

The smartdiagram package*

Claudio Fiandrino

claudio.fiandrino@gmail.com

April 1, 2013

Abstract

This package provides a way to easily draw diagrams in documents and presentations from a list of items thanks to TikZ. The idea came out from [this question](#) on [TeX.StackExchange](#).

Contents

1	Introduction and requirements	2
2	Basic Usage	3
3	Additions	4
4	The options	7
4.1	Setting the options	7
4.2	Available options	7
4.2.1	General options	7
4.2.2	Specific options	8
4.2.3	Options of the additions library	11
5	Gallery of examples	12
6	Recommendations and known issues	20
6.1	Something about colors	20
6.2	Defining styles	21
6.3	Circular, bubble and constellation diagrams	21
6.4	Descriptive diagrams	22
6.5	Decorations	23
6.6	Priority descriptive diagrams	23

*This document corresponds to smartdiagram v0.3, dated 2013/04/01; it is released under and subject to the [L^AT_EX](#) Project Public License (LPPL).

7	Aknowledgements	24
8	Implementation	24
8.1	Initialization and Package Options	24
8.2	Keys and color declaration	25
8.3	Commands	34
8.4	Library Additions	44

1 Introduction and requirements

The aim of the package is to provide a way to draw diagrams starting from a list of items colored automatically. The diagrams created could be used in a simple document or in a presentation: in the latter case, while using the Beamer class, the user could decide if diagrams should be overlay-awared or not.

Automatically, the smartdiagram package loads:

- TikZ;
- etoolbox;
- xparse;
- xstring (from version 0.2).

Moreover, the package loads the following TikZ libraries:

- `backgrounds`;
- `calc`;
- `fadings`;
- `shadows`;
- `shapes.arrows`;
- `shapes.symbols` (from version 0.2).

and it sets a new layer called `smart diagram arrow back`. From version 0.3, by loading the library

```
\usesmartdiagramlibrary{additions}
```

forces the TikZ library `positioning` to be loaded as well.

The package could be loaded by means of `\usepackage{smartdiagram}`.

2 Basic Usage

`\smartdiagram` The basic command is `\smartdiagram[⟨type of diagram⟩]{⟨list of items⟩}`. The `{⟨list of items⟩}` should be comma-delimited: to insert, for example, a comma as part of the item label use `\smartdiagram[...]{elem1,{elem2,text},elem3}`.

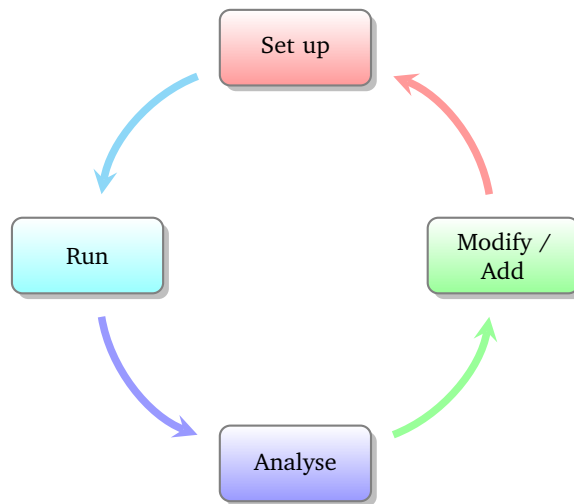
The possible diagrams that could be created are:

- **circular diagram**: the items in the list are displayed around a circle typically in counterclockwise order;
- **circular diagram:clockwise**: same as before, but now items are placed in clockwise order (no space between names and `:`);
- **flow diagram**: the items in the list are displayed as a flow chart;
- **flow diagram:horizontal**: the items in the list are displayed as an horizontal flow chart (no space between names and `:`);
- **descriptive diagram**: a diagram in which are displayed concepts and their description;
- **priority descriptive diagram**: a diagram in which the items are deployed based on their relevance;
- **bubble diagram**: each item is a bubble deployed around a bubble center, which is the first element in the list;
- **constellation diagram**: each item is a circle connected to the center, the first element in the list again;
- **connected constellation diagram**: each item is a circle and, a part from the first element in the list, the other ones are connected together;
- **sequence diagram**: the items in the list are displayed in sequence and each item points to the subsequent.

For example:

```
\begin{center}
\smartdiagram[circular diagram]{Set up,Run,Analyse,Modify~/ Add}
\end{center}
```

produces:



All diagrams could be customized in various ways, from the selection of the background colors to the font size, from the size of the items to the shape of the border. The keys necessary for this task will be analysed in detail in section 4.

For what concern the **bubble diagram**, the **constellation diagram** and the **connected constellation diagram**, they are a bit different from the rest of the diagrams, in the sense that the first item in the $\{\langle list\ of\ items \rangle\}$ is particular: called *bubble center* and *planet*, respectively, its aspect is different and it could be customized with dedicated keys.

From version 0.2 there is available also the **sequence diagram** which displays the $\{\langle list\ of\ items \rangle\}$ with a particular shape.

`\smartdiagramanimated`

Inside presentations, the user could select if the diagram should be displayed in a *persistent* way, or with an animation. The *persistent* way is achieved by exploiting, again, `\smartdiagram`, while to have the diagram *overlay-aware* the command is `\smartdiagramanimated[⟨type of diagram⟩]{⟨list of items⟩}`.

For diagrams like the **circular diagram** and the **flow diagram** the animation runs as follows: at the beginning the first item of the list is displayed, then each time appears an arrow connecting the previous element with the new one; at the end it is displayed the arrow connecting the last element with the first one. Considering instead the **descriptive diagram**, per couple description title-description, first it is shown the description title and subsequently the description. In the **priority descriptive diagram**, very simply, the list of items is deployed starting from the bottom, that is the less relevant item, to the top to progressively show much relevant items. For what concern the **bubble diagram**, the **constellation diagram** and the **connected constellation diagram**, at first it is always shown the *bubble center* and *planet*, respectively and later all the items subsequently. Finally, in the **sequence diagram**, the items are presented one at a time starting from the left one ending with the right one.

3 Additions

`\usesmartdiagramlibrary`

From version 0.3, it is possible to load a separate library called **additions** which allows to create annotations over a smart diagram. Load the library through:

`\usesmartdiagramlibrary{additions}`

`\smartdiagramadd` in the preamble. The basic command introduced by the library is `\smartdiagramadd[⟨type of diagram⟩]{⟨list of items⟩}{⟨list of additions⟩}`. The `{⟨list of additions⟩}` have a special syntax:

`⟨position of module/Annotation text⟩`

where:

- *position* is an anchor of TikZ (i.e above, below right and so on);
- *module* is the name of a module in the smart diagram;
- *position* and *module* should be separated by the string of: spaces before and after the string are *mandatory*.

Smartdiagram defines as names:

- for the diagrams `circular diagram` and `circular diagram:clockwise:`
`moduleprogressive-number` (no space or other symbols in between); example: `module1`;
- for the diagrams `flow diagram` and `flow diagram:horizontal:`
`moduleprogressive-number`; example: `module3`;
- for the diagram `descriptive diagram: module-title``progressive-number` for titles and `moduleprogressive-number` for descriptions; example: `module-title1` and `module1`;
- for the diagram `bubble diagram: center bubble` for the center module and `moduleprogressive-number` for the other modules; example: `center bubble` and `module2`;
- for the diagrams `constallation diagram` and `connected constellation diagram:`
`planet` for the center module and `satelliteprogressive-number` for the other modules; example: `planet` and `satellite3`;
- for the diagram `priority descriptive diagram: moduleprogressive-number`;
- for the diagram `sequence diagram: sequence-item``progressive-number`; example: `sequence-item1`.

The use of the library `additions` in a document requires two compilation runs at least because of the TikZ options `remember picture` and `overlay`.

An example:

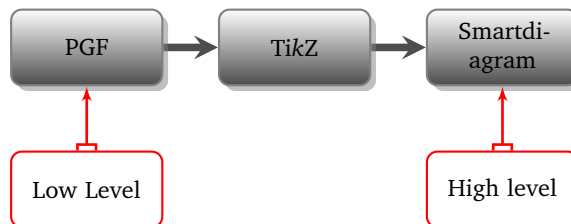
```
\begin{minipage}[t][3.5cm]{\textwidth}
\begin{center}
\smartdiagramset{
uniform color list=gray!60!black for 3 items,
back arrow disabled=true,
additions={
additional item offset=0.85cm,
additional item border color=red,
additional connections disabled=false,
additional arrow color=red,
additional arrow tip=stealth,
additional arrow line width=1pt,
additional arrow style=-latex',
}
```

```

}
\smartdiagramadd[flow diagram:horizontal]{%
PGF,Ti\textit{k}Z,Smartdiagram%
}{%
below of module1/Low Level, below of module3/High level%
}
\end{center}
\end{minipage}

```

The result:



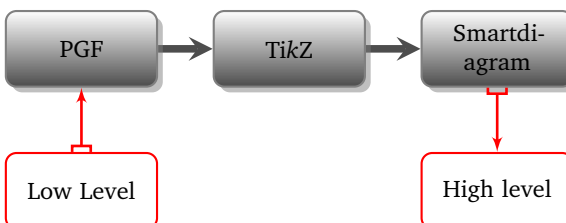
`\smartdiagramconnect` Notice that with the `\smartdiagramadd` facility it is not possible to fine customize the direction of the arrow tips. Since each additional module has as name `additional-moduleprogressive-number`, then by means of the specific command `\smartdiagramconnect{<arrow options>}{<start module/end module>}` one could do better. For example:

```

\begin{minipage}[t][3.5cm]{\textwidth}
\begin{center}
\smartdiagramset{
uniform color list=gray!60!black for 3 items,
back arrow disabled=true,
additions={
additional item offset=0.85cm,
additional item border color=red,
additional arrow color=red,
additional arrow tip=stealth,
additional arrow line width=1pt,
additional arrow style=-latex',
}
}
\smartdiagramadd[flow diagram:horizontal]{%
PGF,Ti\textit{k}Z,Smartdiagram%
}{%
below of module1/Low Level, below of module3/High level%
}
\smartdiagramconnect{{[-latex']}{additional-module1/module1}
\smartdiagramconnect{{[latex'-[]}{additional-module2/module3}
\end{center}
\end{minipage}

```

gives:



Because of the option `overlay`, it is better to protect within a `minipage` the diagram: in this way the missing bounding box update would not affect the subsequent/precedent text.

