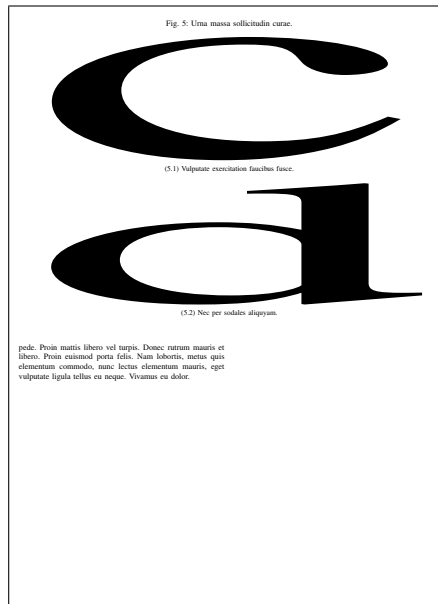


Fig. 5: Urna massa sollicitudin curae

(5.1) Vulputate exercitatio fascibus fasci

(5.2) Nec per sodales aliququam

pede. Proin mattis libero vel turpis. Donec rutrum mauris e libero. Proin euismod porta felis. Nam lobortis, metus qui elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Figure 8 (continued)

Figure 8: The rendered result of Example 8 (with trimmed page margins): a mixture of `figureSeries`, `figure`, and `figure*`.

Example 7 An example using the one-column `llncs` class, rendered as Figure 7.

```

\documentclass{llncs}%
\RequirePackage{graphicx}%
\RequirePackage{lipsum}%
\RequirePackage{figureSeries}%
\begin{document}%
\lipsum%
\figureSeriesFloat{%
My ninth figure series should appear float.%
}%
\figureSeriesRow{%
\figureSeriesElement{Caption 9.1}{\resizebox{0.3\linewidth}{5cm}{\fbox{a}}}%
\figureSeriesElement{Caption 9.2}{\resizebox{0.3\linewidth}{5cm}{\fbox{b}}}%
\figureSeriesElement{Caption 9.3}{\resizebox{0.3\linewidth}{5cm}{\fbox{c}}}%
}\figureSeriesRow{%
\figureSeriesElement{Caption 9.4}{\resizebox{0.3\linewidth}{5cm}{\fbox{d}}}%
\figureSeriesElement{Caption 9.5}{\resizebox{0.3\linewidth}{5cm}{\fbox{e}}}%
\figureSeriesElement{Caption 9.6}{\resizebox{0.3\linewidth}{5cm}{\fbox{f}}}%
}%
\lipsum%
\end{document}%

```

Example 8 An example for a mixture of `figureSeries`, `figure`, and `figure*`, Figure 8.

```

\documentclass{IEEEtran}%
\RequirePackage{lipsum}%
\RequirePackage{graphicx}%
\RequirePackage{figureSeries}%
%
\begin{document}%
%
\title{Test Example: Mixed figure, figure*, and figureSeries}%
\author{Thomas Weise}%
\maketitle%
%
\begin{figure*}%
\resizebox{0.19\linewidth}{0.12\textheight}{a}%
\caption{Interdum primis ultrices augue.}%
\end{figure*}
%
\lipsum[1-7]%
%
\begin{figure*}%
\resizebox{0.19\linewidth}{0.15\textheight}{b}%
\caption{Inceptos eget varius curae.}%
\end{figure*}%
%
\figureSeriesFloat{Urna massa sollicitudin curae.}{%
\figureSeriesRow{%
\figureSeriesElement{Vulputate exercitation faucibus fusce.}{%
\resizebox{0.95\linewidth}{0.21\textheight}{c}}%
}%
\figureSeriesRow{%
\figureSeriesElement{Nec per sodales aliquyam.}{%
\resizebox{0.95\linewidth}{0.21\textheight}{d}}%
}%
}%
%
\begin{figure}%
\resizebox{0.31\columnwidth}{0.13\textheight}{e}%
\caption{Sodales condimentum nascetur tempus?}%
\end{figure}%
%
\begin{figure}%
\resizebox{0.31\columnwidth}{0.15\textheight}{f}%
\caption{Soluta autem nascetur litora.}%
\end{figure}%
%
\lipsum[14]%
%
\end{document}%

```

3 Implementation

The names of all macros for public use are prefixed with `figureSeries`. The names of all internal elements of the package are prefixed with `@figSer@`. This naming convention should prevent any name clashes with other packages.