By default, the arrows are customized by the keys which start with `additional` (explained more in detail in the subsection 4.2.3), but it is possible to override locally this definition, for example with:

```
\smartdiagramconnect{{\latex'-[]},green}{additional-module2/module3}
```

For the moment, the library `additions` *does not work* in the animated mode.

4 The options

4.1 Setting the options

`\smartdiagramset` The options should be introduced similarly to what happens with `\tikzset` in TikZ: `\smartdiagramset{<list of options>}`. As well as in TikZ, it is possible to collect options within styles: see for more details subsection 6.2. Examples in which the options are used are shown in section 5.

4.2 Available options

Here follows the list of general available options: these options are related to generic aspects as color lists or arrows.

4.2.1 General options

- `set color list` (initial: none): this option allows the user to define the list of colors usable in the diagram;
- `uniform color list` (initial: none): this option allows the user to set one single color for the whole list of colors usable in the diagram;
- `use predefined color list` (initial: none): this option allows the user to reuse the pre-defined colors whenever in a previous diagram they have been changed;
- `insert decoration` (initial: none): this option allows to decorate the border of the module; the user should declare properly a decoration style and load manually the libraries required, e.g. `decorations.pathmorphing`; some more hints are given in section 6;

- `arrow line width` (initial: 0.1cm): this option sets the width of the connection arrows within two modules;
- `arrow tip` (initial: stealth): this option allows to select the single arrow tip; possible choices are described in the pgfmanual and for particular types it is recommended to load the library `arrows`;
- `arrow style` (initial: <-): this option allows the user to define a new style for the arrow; as well as the key `arrow tip`, see the pgfmanual to see a list of possible arrow styles;
- `uniform arrow color` (initial: false): this option, set to true, activates the possibility to use one single arrow color for all the connections;
- `arrow color` (initial: gray): this option, when the key `uniform arrow color` is set to true, allows to select the uniform arrow color.

4.2.2 Specific options

Here follows the list of specific available options per type of diagram.

For what concern the `circular diagram`, `circular diagram:clockwise`, the `flow diagram` and the `flow diagram:horizontal`:

- `module minimum width` (initial: 2cm): this option sets the minimum width of the module;
- `module minimum height` (initial: 1cm): this option sets the minimum height of the module;
- `module y sep` (initial: 1.65): this option sets a vertical distance factor among the modules in a `flow diagram`;
- `module x sep` (initial: 2.75): this option sets an horizontal distance factor among the modules in a `flow diagram:horizontal`;
- `module shape` (initial: rectangle,rounded corners): this option should be used to change the shape of the module, but the user should load manually the proper TikZ library; for example, to user an ellipse, load `shapes.geometric` in the preamble;
- `text width` (initial: 1.75cm): this option sets the text width inside the module;
- `font` (initial: \small): this option sets the module font;
- `border color` (initial: gray): this option sets the border color of the module;
- `text color` (initial: black): this option sets the text color of the module;
- `circular distance` (initial: 2.75cm): this option sets the radius of circle around which the modules in a `circular diagram`;

Considering just the `flow diagram` and the `flow diagram:horizontal`, there is a specific option to disable the back arrow going from the final module to the first one:

- `back arrow disabled` (initial: false): the option, set to true disables the back arrow.

Similarly, in the `circular diagram` and in the `circular diagram:clockwise`, there is a specific option to disable the back arrow going from the final module to the first one:

- `circular final arrow disabled` (initial: false): the option, set to true disables the final connection.

For what concern the **descriptive diagram** and the **priority descriptive diagram**:

- **descriptive items y sep** (initial: 1.75): this option sets a vertical distance factor among the descriptive items;
- **description title width** (initial: 1.5cm): this option sets the minimum width of the description-title;
- **description title text width** (initial: 1.25cm): this option sets the text width of the description-title;
- **description text width** (initial: 5cm): this option sets the text width of the description;
- **description title font** (initial: \small): this option sets the font size of the description-title;
- **description font** (initial: \small): this option sets the font size of the description;
- **description width** (initial: 5.5cm): this option sets the minimum width of the description;
- **priority arrow width** (initial: 1.5cm): this option sets the width of the vertical arrow;
- **priority arrow head extend** (initial: 0.15cm): this option sets the width of the arrow head extend;
- **priority tick size** (initial: 5pt): this option sets the size of the line denoting the relevance position of items in the vertical arrow;
- **priority arrow height advance** (initial: 2cm): this option specifies how much the vertical arrow is vertically extended above the most relevant item.

For what concern the **bubble diagram**, the **constellation diagram** and the **connected constellation diagram**:

- **bubble center node size** (initial: 4cm): the option specifies the minimum size of the bubble center node;
- **bubble center node font** (initial: \large): this option sets the font size of the bubble center node;
- **bubble center node color** (initial: lightgray!60): this option allows to customize the background color of the bubble center node;
- **distance center/other bubbles** (initial: 0.8cm): this options specifies which is the distance among the bubble center node and the other bubbles; keep this value under controll to avoid that bubbles do not overlap anymore the bubble center node;
- **distance text center bubble** (initial: 0.5cm): this option sets the distance from the text to the border of the bubble center node;
- **bubble fill opacity** (initial: 0.5): this option sets the opacity at which the bubbles are shown;
- **bubble node size** (initial: 2.5cm): the option specifies the minimum size of the bubbles;
- **bubble text opacity** (initial: 0.8): this option sets the opacity at which the bubble text is shown;

- `bubble node font` (initial: `\normalfont`): this option sets the font size of the bubbles;
- `planet size` (initial: `2.5cm`): the option specifies the minimum size of the planet;
- `planet color` (initial: `lightgray!60`): this option allows to customize the background color of the planet;
- `planet font` (initial: `\large`): this option sets the font size of the planet;
- `distance planet-connection` (initial: `0.1cm`): this option sets the distance from the planet to the arrow directed to the satellites;
- `distance planet-text` (initial: `0.5cm`): this option sets the distance from the text to the border of the planet;
- `planet text width` (initial: `1.75cm`): this option sets the planet text width;
- `satellite size` (initial: `1.75cm`): the option specifies the minimum size of the satellites;
- `satellite font` (initial: `\normalfont`): this option sets the font size of the satellites;
- `satellite fill opacity` (initial: `0.5`): this option sets the opacity at which the satellites are shown;
- `satellite text opacity` (initial: `0.8`): this option sets the opacity at which the satellite text is shown;
- `satellite text width` (initial: `1.5cm`): this option sets the satellite text width;
- `distance satellite-connection` (initial: `0.075cm`): this option sets the distance from the satellites to the arrows directed to the planet;
- `connection line width` (initial: `0.1cm`): this option allows to customize the width of the connections from the planet to the satellites;
- `distance planet-satellite` (initial: `3.5cm`): this option determine the distance among any pair of planet-satellite.

Considering just the the `constellation diagram` and the `connected constellation diagram`:

- `uniform connection color` (initial: `false`): the option, set to `true` overrides the color list definition;
- `connection color` (initial: `gray`): this option allows to specify the color valid for all the connections.

For what concern the `sequence diagram`:

- `sequence item height` (initial: `1cm`): the option specifies the minimum height of the items;
- `sequence item width` (initial: `2cm`): the option specifies the minimum width of the items;
- `sequence item border color` (initial: `gray`): sets the border line color;
- `sequence item border size` (initial: `1.65\pgflinewidth`): sets the border line width;

- `sequence item font size` (initial: `\normalfont`): this option sets the font size of the items;
- `sequence item fill opacity` (initial: 1): sets the opacity at which the item is shown;
- `sequence item text opacity` (initial: 1): sets the opacity at which the item text is shown;
- `sequence item text width` (initial: 1.9cm): the option allows to select the item text width;
- `sequence item text color` (initial: black): the option allows to select the item text color;
- `uniform sequence color` (initial: false): the option, set to true overrides the color list definition;
- `sequence item uniform color` (initial: gray!60!black): this option allows to specify the color valid for all the items in the sequence.

4.2.3 Options of the additions library

The options of the library necessitate to be set within a specific key `additions`; this key is defined as

```
\pgfkeys{/smart diagram/.cd,
additions/.style={/smart diagram/additions/.cd,#1}%
}
```

and it basically sets the correct path; indeed all of these keys are defined in a subtree of the main path: `/smart diagram/additions`. For example:

```
\smartdiagramset{
additions={
additional item offset=0.85cm,
additional item border color=red,
additional arrow color=red,
additional arrow tip=stealth,
additional arrow line width=1pt,
additional arrow style=-latex',
}
}
```

Notice that each key starts with `additional`: in my opinion, although it may seem heavy type this every time, it could avoid some confusion with other keys.

- `additional item shape` (initial: rectangle,rounded corners): this option should be used to change the shape of the additional module; similarly to the key `module shape`, for some shapes the user should load manually the proper TikZ library;
- `additional item border color` (initial: none): sets the border line color;
- `additional item bottom color` (initial: white): this option sets the bottom color of the module; use it if you want to keep the same aspect of the diagrams: `circular diagram`, `flow diagram`, `circular diagram:clockwise` and `flow diagram:horizontal`;

- `additional item fill color` (initial: none): this option sets the fill color of the module; use it if you want to keep the same aspect of the other types of diagrams;
- `additional item text width` (initial: 1.75cm): it defines the text width of the module;
- `additional item width` (initial: 2cm): it defines the minimum width of the module;
- `additional item height` (initial: 1cm): it defines the minimum height of the module;
- `additional item font` (initial: \normalfont): this option allows to customize the font of the module;
- `additional item border decoration` (initial: none): this option allows to customize the module with a decoration; some more hints are given in section 6;
- `additional item offset` (initial: 0.25cm): this option defines the distance between the original module and the additional one;
- `additional item fill opacity` (initial: 1): it sets the fill opacity of the module;
- `additional item text opacity` (initial: 1): it sets the text opacity of the module;
- `additional arrow tip` (initial: stealth): this option defines the single arrow tip of the connection;
- `additional arrow line width` (initial: 0.1cm): this option defines the line width of the connection;
- `additional arrow color` (initial: gray): this option defines the connection color;
- `additional arrow style` (initial: ->): this option allows to customize the connection aspect, that is both arrow tips simultaneously;
- `additional item shadow` (initial: none): it allows to define a shadow for the module; use the usual TikZ shadow options;
- `additional connections disabled` (initial: true): this option when set to false makes all the connections visible.