3.1 Loading of Required Packages

Our `figureSeries` package requires three other packages:

1. The package `caption` [5] for the caption of the `figureSeries`.
2. The package `subcaption` [6] for sub-figure layout and captions.
3. The package `afterpage` [7] for creating the impression that our `figureSeries` can float.

3.1.1 Loading `caption` and `subcaption`

We rely on the packages `caption` and `subcaption` to render the `figureSeries`' and sub-figure's captions. However, Springer's `llncs.cls` [10] seems to be incompatible with the `subcaption` package [6]¹. Yet, we need that package for giving nice captions to the sub-figures. Therefore, we load the `caption` package [5] with option `compatibility=false` (which can solve this issue) if necessary. This is behavior governed by the Boolean flag `@figSer@captionCompatibilityFalse`, initialized to `false`.

```
1 \newif\if@figSer@captionCompatibilityFalse%
2 \@figSer@captionCompatibilityFalsefalse%
3 %
```

After the flag has been allocated and set to `false`, we check for Springer's `llncs.cls` (in a crude way). In case of `llncs.cls`, if we do not set the option `compatibility` of the `caption` package to `false`, we will get the error *! Package caption Error: The 'subcaption' package does not work correctly in compatibility mode*.

```
4 \ifx\spnewtheorem\undefined%
5 \else%
6 \@figSer@captionCompatibilityFalsetrue%
7 \fi%
8 %
```

Now we can load the `caption` [5] package with the right “compatibility” setting.

```
9 \if@figSer@captionCompatibilityFalse%
10 \RequirePackage[compatibility=false]{caption}%
11 \else%
```

¹According to <http://www.michaelshell.org/tex/ieeetran/>, IEEE's `IEEEtrans.cls` [11] may also be incompatible, but it seems to work here.

```

12 \RequirePackage{caption}%
13 \fi%

```

We now load the `subcaption` package [6] and set the caption style for sub-figures to arabic. The reason is that we may have many sub-figures, too many for indexes ranging only from *a*) to *z*). By using arabic numbers, we are on the safe side.

```

14 \RequirePackage{subcaption}%
15 \DeclareCaptionSubType*[arabic]{figure}%

```

3.1.2 Loading the afterpage Package

Floating objects cannot break across pages, so we cannot really make our figure series float. However, by using the `afterpage` package, we can make it look as if it was floating by rendering it on the next page.

```

16 \RequirePackage{afterpage}%

```

3.2 User Interface Macros

This section contains the macros which the package user can/should access, i.e., those macros which have been shortly discussed in Section 2.2.

`\figureSeriesElement` The macro `\figureSeriesElement{<caption>}{<contents>}` inserts an element of the figure series, i.e., one sub-figure. Its first argument is the caption of the element. This argument also may contain a `\label`. The second argument is the graphic to print. It could, e.g., be a call to `\includegraphic` from the `graphicx` package [8].

Spacing between sub-figures is handled dynamically via `\hfill`. We make sure to tell the `subcaption` that the sub-figures are sub-figures by setting `@captive` appropriately.

```

17 \long\gdef\figureSeriesElement#1#2{%
18 \strut\hfill\strut%
19 \edef@captive{figure}%
20 \subcaptionbox{#1}{#2}%
21 \strut\hfill\strut%
22 }%

```

`\figureSeriesRow` `\figureSeriesRow{<contents>}` inserts a new row of elements (sub-figures) into the figure series. Its single argument should thus contain a sequence of `\figureSeriesElements`. Since sub-figures are placed row-by-row, no sub-figure can span multiple rows.

We want to avoid having a figure series broken right below the caption, i.e., having a caption alone at the bottom of a page and the first row of sub-figures coming on the next page. Therefore, we temporarily store the overall caption of the figure series in `\@figSer@delayedCaption` and print it together with the first figure row.

```

23 \global\let\@figSer@delayedCaption\@empty%

```

If the overall caption of the figure series has not yet been printed, it would be stored in `\@figSer@delayedCaption`. Thus, if `\@figSer@delayedCaption` is not empty, this macro first prints the delayed caption and then the contents of the row together in a `\parbox` command which is wrapped into a `center` environment.

Again, the reason why we need to delay caption printing is that L^AT_EX's page breaking algorithm may separate the caption from the first figure row if the figure series starts close to the bottom of the page. Thus, we pack the delayed caption together with the first row of figures into a `\parbox` command. `\parboxes` are (hopefully) not affected by page breaking and (hopefully) always remain as solid objects. If the caption has already been printed, i.e., `\@figSer@delayedCaption` is empty, we do not need the awkward `\parbox`.

Plain `\parboxes`, however, seem to not work with the breaking of the `figureSeries` into multiple pieces to facilitate page breaking later on. Thus, we (need to?) wrap everything into a `center` environment.

```

24 \long\gdef\figureSeriesRow#1{%
25 \begin{center}%
26 \ifx\@figSer@delayedCaption\@empty%
27 \vspace\abovecaptionskip%
28 \strut#1\strut%
29 \else%
30 \parbox[b]{\textwidth}{%
31 \@figSer@delayedCaption%
32 \global\let\@figSer@delayedCaption\@empty%
33 \vspace\abovecaptionskip%
34 \strut#1\strut%
35 }%
36 \fi%
37 \end{center}}%
38 }%
```

The following two macros, `\figureSeriesHere` and `\figureSeriesFloat`, basically act as switches according to the different situations given in Section 3.3.3. Basically, different things need to be done for

1. the non-floating (“Here”) or floating (“Float”) case, as well as for
2. one-column or two-column documents.

We use the command `\@figSer@floatingBody` to temporarily store any floating figure series. If this command is `\@empty`, no floating figure series is pending.

```

39 \global\let\@figSer@floatingBody\@empty%
```

`\figureSeriesHere` The macro `\figureSeriesHere{<caption>}{<contents>}` tries to insert a non-floating figure series at the current position in the document. This means that it may begin wherever, well, it is defined, e.g., in the middle of the page.

The macro has two mandatory parameters, the caption and the contents. The caption will (different from usual `figures`) be put at the beginning of the figure

series. The reason is that a figure series may span over multiple pages and having the caption at the end may be awkward. The caption text may contain a `\label`.

The contents of a figure series should be a sequence of `\figureSeriesRow` calls.

Since figure series are page-wide elements, starting them in the middle of the page only works in one-column documents. In two-column documents, any figure series will behave as specified in macro `\figureSeriesFloat` below.

Furthermore, if a floating figure series is already pending for insertion, we will not print the current figure series here but attach it to the floating one. This will ensure that the order in which figure series' are printed is always the same as the order in which they are declared.

```

40 \long\def\figureSeriesHere#1#2{%
41 \ifx\@figSer@floatingBody\@empty%
42 \@figSer@detectColumns%
43 \if@figSer@isTwoColumns%
44 \@figSer@floatTwoCol{#1}{#2}%
45 \else%
46 \@figSer@print{#1}{#2}%
47 \fi%
48 \else%
49 \g@addto@macro{\@figSer@floatingBody}{\@figSer@print{#1}{#2}}%
50 \fi%
51 }%

```

`\figureSeriesFloat` The macro `\figureSeriesFloat{<caption>}{<contents>}` takes the same parameters as `\figureSeriesHere`, but has a float-like behavior. By using the `\afterpage` command from the `afterpage` package [7], we let it start at the following page. This is different from L^AT_EX' normal floating behavior, but as good as we can get with page-breaking objects, I think.

```

52 \long\gdef\figureSeriesFloat#1#2{%
53 \ifx\@figSer@floatingBody\@empty%
54 \@figSer@detectColumns%
55 \if@figSer@isTwoColumns%
56 \@figSer@floatTwoCol{#1}{#2}%
57 \else%
58 \@figSer@floatOneCol{#1}{#2}%
59 \fi%
60 \else%
61 \g@addto@macro{\@figSer@floatingBody}{\@figSer@print{#1}{#2}}%
62 \fi%
63 }%