5 Gallery of examples

Horizontal flow chart:
custom color list - no
back arrow

Example of an horizontal flow chart with custom color list and back arrow disabled:

```
\begin{center}
\smartdiagramset{border color=none,
  set color list={blue!50!cyan,green!60!lime,orange!50!red,red!80!black},
  back arrow disabled=true}
\smartdiagram[flow diagram:horizontal]{Set up,Run,Analyse,Modify~/ Add}
\end{center}
```

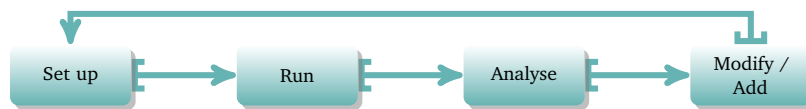


Horizontal flow chart: A similar example with an uniform color list and custom arrow style definition:

```

uniform color list -
custom arrow style
\begin{center}
\smartdiagramset{border color=none,
  uniform color list=teal!60 for 4 items,
  arrow style=[-stealth',
  module x sep=3.75,
}
\smartdiagram[flow diagram:horizontal]{Set up,Run,Analyse,Modify~/ Add}
\end{center}

```



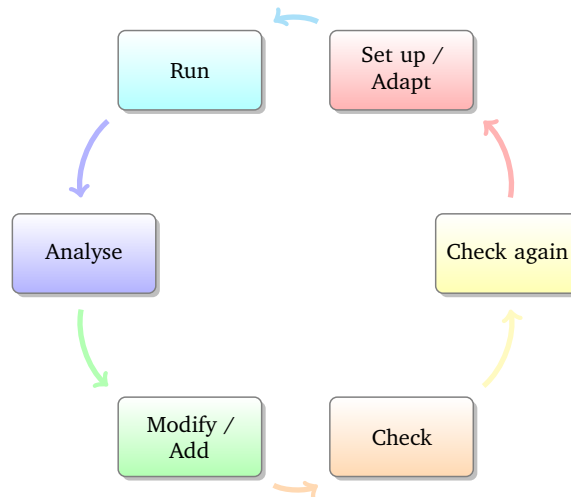
Circular diagram with custom options

Another example:

```

\begin{center}
\smartdiagramset{circular distance=4cm,
  font=\large,
  text width=2.5cm,
  module minimum width=2.5cm,
  module minimum height=1.5cm,
  arrow tip=to}
\smartdiagram[circular diagram]{Set up~/ Adapt,Run,Analyse,Modify~/ Add,
  Check,Check again}
\end{center}

```



Circular diagram with custom shape and final arrow disabled

An example with a diamond shape and the final arrow disabled:

```

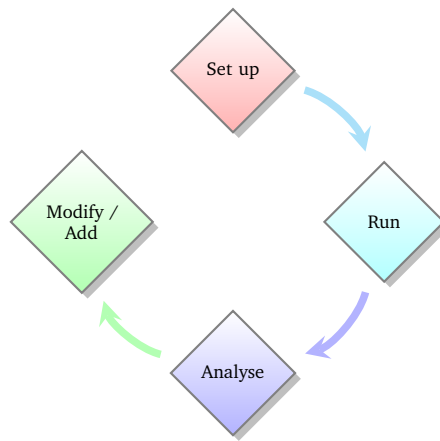
\begin{center}

```

```

\usetikzlibrary{shapes.geometric} % required in the preamble
\smartdiagramset{module shape=diamond,
  font=\scriptsize,
  module minimum width=1cm,
  module minimum height=1cm,
  text width=1cm,
  circular distance=2cm,
  circular final arrow disabled=true,
}
\smartdiagram[circular diagram:clockwise]{Set up,Run,Analyse,Modify~/ Add}
\end{center}

```



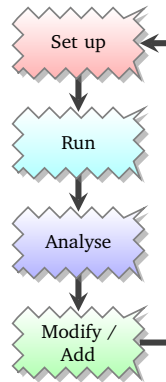
Flow diagram with
decorated border and
uniform arrow color

An example with a decorated shape and uniform arrow color:

```

\usetikzlibrary{decorations.pathmorphing} % required in the preamble
\begin{center}
\tikzset{my decoration/.style={decorate,decoration=zigzag}}
\smartdiagramset{module shape=rectangle,
  insert decoration={my decoration},
  uniform arrow color=true,
  arrow color=gray!50!black,
}
\smartdiagram[flow diagram]{Set up,Run,Analyse,Modify~/ Add}
\end{center}

```

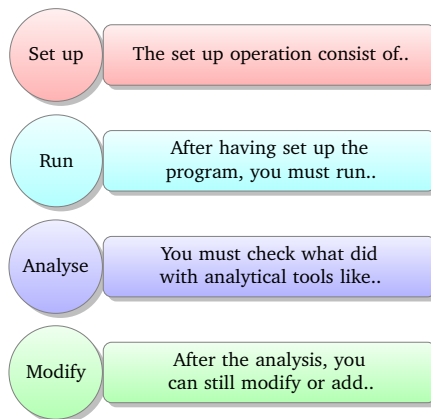


Descriptive diagram An example of descriptive diagram:

```

\begin{center}
\smartdiagram[descriptive diagram]{
{Set up,The set up operation consist of..},
{Run, {After having set up the program, you must run..}},
{Analyse, You must check what did with analytical tools like..},
{Modify, {After the analysis, you can still modify or add..}},
}
\end{center}

```



Bubble diagram An example of bubble diagram:

```

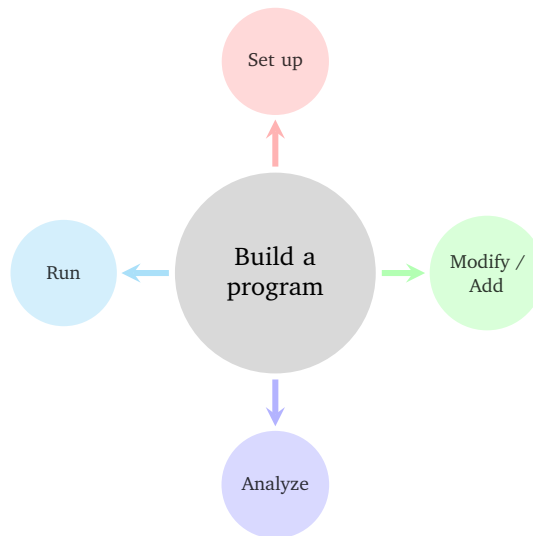
\begin{center}
\smartdiagram[bubble diagram]{
Build a program,Set up,Run,Analyze,Modify~/\ Add,Check
}
\end{center}

```



Constellation diagram An example of constellation diagram:

```
\begin{center}
\smartdiagram[constellation diagram]{
  Build a program,Set up,Run,Analyze,Modify~\\ Add
}
\end{center}
```



Constellation diagram with custom colors An example of constellation diagram with custom colors:

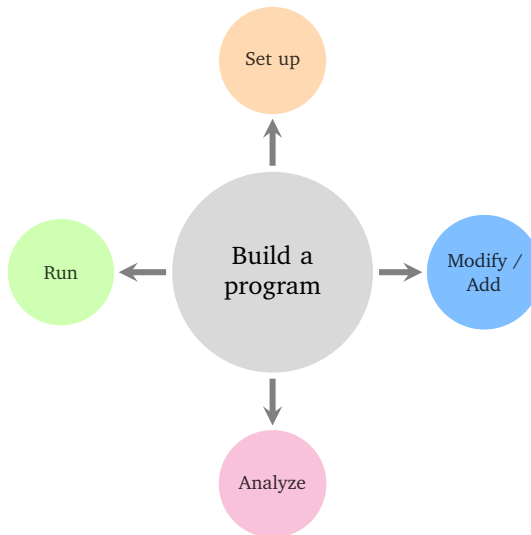
```
\begin{center}
\smartdiagramset{set color list={orange!60, green!50!lime!60,magenta!60,
  blue!50!cyan},
```



```

    uniform connection color=true
}
\smartdiagram[constellation diagram]{
  Build a program,Set up,Run,Analyze,Modify~/\ Add
}
\end{center}

```

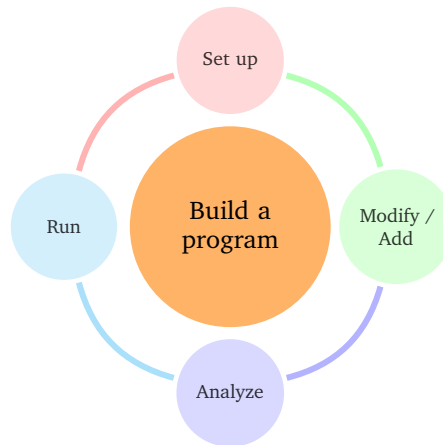


Connected constellation diagram An example of connected constellation diagram:

```

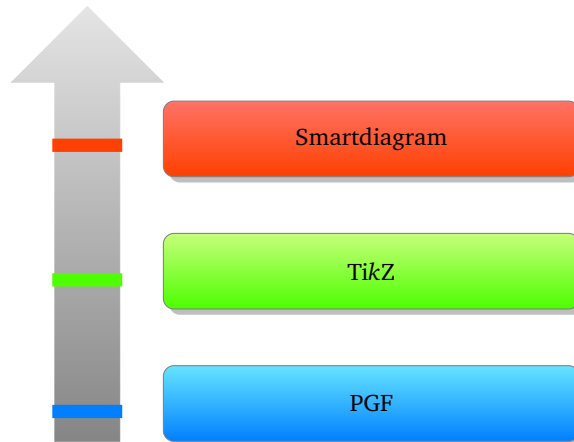
\begin{center}
\smartdiagramset{planet color=orange!60,
  distance planet-satellite=1cm
}
\smartdiagram[connected constellation diagram]
{Build a program,Set up,Run,Analyze,Modify~/\ Add,Check}
\end{center}

```



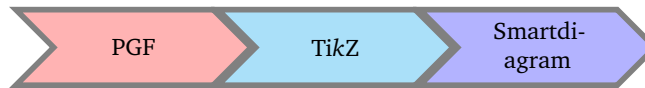
Priority descriptive diagram An example of priority descriptive diagram describing that TikZ is built on top of PGF and Smartdiagram on top of TikZ:

```
\begin{center}
\smartdiagramset{
  set color list={blue!50!cyan,green!60!lime,orange!50!red},
  priority arrow width=2cm,
  priority arrow height advance=2.25cm
}
\smartdiagram[priority descriptive diagram]{PGF,Ti\textit{k}Z,Smartdiagram}
\end{center}
```



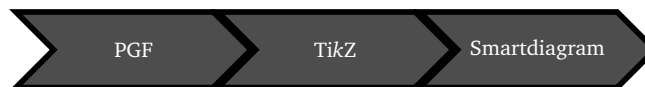
Sequence diagram The same previous example with a sequence diagram:

```
\begin{center}
\smartdiagram[sequence diagram]{PGF,Ti\textit{k}Z,Smartdiagram}
\end{center}
```