```

3.3 Internal Utility Definitions and Macros

The way in which we can create figure series in two-column mode is different (and very significantly more complex and dodgy) from how it works in one-column mode. We thus need to detect in which mode we are. This is done in the following code. Since my knowledge of L^AT_EX and T_EX is very limited, this may be quite a

crude approach. Also, this code will probably not work if you load the `multicol` package. It will definitely not work with more than two columns.

3.3.1 Printing a Figure Series

Our figure series are basically nothing else than simple arrays of rows. Each `\figureSeriesRow` is wrapped into a `center` environment and the `\figureSeriesElements` are horizontally distributed inside using `\struts` and `\hfills`.

Here we define the macro for printing our figure series. The rest of the package will deal with the logic where and how to invoke it.

```
\@figSer@print The macro \@figSer@print{<caption>}{<contents>} does the work of printing the
figure series.
64 \long\def\@figSer@print#1#2{%
65 \def\@figSer@delayedCaption{%
66 \noindent\parbox{\textwidth}{%
67 \captionof{figure}{#1}%
68 \global\advance\c@figure by 0%
69 }%
70 \par}%
71 \begin{center}%
72 #2%
73 \end{center}%
74 \medskip%
75 }%
```

3.3.2 Two-Column Mode Detection

Anyway, I first define a new Boolean flag `\if@figSer@isTwoColumns`, which will denote whether or not we are in two-column mode. It is initialized to `false`.

```
76 \newif\if@figSer@isTwoColumns%
77 \@figSer@isTwoColumnsfalse%
```

```
\@figSer@detectColumns The internal macro \@figSer@detectColumns detects whether or not we are in
two-column mode. It will set \if@figSer@isTwoColumns to \@figSer@isTwoColumnstrue
if yes and to \@figSer@isTwoColumnsfalse otherwise.
78 \def\@figSer@detectColumns{%
79 \@figSer@isTwoColumnsfalse%
80 \ifx\multicols\@undefined%
81 \else%
82 \ifnum\col@number>\@ne%
83 \@figSer@isTwoColumnstrue%
84 \fi%
85 \fi%
86 \if@twocolumn%
87 \@figSer@isTwoColumnstrue%
88 \fi%
89 }%
```

3.3.3 The Four Cases

If no figure series is already pending for layout (we took care of this in the user interface macros), then there are four situations that need to be distinguished when building a `figureSeries` and for each of those, we define an appropriate internal macro:

1. Single-column mode and `\figureSeriesHere`. This is the easiest situation, we just apply `\@figSer@print{#1}{#2}` directly (in the user interface macros). The figure series will perfectly line into the existing text.
2. Single-column mode and `\figureSeriesFloat`. In this case, we just put our macro into an `\afterpage` call.
3. Two-column mode and `\figureSeriesFloat`: This is the tricky part. We need to leave two-column mode to one-column mode, insert our environment, and revert back to two-column mode. This causes all sorts of problems and oddities, as discussed later.
4. Two-column mode and `\figureSeriesHere`: Same as two-column mode and `\figureSeriesFloat`, since we need to switch to one-column mode for the series anyway.

Additionally, we need to deal with are documents that contain only a `figureSeries` but no other contents. In that case, no output would be produced as no page will ever end and, hence, the contents of `\afterpage` will never be invoked. We therefore add a `\AtEndDocument` hook to create the necessary page. We discuss this issue and present the corresponding code in Section 3.3.6.

`\@figSer@floatOneCol` The internal macro performing the work of `\figureSeriesHere` in one-column mode for floating figure series. This macro is called only if no other floating figure series is pending, i.e., `\@figSer@floatingBody` will be `\@empty` and no `\afterpage` call is pending (from the `figureSeries` package, that is).

```

90 \long\def\@figSer@floatOneCol#1#2{%
91 \@figSer@hook%
92 \gdef\@figSer@floatingBody{\@figSer@print{#1}{#2}}%
93 \afterpage{%
94 \@figSer@floatingBody%
95 \global\let\@figSer@floatingBody\@empty%
96 \@figSer@unhook%
97 }%
98 }%
```

3.3.4 Two-Column Mode

Documents with two columns create a big headache. After a long search, I think I am close to working solution. The problem is as follows:

A. Figure series make only sense as page-wide constructs. Figure series can include a sequence of many figures which may span over multiple pages. If we would restrict them to one-column elements, then they could start in the middle of left column and end in the middle of the right column. That would look very odd. Also, it is our goal to create something which looks more or less the same, regardless whether the document is typeset as single- or two-column.

B. This means we need to switch to one-column mode if we are in two-column mode. We can do that with the `\onecolumn` command, but this will always begin a new page. This could mean that we get a (true) page break after only half of the left column. This would look odd. But OK, we can define figure series in two-column mode as floating objects by default and use the same `\afterpage` facility that we used to make figure series in one-column mode float. We then just put the `\onecolumn` into the stuff we float away with `\afterpage` and avoid the ugly page break in the middle of the text.

C. In two-column mode, `\afterpage` seemingly behaves more like “`\aftercolumn`”. That is, if we are currently in the left column, its contents will not be rendered at the start of the next page, but at the start of the right column. Only if we are in the right column, it will put our stuff on the next page. Well, this is already the answer: If we are in the left column, we do something like `\afterpage{\afterpage{...}}`.

D. The biggest problem is that we need to switch back to two-column mode after our figure series. We can do that with the `\twocolumn` command, but this will entail a `\clearpage` afterwards. This means that even if our figure series only occupies the top 20% of a page, there will be a page break, leaving the rest of the page blank before continuing in two-column mode.

E. After searching for a long time for a solution, I found a lead in an answer to the StackOverflow question <http://tex.stackexchange.com/questions/74433> on a vaguely related issue. In his answer [14], Tomáš Hejda introduced a way to switch to the `\twocolumn` mode and prepending a long series of vertical boxes. I derive my solution from his code from the answer into this package but renamed all involved entities to fit to our internal naming scheme.

F. If more than one figure series is started on the same page, then we may get an empty page inbetween because of the `\clearpage` done by `\onecolumn`. This is one of the reason of the automatic coalescing done in the user interfaces: If another figure series is pending to be laid out (`\@figSer@floatingBody` is not `\@empty`), we attach the contents of our new figure series to the contents of the pending one via `\g@addto@macro`. After the figure series is actually laid out, the body macro is emptied.

G. Another problem we need to deal with is the fact that `afterpage` [7] has neither been designed for two-column documents nor for situations with other floats flying around. This may lead to ! `LaTeX Error: Float(s) lost.` errors.

Currently, this problem remains unsolved.

It should be noted that Carlisle [7] explicitly states in the release notes of the `afterpage` package: “*This implementation does not work in two column mode, and can get ‘confused’ by L^AT_EX’s floating environments.*” This now comes back here to haunt me.

`\@figSer@floatTwoCol` The internal macro performing the work of `\figureSeriesFloat` in two-column mode.

```

99 \long\def\@figSer@floatTwoCol#1#2{%
100 \@figSer@hook%
101 \gdef\@figSer@floatingBody{\@figSer@print{#1}{#2}}%
102 \global\let\@figSer@breakBodyRepeat\@figSer@breakBodyRepeat@Orig%
103 \if@firstcolumn%
104 \afterpage{\afterpage{%
105 \@figSer@breakBodyRepeat%
106 \@figSer@unhook%
107 }}%
108 \else%
109 \afterpage{%
110 \@figSer@breakBodyRepeat%
111 \@figSer@unhook%
112 }%
113 \fi%
114 }%
```

3.3.5 The Two-Column Layout Algorithm (based on [14])

I derive my solution based on the code provided by Tomáš Hejda in his answer to question <http://tex.stackexchange.com/questions/74433> [14]. Instead of copy-pasting it, I *a)* changed all naming to match this package’s convention and made one minor modification, and *b)* changed quite a bit of the code’s internal structure.