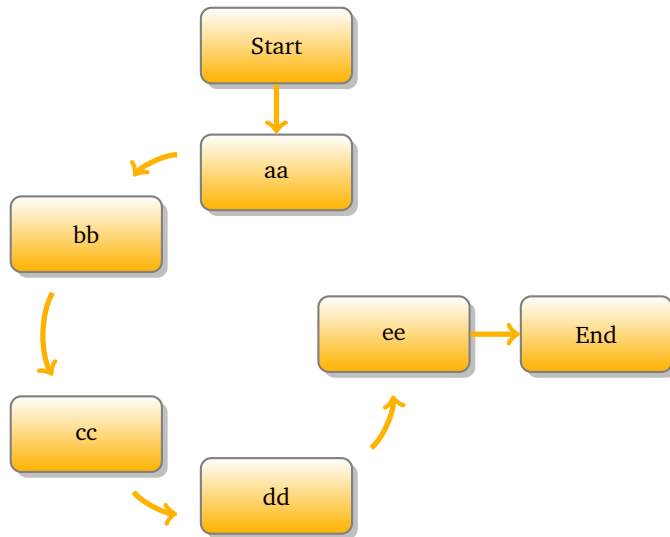
and with `uniform sequence color` set to true:

```
\begin{center}
\smartdiagramset{uniform sequence color=true,
sequence item border color=black,sequence item font size=\footnotesize,
sequence item text color=white
}
\smartdiagram[sequence diagram]{PGF,Ti\textit{k}Z,Smartdiagram}
\end{center}
```













Put additions to a diagram Here is an example of a circular diagram with some additions:

```
\usesmartdiagramlibrary{additions} % required in the preamble
\usetikzlibrary{arrows} % required in the preamble
\bigskip
\begin{minipage}[c][8cm]{\textwidth}
\centering
\smartdiagramset{
uniform color list=orange!60!yellow for 5 items,
circular final arrow disabled=true,
circular distance=2.25cm,
arrow tip=to,
arrow line width=2pt,
additions={
additional item bottom color=orange!60!yellow,
additional item border color=gray,
additional item shadow=drop shadow,
additional item offset=0.65cm,
additional arrow line width=2pt,
additional arrow tip=to,
additional arrow color=orange!60!yellow,
}
}
\smartdiagramadd[circular diagram]{
aa,bb,cc,dd,ee
}{
above of module1/Start,right of module5/End
}
\smartdiagramconnect{to-}{module1/additional-module1}
\smartdiagramconnect{-to}{module5/additional-module2}
\end{minipage}
```



6 Recommendations and known issues

6.1 Something about colors

As seen in the section 4, the colors could be customized by means of the key `set color list`. By default there are 10 predefined colors; in order: red!40 , cyan!40 , blue!40 , green!40 , orange!40 , yellow!40 , magenta!40 , brown!40 , violet!40 , and teal!40 . This implies that, by default, it is not possible to have lists longer than 10 items. This, of course, could be avoided by declaring proper lists with, say, 20 colors and therefore develop diagrams with more than 10 items. In any case, it is always possible to reset custom color lists by means of:

```
\smartdiagramset{use predefined color list}
```

In order to use one color for all the items it is possible to exploit the `uniform color list`; it has a particular syntax:

```
\smartdiagramset{uniform color list=<some color> for <n> items}
```

The `<some color>` is set for a list of `<n>` items and nothing more, so in order to avoid problems make sure you dimension `<n>` correctly. Indeed, in case `<n>` is lower than the number of items inside the diagram the following happens:

```
\begin{center}
\smartdiagramset{
  uniform color list=gray!60!black for 2 items,
  back arrow disabled=true,
}
```

```

}
\smartdiagram[flow diagram:horizontal]{PGF,Ti\textit{k}Z,Smartdiagram}
\end{center}

```



In conclusion, the uniform setting is extend only for $\langle n \rangle$ items, for the remaining ones the predefined or a custom color list is used.

Notice also that the key `uniform color list` makes the arrow color be uniform for $\langle n \rangle$ items, but it has no relation with the `uniform arrow color` which automatically makes *all* items with an uniform color. Indeed, the color taken by the arrows with:

```
\smartdiagramset{uniform arrow color=true}
```

could be customized through the key `arrow color`, while `uniform color list` make modules and arrows be rendered with the same color.

6.2 Defining styles

The `smartdiagram` package admits the definition of styles to collect key-definitions; for example:

```

\smartdiagramset{my diagram style/.style={
    module shape=diamond,
    font=\scriptsize,
    module minimum width=1cm,
    module minimum height=1cm,
    text width=1cm
}
}

```

can be subsequently used in:

```

\begin{center}
\smartdiagramset{my diagram style, arrow tip=to}
\smartdiagram[circular diagram]{Do, This, Only, For, Me}
\end{center}
\begin{center}
\smartdiagramset{my diagram style, module y sep=2.5}
\smartdiagram[flow diagram]{Do, This, For, Me}
\end{center}

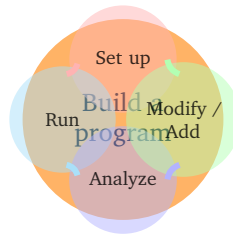
```

6.3 Circular, bubble and constellation diagrams

For these type of diagrams, the number of items is relevant: too many items lead to overlapping satellites and bubbles besides any attempt to resize things by means of keys that reduce the radius.

Notice also that imposing a too short distance from the planet to satellites is bad and leads to something like:

```
\begin{center}
\smartdiagramset{planet color=orange!60, distance planet-satellite=1cm}
\smartdiagram[connected constellation diagram]
{Build a program,Set up,Run,Analyze,Modify~/\ Add,Check}
\end{center}
```



6.4 Descriptive diagrams

When the user has to build a **descriptive diagram**, simple or animated, the following rules have to be respected:

- each description title and description should be separated by a comma;
- to use a comma inside a description, enclose by { } the description;
- use a comma after the last couple description title-description.

A working example:

```
\smartdiagram[descriptive diagram]{
{Set up, The set up operation consist of..},
{Run, {After having set up the program, you must run..}},
{Analyze, You must check what did with analytical tools like..},
}
```

A non-working example:

```
\smartdiagram[descriptive diagram]{
{Set up, The set up operation consist of..},
{Run, {After having set up the program, you must run..}},
{Analyze, You must check what did with analytical tools like..}
}
```

The *mandatory* final comma is missing thus, as result, the last couple description title-description will be entirely treated as a description title.

Another suggestion regarding descriptive diagrams is about the description title: it should be short in order to avoid the size of the circle explode. To keep it under controll, the keys `description title width`, `description title text width` and `description title font` are of help.

6.5 Decorations

To decorate the border of modules, it is kindly recommended to first declare the decoration chosen by means of an apposite style and then apply the style. The procedure, in code, should be as follows:

```
\tikzset{my wonderful decoration/.style={decorate,decoration=bent}}
\smartdiagramset{insert decoration=my wonderful decoration,...}
```

where the dots represent the other options.

The insertion of decorations inside a `descriptive diagram` are problematic: only random steps, bent and coil do not raise errors. Other decorations, like snake, raise as error:

```
! Dimension too large.
```

but, after all, the decoration is deployed anyway.

In case the user wants to decorate a border with a decoration that involves random numbers, such as random steps or other custom-built decorations, it is preferable to set a seed for the animated diagram, to avoid that at each step the border of the same module changes. An example:

```
\begin{frame}
\begin{center}
\pgfmathsetseed{12354}
\tikzset{my decoration/.style={decorate,decoration=random steps}}
\smartdiagramset{insert decoration=my decoration}
\smartdiagramanimated[descriptive diagram]{
{Set up, The set up operation consist of..},
{Run, {After having set up the program, you must run..}},
{Analyze, You must check what did with analytical tools like..},
{Modify, {After the analysis, you can still modify or add..}},
}
\end{center}
\end{frame}
```

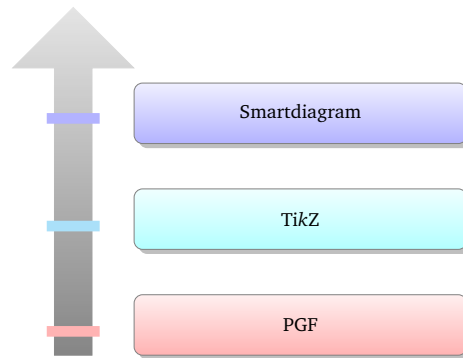
6.6 Priority descriptive diagrams

The vertical arrow is drawn in background with a fading effect: it may occur that under some pdf editor this effect is not shown (this happened to me with evince, but it perfectly worked with okular).

For what concern these diagrams there is a bug in displaying the tick line when the `priority arrow head extend` is set with a bigger size than the default; a minimal-non-working example:

```
\begin{center}
\smartdiagramset{priority arrow width=2cm,
priority arrow height advance=2.25cm,
priority arrow head extend=0.3cm}
```

```
\smartdiagram[priority descriptive diagram]{PGF,Ti\textit{k}Z,Smartdiagram}
\end{center}
```



7 Acknowledgements

I would like to acknowledge first of all [Alain Matthes](#) and [Mohsen](#) because the [bubble diagram](#) and the [constellation diagram](#) are based on [Alain's answer](#) while the [circular diagram](#) is based on [Mohsen's answer](#).

I would also like to thank [Enrico Gregorio](#) and [Ahmed Musa](#) for the courtesy of explaining me why my poor attempt in creating the `set color list` failed and for providing me valid solutions.

Last, but not least, I would like to thank prof. Ludger Humbert for suggesting and providing the code for the `circular diagram:clockwise` and André Hilbig for suggesting the key `back arrow disabled`.