The name changing has two reasons: First, it ensures that all parts of this package follow the same naming convention and can easily be identified as parts of this package. Second, if Tomáš Hejda should turn his code into a package (as his answer [14] indicated), there will be no name clashes between his package and mine.

Some of the numbers, like for the page height fraction (0.84) or the `vspace` (15pt plus 15pt), seem to be fixed to somewhat arbitrary values, which may be better taken from the current configuration (like `\floatpagefraction` or `\medskip` something). Right now, I don’t have the nerve to check for this, but I may check this again in the future.

First, two `\newsaveboxes` are allocated: `\@figSer@wholeBodyBox` will contain the figure series body from which pieces are chopped off. `\@figSer@printBox` is

used to store the piece of the body which will be printed.

```
115 \newsavebox\@figSer@wholeBodyBox%
116 \newsavebox\@figSer@printBox%
```

\@figSer@breakBodyRepeat@Orig This macro is the inner layout loop: It repetitively slices chunks off \@figSer@wholeBodyBox as big as possible but small enough to fit on a page are cut off and printed. Different from [14], we implement this macro for re-use in the next figure series. I also removed some boxing whose purpose I do not understand.

```
117 \def\@figSer@breakBodyRepeat@Orig{%
118 \twocolumn[ {%
119 \ifx\@figSer@floatingBody\@empty%
120 \else%
121 \global\setbox\@figSer@wholeBodyBox\vbox{\@figSer@floatingBody}%
122 \global\let\@figSer@floatingBody\@empty%
123 \fi%
124 \ifdim\ht\@figSer@wholeBodyBox>\textheight%
125 \setbox\@figSer@printBox\vsplit\@figSer@wholeBodyBox to \textheight%
126 \global\setbox\@figSer@wholeBodyBox\vbox{\unvbox\@figSer@wholeBodyBox}%
127 \box\@figSer@printBox%
128 \else%
129 \ifdim\ht\@figSer@wholeBodyBox>0.84\textheight%
130 \global\let\@figSer@breakBodyRepeat\clearpage%
131 \else%
132 \global\let\@figSer@breakBodyRepeat\relax%
133 \fi%
134 \unvbox\@figSer@wholeBodyBox%
135 \vspace{15pt plus 15pt}%
136 \fi%
137 }]%
138 \@figSer@breakBodyRepeat%
139 }%
```

3.3.6 Empty Documents

Empty documents pose a threat. If no page is to end, the contents of \afterpage will not be invoked. Hence, we add a hook via \AtEndDocument. This hook (\@figSer@AtEndDocumentHook) will be \relax, unless a figure series is currently pending in an \afterpage. In this case, it is \clearpage. This should only happen if the document was empty. As soon as the figure series is laid out, the hook will be reset to \relax.

```
140 \def\@figSer@hook{\global\let\@figSer@AtEndDocumentHook\clearpage}%
141 \def\@figSer@unhook{\global\let\@figSer@AtEndDocumentHook\relax}%
142 \@figSer@unhook%
143 \AtEndDocument{\@figSer@AtEndDocumentHook}%

```

3.4 Errors, Tests, and Incompatibilities

Currently, the biggest issue with the package is that it causes `! LaTeX Error: Float(s) lost.` errors in two-column mode in conjunction with other floating objects hanging around, as mentioned in Section 3.3.4. This is a huge pain and currently remains unsolved.

`figureSeries` loads the packages `caption` [5], `subcaption` [6], and `afterpage` [7]. Therefore it inherits all incompatibilities of these packages. The `subcaption` package, for instance, is not compatible with the packages `subfigure` and `subfig`.