8 Implementation

8.1 Initialization and Package Options

This subsection highlights which are the package loaded and the `tikzlibraries` needed.

```
1 \RequirePackage{tikz}
2
3 \RequirePackage{etoolbox}
4 \RequirePackage{xparse}
5 \RequirePackage{xstring}
6
7 \usetikzlibrary{backgrounds,
8   calc,
9   fadings,
10  shadows,
11  shapes.arrows,
12  shapes.symbols
13 }
14 \pgfdeclarelayer{smart diagram arrow back}
```



```

15 \pgfsetlayers{background,smart diagram arrow back,main}
16
17
18 \def\usesmartdiagramlibrary{\pgfutil@ifnextchar[{\use@smartdiagramlibrary}{\use@@smartdiagramlibrary}}
19 \def\use@smartdiagramlibrary[#1]{\use@@smartdiagramlibrary{#1}}
20 \def\use@@smartdiagramlibrary#1{%
21   \edef\pgf@list{#1}%
22   \pgfutil@for\pgf@temp:=\pgf@list\do{%
23     \expandafter\pgfkeys@spdef\expandafter\pgf@temp\expandafter{\pgf@temp}%
24     \ifx\pgf@temp\pgfutil@empty
25       \else
26         \expandafter\ifx\csname smartdiagram@library@\pgf@temp @loaded\endcsname\relax%
27           \expandafter\global\expandafter\let\csname smartdiagram@library@\pgf@temp @loaded\endcsname\relax%
28           \expandafter\edef\csname smartdiagram@library@#1@atcode\endcsname{\the\catcode'\@}
29           \expandafter\edef\csname smartdiagram@library@#1@barcode\endcsname{\the\catcode'\|}
30           \catcode'\@=11
31           \catcode'\|=12
32           \pgfutil@InputIfFileExists{smartdiagramlibrary\pgf@temp.code.tex}{%}{%
33             \PackageError{smartdiagram}{I did not find the smartdiagram library '\pgf@temp'.}{%
34             }%
35             \catcode'\@=\csname smartdiagram@library@#1@atcode\endcsname
36             \catcode'\|=\csname smartdiagram@library@#1@barcode\endcsname
37           \fi%
38         \fi
39       }%
40     }
41
42 \usesmartdiagramlibrary{core.definitions}
43 \usesmartdiagramlibrary{core.styles}
44 \usesmartdiagramlibrary{core.commands}

```

8.2 Keys and color declaration

The predefined colors:

```

45 \@namedef{color@1}{red!40}
46 \@namedef{color@2}{cyan!40}
47 \@namedef{color@3}{blue!40}
48 \@namedef{color@4}{green!40}
49 \@namedef{color@5}{orange!40}
50 \@namedef{color@6}{yellow!40}
51 \@namedef{color@7}{magenta!40}
52 \@namedef{color@8}{brown!40}
53 \@namedef{color@9}{violet!40}
54 \@namedef{color@10}{teal!40}

```

Basic shape definition and function to compute the height of the **priority descriptive diagram**:

```

55 \tikzset{rnd rectangle/.style={rectangle,rounded corners}
56 }
57

```

```

58 \def\CalcHeight(#1,#2)#3{%
59 \pgfpointdiff{\pgfpointanchor{#1}{south west}}{\pgfpointanchor{#2}{north west}}
60 \pgfmathsetmacro{\myheight}{veclen(\pgf@x,\pgf@y)}
61 \global\expandafter\edef\csname #3\endcsname{\myheight}
62 }

```

The key definition and the functions to set them:

```

63 \pgfkeys{/smart diagram/.cd, module minimum width/.initial=2cm,
64   module minimum height/.initial={1cm},
65   module y sep/.initial={1.65},
66   module x sep/.initial={2.75},
67   descriptive items y sep/.initial={1.75},
68   text width/.initial={1.5cm},
69   description title width/.initial={1.5cm},
70   description text width/.initial={5cm},
71   description title text width/.initial={1.25cm},
72   description title font/.initial={\small},
73   description font/.initial={\small},
74   description width/.initial={5.5cm},
75   font/.initial={\small},
76   border color/.initial={gray},
77   circular distance/.initial={2.75cm},
78   arrow line width/.initial={0.1cm},
79   module shape/.initial={rnd rectangle},
80   insert decoration/.initial={},
81   arrow tip/.initial={stealth},
82   arrow color/.initial={gray},
83   bubble center node size/.initial={4cm},
84   bubble center node font/.initial={\large},
85   bubble center node color/.initial={lightgray!60},
86   distance center/other bubbles/.initial=0.8cm,
87   distance text center bubble/.initial={0.5cm},
88   bubble fill opacity/.initial={0.5},
89   bubble node size/.initial={2.5cm},
90   bubble text opacity/.initial={0.8},
91   bubble node font/.initial={\normalfont},
92   planet size/.initial={2.5cm},
93   planet color/.initial={lightgray!60},
94   planet font/.initial={\large},
95   distance planet-connection/.initial={0.1cm},
96   distance planet-text/.initial={0.5cm},
97   planet text width/.initial={1.75cm},
98   satellite size/.initial={1.75cm},
99   satellite font/.initial={\normalfont},
100  satellite fill opacity/.initial={0.5},
101  satellite text opacity/.initial={0.8},
102  satellite text width/.initial={1.5cm},
103  distance satellite-connection/.initial={0.075cm},
104  connection line width/.initial={0.1cm},
105  connection color/.initial={gray},

```

```

106 distance planet-satellite/.initial={3.5cm},
107 priority arrow width/.initial={1.5cm},
108 priority arrow head extend/.initial={0.15cm},
109 priority tick size/.initial={5pt},
110 priority arrow height advance/.initial={2cm},
111 sequence item height/.initial={1cm},
112 sequence item width/.initial={2cm},
113 sequence item border color/.initial={gray},
114 sequence item border size/.initial={1.75\pgflinewidth},
115 sequence item font size/.initial={\normalfont},
116 sequence item fill opacity/.initial={1},
117 sequence item text opacity/.initial={1},
118 sequence item text width/.initial={1.9cm},
119 sequence item text color/.initial={black},
120 sequence item uniform color/.initial={gray!60!black},
121 arrow style/.initial={<-},
122 text color/.initial={black},
123 }%
124
125 \pgfkeys{/smart diagram/.cd, module minimum width/.get=\sm@core@modulewidth,
126 module minimum height/.get=\sm@core@moduleheight,
127 module y sep/.get=\sm@core@moduleysep,
128 module x sep/.get=\sm@core@modulexsep,
129 descriptive items y sep/.get=\sm@core@descriptiveitemsysep,
130 text width/.get=\sm@core@moduletextwidth,
131 description title width/.get=\sm@core@descriptiontitlewidth,
132 description text width/.get=\sm@core@descriptiontextwidth,
133 description title text width/.get=\sm@core@descriptiontitletextwidth,
134 description title font/.get=\sm@core@descriptiontitlefontsize,
135 description font/.get=\sm@core@descriptionfontsize,
136 description width/.get=\sm@core@descriptionwidth,
137 font/.get=\sm@core@modulefontsize,
138 border color/.get=\sm@core@bordercolor,
139 circular distance/.get=\sm@core@circulardistance,
140 arrow line width/.get=\sm@core@arrowlinewidth,
141 module shape/.get=\sm@core@moduleshape,
142 insert decoration/.get=\sm@core@borderdecoration,
143 arrow tip/.get=\sm@core@arrowtip,
144 arrow color/.get=\sm@core@arrowcolor,
145 bubble center node size/.get=\sm@core@bubblecenternodesize,
146 bubble center node font/.get=\sm@core@bubblecenternodefont,
147 bubble center node color/.get=\sm@core@bubblecenternodecolor,
148 distance center/other bubbles/.get=\sm@core@distancecenterotherbubbles,
149 distance text center bubble/.get=\sm@core@distancetextcenterbubble,
150 bubble fill opacity/.get=\sm@core@bubblefillopacity,
151 bubble node size/.get=\sm@core@bubblenodesize,
152 bubble text opacity/.get=\sm@core@bubbletextopacity,
153 bubble node font/.get=\sm@core@bubblenodefont,
154 planet size/.get=\sm@core@planetminimumsize,
155 planet color/.get=\sm@core@planetcolor,

```

```

156 planet font/.get=\sm@core@planetfont,
157 distance planet-connection/.get=\sm@core@planetoutersep,
158 distance planet-text/.get=\sm@core@planetinnersep,
159 planet text width/.get=\sm@core@planettextwidth,
160 satellite size/.get=\sm@core@satelliteminimumsize,
161 satellite font/.get=\sm@core@satellitefont,
162 satellite fill opacity/.get=\sm@core@satellitefillopacity,
163 satellite text opacity/.get=\sm@core@satellitertextopacity,
164 satellite text width/.get=\sm@core@satellitertextwidth,
165 distance satellite-connection/.get=\sm@core@satelliteoutersep,
166 connection line width/.get=\sm@core@connectionlinewidth,
167 connection color/.get=\sm@core@connectioncolor,
168 distance planet-satellite/.get=\sm@core@distanceplanetsatellite,
169 priority arrow width/.get=\sm@core@priorityarrowwidth,
170 priority arrow head extend/.get=\sm@core@priorityarrowheadextend,
171 priority tick size/.get=\sm@core@prioritytick,
172 priority arrow height advance/.get=\sm@core@priorityarrowheightadvance,
173 sequence item height/.get=\sm@core@seqitemheight,
174 sequence item width/.get=\sm@core@seqitemwidth,
175 sequence item border color/.get=\sm@core@seqitembordercolor,
176 sequence item border size/.get=\sm@core@seqlinewidth,
177 sequence item font size/.get=\sm@core@seqitemfont,
178 sequence item fill opacity/.get=\sm@core@seqitemfillopacity,
179 sequence item text opacity/.get=\sm@core@seqitemtextopacity,
180 sequence item text width/.get=\sm@core@seqitemtextwidth,
181 sequence item text color/.get=\sm@core@seqitemtextcolor,
182 sequence item uniform color/.get=\sm@core@seqitemuniformcol,
183 arrow style/.get=\sm@core@arrowstyle,
184 text color/.get=\sm@core@textcolor,
185 }%

```

The specific key to set the list of colors:

```

186 \pgfkeys{/smart diagram/.cd, set color list/.code={
187     \foreach \listitem [count=\i] in {#1}{
188         \global\@namedef{color@\i\expandafter}\expandafter{\listitem}
189     }
190 }
191 }
192
193 \pgfkeys{/smart diagram/.cd, uniform color list/.code args={#1 for #2 items}{
194     \foreach \listitem [count=\i] in {1,...,#2}{
195         \global\@namedef{color@\i\expandafter}\expandafter{#1}
196     }
197 }
198 }
199
200 \pgfkeys{/smart diagram/.cd, use predefined color list/.code={
201     \@namedef{color@1}{red!30}
202     \@namedef{color@2}{cyan!30}
203     \@namedef{color@3}{blue!30}

```

```

204 \namedef{color@4}{green!30}
205 \namedef{color@5}{orange!30}
206 \namedef{color@6}{yellow!30}
207 \namedef{color@7}{magenta!30}
208 \namedef{color@8}{brown!30}
209 \namedef{color@9}{violet!30}
210 \namedef{color@10}{teal!30}
211 }
212 }

```

The specific key to disable the back arrow in the `flow diagram` and in the `flow diagram:horizontal`:

```

213 \newif\ifbackarrowdisabled
214 \pgfkeys{/smart diagram/.cd,
215   back arrow disabled/.is if=backarrowdisabled,
216   back arrow disabled=false,
217 }

```

The specific key to disable the final arrow in the `circular diagram` and in the `circular diagram: clockwise`:

```

218 \newif\ifcircularfinalarrowdisabled
219 \pgfkeys{/smart diagram/.cd,
220   circular final arrow disabled/.is if=circularfinalarrowdisabled,
221   circular final arrow disabled=false,
222 }

```

The command to activate the various keys:

```

223 \NewDocumentCommand{\smartdiagramset}{m}{%
224   \pgfkeys{/smart diagram/.cd,#1}%
225   \pgfkeys{/smart diagram/.cd, module minimum width/.get=\sm@core@modulewidth,
226     module minimum height/.get=\sm@core@moduleheight,
227     module y sep/.get=\sm@core@moduleysep,
228     module x sep/.get=\sm@core@modulexsep,
229     descriptive items y sep/.get=\sm@core@descriptiveitemsysep,
230     text width/.get=\sm@core@moduletextwidth,
231     description title width/.get=\sm@core@descriptiontitlewidth,
232     description text width/.get=\sm@core@descriptiontextwidth,
233     description title text width/.get=\sm@core@descriptiontitletextwidth,
234     description title font/.get=\sm@core@descriptiontitlefontsize,
235     description font/.get=\sm@core@descriptionfontsize,
236     description width/.get=\sm@core@descriptionwidth,
237     font/.get=\sm@core@modulefontsize,
238     border color/.get=\sm@core@bordercolor,
239     circular distance/.get=\sm@core@circulardistance,
240     arrow line width/.get=\sm@core@arrowlinewidth,
241     module shape/.get=\sm@core@moduleshape,
242     insert decoration/.get=\sm@core@borderdecoration,
243     arrow tip/.get=\sm@core@arrowtip,
244     arrow color/.get=\sm@core@arrowcolor,
245     bubble center node size/.get=\sm@core@bubblecenternodesize,
246     bubble center node font/.get=\sm@core@bubblecenternodefont,

```

```

247 bubble center node color/.get=\sm@core@bubblecenternodecolor,
248 distance center/other bubbles/.get=\sm@core@distancecenterotherbubbles,
249 distance text center bubble/.get=\sm@core@distancetextcenterbubble,
250 bubble fill opacity/.get=\sm@core@bubblefillopacity,
251 bubble node size/.get=\sm@core@bubblenodesize,
252 bubble text opacity/.get=\sm@core@bubbletextopacity,
253 bubble node font/.get=\sm@core@bubblenodefont,
254 planet size/.get=\sm@core@planetminimumsize,
255 planet color/.get=\sm@core@planetcolor,
256 planet font/.get=\sm@core@planetfont,
257 distance planet-connection/.get=\sm@core@planetoutersep,
258 distance planet-text/.get=\sm@core@planetinnersep,
259 planet text width/.get=\sm@core@planettextwidth,
260 satellite size/.get=\sm@core@satelliteminimumsize,
261 satellite font/.get=\sm@core@satellitefont,
262 satellite fill opacity/.get=\sm@core@satellitefillopacity,
263 satellite text opacity/.get=\sm@core@satellitertextopacity,
264 satellite text width/.get=\sm@core@satellitertextwidth,
265 distance satellite-connection/.get=\sm@core@satelliteoutersep,
266 connection line width/.get=\sm@core@connectionlinewidth,
267 connection color/.get=\sm@core@connectioncolor,
268 distance planet-satellite/.get=\sm@core@distanceplanetsatellite,
269 priority arrow width/.get=\sm@core@priorityarrowwidth,
270 priority arrow head extend/.get=\sm@core@priorityarrowheadextend,
271 priority tick size/.get=\sm@core@prioritytick,
272 priority arrow height advance/.get=\sm@core@priorityarrowheightadvance,
273 sequence item height/.get=\sm@core@seqitemheight,
274 sequence item width/.get=\sm@core@seqitemwidth,
275 sequence item border color/.get=\sm@core@seqitembordercolor,
276 sequence item border size/.get=\sm@core@seqlinewidth,
277 sequence item font size/.get=\sm@core@seqitemfont,
278 sequence item fill opacity/.get=\sm@core@seqitemfillopacity,
279 sequence item text opacity/.get=\sm@core@seqitemtextopacity,
280 sequence item text width/.get=\sm@core@seqitemtextwidth,
281 sequence item text color/.get=\sm@core@seqitemtextcolor,
282 sequence item uniform color/.get=\sm@core@seqitemuniformcol,
283 arrow style/.get=\sm@core@arrowstyle,
284 text color/.get=\sm@core@textcolor,
285 }%
286 }%

```

Key to let the sequence color be uniform:

```

287 \pgfkeys{/smart diagram/.cd,%
288   uniform sequence color/.is choice,%
289   uniform sequence color/true/.code={%
290     \tikzset{sequence item/.append style={
291       fill=\sm@core@seqitemuniformcol,
292     },
293   }
294 },%

```

```

295     uniform sequence color/false/.style={sequence item},%
296     uniform sequence color/.default=false,
297 }%

```

Key to let the connection planet satellite color be uniform:

```

298 \pgfkeys{/smart diagram/.cd,%
299     uniform connection color/.is choice,%
300     uniform connection color/true/.code={%
301         \tikzset{connection planet satellite/.append style={
302             \sm@core@connectioncolor
303         },
304     }
305 },%
306     uniform connection color/false/.style={connection planet satellite},%
307     uniform connection color/.default=false,
308 }%

```

Key to let the arrow color be uniform:

```

309 \pgfkeys{/smart diagram/.cd,%
310     uniform arrow color/.is choice,%
311     uniform arrow color/true/.code={%
312         \tikzset{diagram arrow type/.append style={
313             \sm@core@arrowcolor
314         },
315     }
316 },%
317     uniform arrow color/false/.style={diagram arrow type},%
318     uniform arrow color/.default=false,
319 }%
320

```

The fading style applied to the **priority descriptive diagram** and styles diagram definition:

```

321 \tikzfading[name=priorityarrowfading,
322     bottom color=transparent!5,
323     top color=transparent!80
324 ]
325 \tikzset{priority arrow fill/.style={
326     fill=gray,
327     path fading=priorityarrowfading
328 }
329 }
330
331 \tikzset{module/.style={
332     \pgfkeysvalueof{/smart diagram/module shape},
333     thick,
334     draw=\sm@core@bordercolor,
335     top color=white,
336     bottom color=\col,
337     text=\sm@core@textcolor,
338     text width=\sm@core@moduletextwidth,

```

```

339     minimum width=\sm@core@modulewidth,
340     minimum height=\sm@core@moduleheight,
341     font=\sm@core@modulefontsize,
342     \sm@core@borderdecoration
343 },
344 diagram arrow type/.style={
345     \sm@core@arrowstyle,
346     >=\sm@core@arrowtip,
347     line width=\sm@core@arrowlinewidth,
348     \col
349 },
350 }
351 \tikzset{description title/.style={
352     circle,
353     draw=\sm@core@bordercolor,
354     minimum width=\sm@core@descriptiontitlewidth,
355     anchor=east,
356     bottom color=\col,
357     top color=white!80!\col,
358     font=\sm@core@descriptiontitlefontsize,
359     text width=\sm@core@descriptiontitletextwidth,
360     \sm@core@borderdecoration,
361 },
362 description/.style={
363     \pgfkeysvalueof{/smart diagram/module shape},
364     text width=\sm@core@descriptiontextwidth,
365     draw=\sm@core@bordercolor,
366     anchor=west,
367     minimum height=\sm@core@moduleheight,
368     minimum width=\sm@core@descriptionwidth,
369     bottom color=\col,
370     top color=white!80!\col,
371     font=\sm@core@descriptionfontsize,
372     \sm@core@borderdecoration,
373 }
374 }
375 \tikzset{priority arrow/.style={
376     draw=\sm@core@bordercolor,
377     single arrow,
378     minimum height=\distancemodules,
379     minimum width=\sm@core@priorityarrowwidth,
380     priority arrow fill,
381     rotate=90,
382     single arrow head extend=\sm@core@priorityarrowheadextend,
383     anchor=west,
384 }
385 }
386 \tikzset{bubble center node/.style={
387     minimum size=\sm@core@bubblecenternodesize,
388     circle,

```



```

389     fill=\sm@core@bubblecenternodecolor,
390     font=\sm@core@bubblecenternodefont,
391     outer sep=\sm@core@distancecenterotherbubbles,
392     inner sep=\sm@core@distancetextcenterbubble,
393 },
394 bubble node/.style={
395     minimum size=\sm@core@bubblenodesize,
396     circle,
397     ultra thick,
398     font=\sm@core@bubblenodefont,
399     draw=white,
400     fill opacity=\sm@core@bubblefillopacity,
401     fill=\col,
402     text opacity=\sm@core@bubbletextopacity,
403 }
404 }
405 \tikzset{planet/.style={
406     minimum size=\sm@core@planetminimumsize,
407     circle,
408     fill=\sm@core@planetcolor,
409     font=\sm@core@planetfont,
410     outer sep=\sm@core@planetoutersep,
411     inner sep=\sm@core@planetinnersep,
412     text width=\sm@core@planettextwidth,
413 },
414 satellite/.style={
415     minimum size=\sm@core@satelliteminimumsize,
416     circle,
417     font=\sm@core@satellitefont,
418     fill opacity=\sm@core@satellitefillopacity,
419     fill=\col,
420     text opacity=\sm@core@satellitetextopacity,
421     text width=\sm@core@satellitetextwidth,
422     outer sep=\sm@core@satelliteoutersep,
423 },
424 connection planet satellite/.style={
425     ->,
426     line width=\sm@core@connectionlinewidth,
427     >=\sm@core@arrowtip,
428     \col,
429 }
430 }
431
432 \tikzset{sequence item/.style={
433     minimum height=\sm@core@seqitemheight,
434     minimum width=\sm@core@seqitemwidth,
435     signal,
436     signal from=west,
437     signal to=east,
438     draw=\sm@core@seqitembordercolor,