3.5 Related Work

The `longfigure` package [13] provides a similar functionality, i.e., a figure environment that can wrap over multiple pages. This environment can be made to float by using `\afterpage`, but does not work in two-column documents.

Tomáš Hejda's method [14], which is used by our package, may be extended to provide similar functionality, if it is published as package as indicated in [14].

3.6 License

The copyright (c) of this work is with Thomas Weise (<http://www.it-weise.de>) except for the code in Section 3.3.5. The code in Section 3.3.5 is a modified version from Tomáš Hejda's (tohecz@gmail.com) answer [14] to question <http://tex.stackexchange.com/questions/74433>. The copyright of this is thus not clear to me, so I won't claim it :-)

This document, the package, and its documentation are under the LaTeX Project Public License, version 1.3, which may be found online at <http://www.latex-project.org/lppl.txt>.

The distribution of this package may also contain the L^AT_EX document classes `llncs.cls` [10], `IEEEtran.cls` [11], and `sig-alternate.cls` [12]. The copyrights of these files are with their respective owners and these owners alone can determine the license terms for these files. The files are just included here to make the examples stand-alone. They may not be up-to-date, so please download their latest versions from their respective locations.

If you feel that this document, its presentation, or any parts I provide for download violate your (copy)right, please contact me (<http://www.it-weise.de>, tweise@ustc.edu.cn, tweise@gmx.de) and I will immediately resolve the issues.

References

- [1] Thomas Weise, Raymond Chiong, Ke Tang, Jörg Lässig, Shigeyoshi Tsutsui, Wenxiang Chen, Zbigniew Michalewicz, and Xin Yao. Benchmarking Optimization Algorithms: An Open Source Framework for the Traveling Salesman Problem. *IEEE Computational Intelligence Magazine (CIM)*, 9 (3), August 2014. doi: 10.1109/MCI.2014.2326101, <http://www.it-weise.de/documents/files/WCTLTCMY2014B0AAOSFFTTSP.pdf>

- [2] Hemant Ritturaj Kushwah. Subfigs of a Figure on Multiple Pages, July 3, 2009. URL <http://stackoverflow.com/questions/1078370>.
- [3] arjimage and Hendrik Vogt. Splitting subfigures across multiple pages – but programmatically!, February 14, 2011. URL <http://tex.stackexchange.com/questions/11059>.
- [4] Hanns-Martin Münch and Don Hosec. *The morefloats Package*, January 28, 2012, version v1.0f URL <http://www.ctan.org/pkg/morefloats>
- [5] Axel Sommerfeldt. *The caption Package – Customising captions in floating environments*, November 2, 2011. URL <http://www.ctan.org/pkg/caption>.
- [6] Axel Sommerfeldt. *The subcaption package – Support for sub-captions*, April 16, 2013. URL <http://www.ctan.org/pkg/subcaption>.
- [7] David P. Carlisle. *The afterpage Package – Execute command after the next page break*, October 27, 1995. URL <http://www.ctan.org/pkg/afterpage>.
- [8] David P. Carlisle and Sebastian Rahtz. *Packages in the ‘graphics’ bundle*, April 27, 2014. URL <http://www.ctan.org/pkg/graphicx>.
- [9] Patrick Happel. *lipsum – Access to 150 paragraphs of Lorem ipsum dummy text*, July 27, 2014. URL <http://www.ctan.org/pkg/lipsum>.
- [10] LLNCS Document Class – Springer Verlag LaTeX2e support for Lecture Notes in Computer Science, June 12, 2010. URL <ftp://ftp.springer.de/pub/tex/latex/llncls/latex2e/llncls2e.zip>.
- [11] Gerald Murray, Silvano Balemi, Jon Dixon, Peter Nüchter, Jürgen von Hagen, and Michael Shell. Official IEEE LaTeX Class for Authors of the Institute of Electrical and Electronics Engineers (IEEE) Transactions Journals and Conferences, May 3, 2007. URL <http://www.michaelshell.org/tex/ieeetran/>.
- [12] Gerald Murray and G.K.M. Tobin. SIG-ALTERNATE.CLS – Version 2.4 (Compatible with the ACM-PROC-ARTICLE-SP.CLS“ V3.2SP), April 22, 2009. URL <http://www.acm.org/sigs/publications/proceedings-templates>.
- [13] Tim Arnold. *The longfigure Package – Provides a figure-like environment that break over pages*, January 6, 2014. URL <http://www.ctan.org/pkg/longfigure>.
- [14] Tomáš Hejda (answer, email: tohecz@gmail.com), Augustin (question) *Multi-Page One-Column Abstract in a Two-Column Document*, September 27, 2012. URL <http://tex.stackexchange.com/questions/74433>.

Change History

v0.9.0	column starts are now aligned well. 1
General: Initial Draft Version 1	
v0.9.1	v0.9.3
General: Better examples showing the shortcomings of the package (in particular in two-column mode). 1	General: Failed attempt to fix “LaTeX Error: Float(s) lost.” errors. 1
v0.9.2	v0.9.4
General: Shortcomings in two-column mode fixed: vertical	General: Some minor fixes, but the problem that I thought I solved in release 0.9.3 remains. 1

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols 91, 100, 140	\captionof 67
\@capytype 19	\@figSer@isTwoColumnsfalse\clearpage ... 130, 140
\@empty 23, 26, 32, 39, 41, 53, 95, 119, 122	\col@number 82
\@figSer@isTwoColumnstrue	
\@figSer@AtEndDocumentHook 83, 87	D
.... 140, 141, 143	\DeclareCaptionSubType
\@figSer@breakBodyRepeat 49, 61, <u>64</u> , 92, 101 15
.... 102, 105,	E
110, 130, 132, 138	\edef 19
\@figSer@breakBodyRepeat@Orig 96,	\else .. 5, 11, 29, 45,
..... 102, <u>117</u>	48, 57, 60, 81,
\@figSer@captionCompatibilityFalse 115, 121,	108, 120, 128, 131
..... 2	\end 37, 73
\@figSer@captionCompatibilityFalsetrue 124–126, 129, 134	
..... 6	\@one 82
\@figSer@delayedCaption . 23, 26, 31, 32, 65	\@undefined 4, 80
\@figSer@detectColumns 42, 54, <u>78</u>	F
\@figSer@floatOneCol 58, <u>90</u>	\figureSeriesElement <u>17</u>
\@figSer@floatTwoCol 44, 56, <u>99</u>	\figureSeriesFloat . <u>52</u>
\@figSer@floatingBody . 39, 41, 49, 53,	\figureSeriesHere . <u>40</u>
61, 92, 94, 95,	\figureSeriesRow .. <u>23</u>
101, 119, 121, 122	
\@figSer@hook C	G
\@figure 68	\g@addto@macro .. 49, 61
	\gdef . 17, 24, 52, 92, 101
	\global 23, 32,
	39, 68, 95, 102,
	121, 122, 126,
	130, 132, 140, 141

H	\long 17, 24,	S
\hfill 18, 21	40, 52, 64, 90, 99	\setbox . . . 121, 125, 126
\ht 124, 129		\spnewtheorem 4
	M	\strut . . . 18, 21, 28, 34
I	\medskip 74	\subcaptionbox 20
\if@figSer@captionCompatibility 1, 9	\multicols 80	
\if@figSer@isTwoColumns 43, 55, 76	N	T
\if@firstcolumn . . . 103	\newif 1, 76	\textheight 124, 125, 129
\if@twocolumn 86	\newsavebox . . . 115, 116	\textwidth 30, 66
\ifdim 124, 129	\noindent 66	\twocolumn 118
\ifnum 82	P	U
\ifx 4, 26, 41, 53, 80, 119	\par 70	\unvbox 126, 134
	\parbox 30, 66	
L	R	V
\let 23, 32, 39, 95, 102, 122, 130, 132, 140, 141	\relax 132, 141	\vbox 121, 126
	\RequirePackage 10, 12, 14, 16	\vspace 27, 33, 135
		\vsplit 125