```

```

439     line width=\sm@core@seqlinewidth,
440     font=\sm@core@seqitemfont,
441     fill opacity=\sm@core@seqitemfillopacity,
442     fill=\col,
443     text opacity=\sm@core@seqitemtextopacity,
444     text width=\sm@core@seqitemtextwidth,
445     text=\sm@core@seqitemtextcolor,
446   }
447 }
448 % let the first word of the item be hyphenate
449 \tikzset{let hyphenation/.style={
450   execute at begin node={%
451     \hspace{0pt}%
452   }
453 }
454 }%

```

The definition of the visibility style:

```

455 \tikzset{
456   sminvisible/.style={opacity=0,text opacity=0},
457   smvisible on/.style={smalt=#1}{sminvisible}},
458   smalt/.code args={<#1>#2#3}{%
459     \alt<#1>{\pgfkeysalso{#2}}{\pgfkeysalso{#3}}
460   },
461 }

```

8.3 Commands

Definition of the two commands. The diagrams:

```

462 \NewDocumentCommand{\smartdiagram}{r[] m}{%
463   \StrCut{#1}{:}\diagramtype\option
464   \IfNoValueTF{#1}{% true-no value 1
465     \PackageError{smartdiagram}%
466     {Type of the diagram not inserted. Please insert it}%
467     {Example: \protect\smartdiagram[flow diagram]}}
468   {%false-no value 1
469   \IfStrEq{\diagramtype}{c}{%
470     \PackageError{smartdiagram}{Type of the diagram not inserted. Please insert it}%
471     {Example: \protect\smartdiagram[flow diagram]}
472   }{
473     \IfStrEq{\diagramtype}{circular diagram}{% true-circular diagram
474     \begin{tikzpicture}[every node/.style={align=center,let hyphenation}]
475
476     \foreach \smitem [count=\xi] in {#2} {\global\let\maxsmitem\xi}
477
478     \foreach \smitem [count=\xi] in {#2}{%
479     \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
480       \pgfmathtruncatemacro{\angle}{180+360/\maxsmitem*\xi}
481     }{% false-clockwise-circular diagram
482       \pgfmathtruncatemacro{\angle}{360/\maxsmitem*\xi}

```

```

483 }
484 \edef\col{\@nameuse{color@\xi}}
485 \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
486   \node[module,drop shadow] (module\xi)
487     at (-\angle:\sm@core@circulardistance) {\smitem };
488 }{% false-clockwise-circular diagram
489   \node[module,drop shadow] (module\xi)
490     at (\angle:\sm@core@circulardistance) {\smitem };
491 }
492 }%
493 \foreach \smitem [count=\xi] in {#2}{%
494   \ifnum\xi=\maxsmitem
495     \ifcircularfinalarrowdisabled
496       \relax
497     \else
498       \pgfmathtruncatemacro\xj{\mod(\xi, \maxsmitem) + 1}
499       \edef\col{\@nameuse{color@\xj}}
500       \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
501         \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm]
502           (module\xj) to[bend right] (module\xi);
503       }{% false-clockwise-circular diagram
504         \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm]
505           (module\xj) to[bend left] (module\xi);
506       }
507     \fi
508   \else
509     \pgfmathtruncatemacro\xj{\mod(\xi, \maxsmitem) + 1}
510     \edef\col{\@nameuse{color@\xj}}
511     \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
512       \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm]
513         (module\xj) to[bend right] (module\xi);
514     }{% false-clockwise-circular diagram
515       \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm]
516         (module\xj) to[bend left] (module\xi);
517     }
518   \fi
519 }%
520 }%
521 \end{tikzpicture}
522 }{% end-circular diagram
523 \IfStrEq{\diagramtype}{flow diagram}{% true-flow diagram
524   \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
525
526   \foreach \smitem [count=\xi] in {#2} {\global\let\maxsmitem\xi}
527
528   \foreach \smitem [count=\xi] in {#2}{%
529     \edef\col{\@nameuse{color@\xi}}
530     \IfStrEq{\option}{horizontal}{% true-horizontal-flow diagram
531       \path let \n1 = {int(0-\xi)}, \n2={0+\xi*\sm@core@modulexsep} in
532         node[module,drop shadow] (module\xi) at +(\n2,0) {\smitem};

```

```

533 }{% false-horizontal-flow diagram
534   \path let \n1 = {int(0-\xi)}, \n2={0-\xi*\sm@core@moduleysep} in
535     node[module,drop shadow] (module\xi) at +(0,\n2) {\smitem};
536 }
537 }%
538
539 \foreach \smitem [count=\xi] in {#2}{%
540   \pgfmathtruncatemacro{xj}{mod(\xi, \maxsmitem) + 1}}
541   \edef\col{\@nameuse{color@\xj}}
542   \ifnum\xi<\maxsmitem
543     \begin{pgfonlayer}{smart diagram arrow back}
544     \draw[diagram arrow type] (module\xj) -- (module\xi);
545     \end{pgfonlayer}
546   \fi
547   % last arrow - not display it in background - check if disabled
548   \ifbackarrowdisabled
549     \relax
550   \else
551     \ifnum\xi=\maxsmitem
552       \IfStrEq{\option}{horizontal}{% true-horizontal-flow diagram
553         \tikzset{square arrow/.style={
554           to path={-- ++(0,0.5) -| (\tikztotarget)}
555         }}
556       }
557       \draw[diagram arrow type, square arrow]
558         (module\xj.north) to (module\xi.north);
559     }{% false-horizontal-flow diagram
560       \tikzset{square arrow/.style={
561         to path={-- ++(0.5,0) |- (\tikztotarget)}
562       }}
563     }
564     \draw[diagram arrow type,square arrow]
565       (module\xj.east) to (module\xi);
566   }
567   \fi
568 }%
569 }%
570 \end{tikzpicture}
571 }{% end-flow diagram
572 \IfStrEq{\diagramtype}{descriptive diagram}{% true-descr. diagram
573 \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
574 \foreach \smitem [count=\xi] in {#2}{%
575 \edef\col{\@nameuse{color@\xi}}
576 \foreach \subitem [count=\xii] in \smitem{%
577 \ifnumequal{\xii}{1}{% true
578 \node[description title,drop shadow]
579   (module-title\xi) at (0,0-\xi*\sm@core@descriptiveitemsysep) {\subitem};
580 }{
581 \ifnumequal{\xii}{2}{% true
582 \node[description,drop shadow](module\xi)

```

```

583   at (0,0-\xi*\sm@core@descriptiveitemsysep) {\subitem};
584 }{}
585 }%
586 }%
587 \end{tikzpicture}
588 }{}% end-descr. diagram
589 \IfStrEq{\diagramtype}{bubble diagram}{% true-bubble diagram
590 \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
591 \foreach \smitem [count=\xi] in {\#2}{\global\let\maxsmitem\xi}
592 \pgfmathtruncatemacro\actualnumitem{\maxsmitem-1}
593 \foreach \smitem [count=\xi] in {\#2}{%
594 \ifnumequal{\xi}{1}{ %true
595 \node[bubble center node](center bubble){\smitem};
596 }{%false
597 \pgfmathtruncatemacro\xj{\xi-1}
598 \pgfmathtruncatemacro\angle{360/\actualnumitem*\xj}
599 \edef\col{\@nameuse{color@\xj}}
600 \node[bubble node] (module\xi)
601   at (center bubble.\angle) {\smitem };
602 }%
603 }%
604 \end{tikzpicture}
605 }{}%end-bubble diagram
606 \IfStrEq{\diagramtype}{constellation diagram}{% true-const diagram
607 \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
608 \foreach \smitem [count=\xi] in {\#2}{\global\let\maxsmitem\xi}
609 \pgfmathtruncatemacro\actualnumitem{\maxsmitem-1}
610 \foreach \smitem [count=\xi] in {\#2}{%
611 \ifnumequal{\xi}{1}{ %true
612 \node[planet](planet){\smitem};
613 }{%false
614 \pgfmathtruncatemacro\xj{\xi-1}
615 \pgfmathtruncatemacro\angle{360/\actualnumitem*\xj}
616 \edef\col{\@nameuse{color@\xj}}
617 \node[satellite] (satellite\xi)
618   at (\angle:\sm@core@distanceplanetsatellite) {\smitem };
619 \draw[connection planet satellite] (planet) -- (satellite\xi);
620 }%
621 }%
622 \end{tikzpicture}
623 }{}%end-const diagram
624 \IfStrEq{\diagramtype}{connected constellation diagram}{% true-conn const diagram
625 \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
626 \foreach \smitem [count=\xi] in {\#2}{\global\let\maxsmitem\xi}
627 \pgfmathtruncatemacro\actualnumitem{\maxsmitem-1}
628 \foreach \smitem [count=\xi] in {\#2}{%
629 \ifnumequal{\xi}{1}{ %true
630 \node[planet](planet){\smitem};
631 }{%false
632 \pgfmathtruncatemacro\xj{\xi-1}

```

```

633 \pgfmathtruncatemacro{\angle}{360/\actualnumitem*\xj}
634 \edef\col{\@nameuse{color@\xj}}
635 \node[satellite] (satellite\xj)
636     at (\angle:\sm@core@distanceplanetsatellite) {\smitem };
637 }%
638 }%
639 \foreach \smitem [count=\xi] in {#2}{%
640     \ifnumgreater{\xi}{1}{ %true
641         \pgfmathtruncatemacro{\xj}{\xi-1}
642         \edef\col{\@nameuse{color@\xj}}
643         \pgfmathtruncatemacro{\xk}{mod(\xj,\actualnumitem) +1}
644         \path[connection planet satellite,-]
645             (satellite\xj) edge[bend right] (satellite\xk);
646     }%
647 }%
648 \end{tikzpicture}
649 }%end-connected constellation diagram
650 \IfStrEq{\diagramtype}{priority descriptive diagram}{% true-priority descriptive diagram
651     \pgfmathparse{subtract(\sm@core@priorityarrowwidth,\sm@core@priorityarrowheadextend)}
652     \pgfmathsetmacro\sm@core@priorityticksize{\pgfmathresult/2}
653     \pgfmathsetmacro\arrowtickxshift{(\sm@core@priorityarrowwidth-\sm@core@priorityticksize)/2}
654     \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
655         \foreach \smitem [count=\xi] in {#2}{\global\let\maxsmitem\xi}
656         \foreach \smitem [count=\xi] in {#2}{%
657             \edef\col{\@nameuse{color@\xi}}
658             \node[description,drop shadow](module\xi)
659                 at (0,0+\xi*\sm@core@descriptiveitemsysep) {\smitem};
660 \draw[line width=\sm@core@prioritytick,\col]
661             ([xshift=-\arrowtickxshift pt]module\xi.base west)--
662             ($([xshift=-\arrowtickxshift pt]module\xi.base west)-(\sm@core@priorityticksize pt,0)$);
663         }%
664         \coordinate (A) at (module1);
665         \coordinate (B) at (module\maxsmitem);
666         \CalcHeight(A,B){heightmodules}
667         \pgfmathadd{\heightmodules}{\sm@core@priorityarrowheightadvance}
668         \pgfmathsetmacro{\distancemodules}{\pgfmathresult}
669         \pgfmathsetmacro\arrowxshift{\sm@core@priorityarrowwidth/2}
670         \begin{pgfonlayer}{background}
671             \node[priority arrow] at ([xshift=-\arrowxshift pt]module1.south west){};
672         \end{pgfonlayer}
673     \end{tikzpicture}
674 }%end-priority descriptive diagram
675 \IfStrEq{\diagramtype}{sequence diagram}{% true-sequence diagram
676     \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
677         \foreach \x[count=\xi, count=\prevx from 0] in {#2}{%
678             \edef\col{\@nameuse{color@\xi}}
679             \ifnum\xi=1
680                 \node[sequence item] (sequence-item\xi) {\x};
681             \else
682                 \node[sequence item,anchor=west] (sequence-item\xi) at (sequence-item\prevx.east) {\x};

```

```

683 \fi
684 }
685 \end{tikzpicture}
686 }{}% end-sequence diagram
687 }% end-no value 1
688 }% end-command

```

The command definition for the animated diagrams:

```

689 \NewDocumentCommand{\smartdiagramanimated}{r[] m}{%
690   \StrCut{#1}{:}\diagramtype\option
691   \IfNoValueTF{#1}{% true-no value 1
692     \PackageError{smartdiagram}{Type of the diagram not inserted. Please insert it}
693     {Example: \protect\smartdiagram[flow diagram]}}
694   {%false-no value 1
695   \IfStrEq{\diagramtype}{}{%
696     \PackageError{smartdiagram}{Type of the diagram not inserted. Please insert it}
697     {Example: \protect\smartdiagram[flow diagram]}}
698   }{}
699   \IfStrEq{\diagramtype}{circular diagram}{% true-circular diagram
700   \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
701   \foreach \smitem [count=\xi] in {#2} {\global\let\maxsmitem\xi}
702   \foreach \smitem [count=\xi] in {#2}{%
703   \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
704     \pgfmathtruncatemacro{\angle}{180+360/\maxsmitem*\xi}
705   }{% false-clockwise-circular diagram
706     \pgfmathtruncatemacro{\angle}{360/\maxsmitem*\xi}
707   }
708   \edef\col{\@nameuse{color@\xi}}
709   \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
710     \node[module,
711       drop shadow={smvisible on=<\xi->},
712       smvisible on=<\xi->] (module\xi)
713       at (-\angle:\sm@core@circulardistance) {\smitem};
714   }{% false-clockwise-circular diagram
715     \node[module,
716       drop shadow={smvisible on=<\xi->},
717       smvisible on=<\xi->] (module\xi)
718       at (\angle:\sm@core@circulardistance) {\smitem};
719     }
720   }%
721   \foreach \smitem [count=\xi] in {#2}{%
722   \ifnum\xi=\maxsmitem
723     \ifcircularfinalarrowdisabled
724       \relax
725     \else
726       \pgfmathtruncatemacro{\xj}{mod(\xi, \maxsmitem) + 1)}
727       \pgfmathtruncatemacro{\adv}{\xi + 1)}
728       \edef\col{\@nameuse{color@\xj}}
729       \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
730         \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm]

```

```

731         (module\xj) to[bend right] (module\xi);
732     }{% false-clockwise-circular diagram
733         \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm]
734         (module\xj) to[bend left] (module\xi);
735     }
736     \fi
737 \else
738     \pgfmathtruncatemacro{\xj}{mod(\xi, \maxsmitem) + 1)}
739     \pgfmathtruncatemacro{\adv}{\xi + 1)}
740     \edef\col{\@nameuse{color@\xj}}
741     \IfStrEq{\option}{clockwise}{% true-clockwise-circular diagram
742         \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm,
743             smvisible on=<\adv->]
744             (module\xj) to[bend right] (module\xi);
745     }{% false-clockwise-circular diagram
746         \draw[diagram arrow type,shorten <=0.3cm,shorten >=0.3cm,
747             smvisible on=<\adv->]
748             (module\xj) to[bend left] (module\xi);
749     }
750
751     \fi
752 }%
753 \end{tikzpicture}
754 }{% end-circular diagram
755 \IfStrEq{\diagramtype}{flow diagram}{% true-flow diagram
756     \begin{tikzpicture}[every node/.style={align=center,let hyphenation}]
757
758     \foreach \smitem [count=\xi] in {\#2} {\global\let\maxsmitem\xi}
759
760     \foreach \smitem [count=\xi] in {\#2}{%
761         \edef\col{\@nameuse{color@\xi}}
762         \IfStrEq{\option}{horizontal}{% true-horizontal-flow diagram
763             \path let \n1 = {int(0-\xi)}, \n2={0+\xi*\sm@core@modulexsep}
764                 in node[module,drop shadow={smvisible on=<\xi->},
765                     smvisible on=<\xi->] (module\xi) at +(\n2,0) {\smitem};
766         }{% false-horizontal-flow diagram
767             \path let \n1 = {int(0-\xi)}, \n2={0-\xi*\sm@core@moduleysep}
768                 in node[module,drop shadow={smvisible on=<\xi->},
769                     smvisible on=<\xi->] (module\xi) at +(0,\n2) {\smitem};
770         }
771     }%
772
773     \foreach \smitem [count=\xi] in {\#2}{%
774         \pgfmathtruncatemacro{\xj}{mod(\xi, \maxsmitem) + 1)}
775         \edef\col{\@nameuse{color@\xj}}
776         \ifnum\xi<\maxsmitem
777             \begin{pgfonlayer}{smart diagram arrow back}
778             \draw[diagram arrow type,smvisible on=<\xi->]
779                 (module\xj) -- (module\xi);
780             \end{pgfonlayer}

```



```

781 \fi
782 % last arrow - not display it in background - check if disabled
783 \ifbackarrowdisabled
784 \relax
785 \else
786 \ifnum\xi=\maxsmitem
787 \IfStrEq{\option}{horizontal}{% true-horizontal-flow diagram
788 \tikzset{square arrow/.style={
789 to path={-- ++(0,0.5) -| (\tikztotarget)}}
790 }
791 }
792 \draw[diagram arrow type, square arrow,smvisible on=<\xi->]
793 (module\xj.north) to (module\xi.north);
794 }{% false-horizontal-flow diagram
795 \tikzset{square arrow/.style={
796 to path={-- ++(0.5,0) |- (\tikztotarget)}}
797 }
798 }
799 \draw[diagram arrow type,square arrow,smvisible on=<\xi->]
800 (module\xj.east) to (module\xi);
801 }
802 \fi
803 \fi
804 }%
805 \end{tikzpicture}
806 }}% end-flow diagram
807 \IfStrEq{\diagramtype}{descriptive diagram}{% true-descriptive diagram
808 \begin{tikzpicture}[every node/.style={align=center,let hyphenation}]
809 \foreach \smitem [count=\xi] in {#2}{%
810 \edef\col{\@nameuse{color@\xi}}
811
812 \foreach \subitem [count=\xii] in \smitem{%
813 \pgfmathtruncatemacro\subitemvisible{\xi}
814 \ifnumequal{\xii}{1}{% true
815 \node[description title,drop shadow, smvisible on=<\subitemvisible->]
816 (module-title\xi) at (0,0-\xi*\sm@core@descriptiveitemsysep) {\subitem};\pause
817 }{
818 \ifnumequal{\xii}{2}{% true
819 \node[description,drop shadow,smvisible on=<\subitemvisible->]
820 (module\xi)at (0,0-\xi*\sm@core@descriptiveitemsysep) {\subitem};\pause
821 }{
822 }%
823 }%
824 \end{tikzpicture}
825 }}% end-descriptive diagram
826 \IfStrEq{\diagramtype}{bubble diagram}{% true-bubble diagram
827 \begin{tikzpicture}[every node/.style={align=center,let hyphenation}]
828 \foreach \smitem [count=\xi] in {#2}{\global\let\maxsmitem\xi
829 \pgfmathtruncatemacro\actualnumitem{\maxsmitem-1}
830 \foreach \smitem [count=\xi] in {#2}{%

```

```

831 \ifnumequal{\xi}{1}{ %true
832 \node[bubble center node, smvisible on=<\xi->](center bubble){\smitem};
833 }{%false
834 \pgfmathtruncatemacro{\xj}{\xi-1}
835 \pgfmathtruncatemacro{\angle}{360/\actualnumitem*\xj}
836 \edef\col{\@nameuse{color@\xj}}
837 \node[bubble node, smvisible on=<\xi->](module\xi)
838     at (center bubble.\angle) {\smitem };
839 }%
840 }%
841 \end{tikzpicture}
842 }{%end-bubble diagram
843 \IfStrEq{\diagramtype}{constellation diagram}{% true-const diagram
844 \begin{tikzpicture}[every node/.style={align=center,let hyphenation}]
845 \foreach \smitem [count=\xi] in {\#2}{\global\let\maxsmitem\xi}
846 \pgfmathtruncatemacro\actualnumitem{\maxsmitem-1}
847 \foreach \smitem [count=\xi] in {\#2}{%
848 \ifnumequal{\xi}{1}{ %true
849 \node[planet, smvisible on=<\xi->](planet){\smitem};
850 }{%false
851 \pgfmathtruncatemacro{\xj}{\xi-1}
852 \pgfmathtruncatemacro{\angle}{360/\actualnumitem*\xj}
853 \edef\col{\@nameuse{color@\xj}}
854 \node[satellite, smvisible on=<\xi->] (satellite\xi)
855     at (\angle:\sm@core@distanceplanetsatellite) {\smitem };
856 \draw[connection planet satellite, smvisible on=<\xi->]
857     (planet) -- (satellite\xi);
858 }%
859 }%
860 \end{tikzpicture}
861 }{%end-constellation diagram
862 \IfStrEq{\diagramtype}{connected constellation diagram}{% true-conn const diagram
863 \begin{tikzpicture}[every node/.style={align=center,let hyphenation}]
864 \foreach \smitem [count=\xi] in {\#2}{\global\let\maxsmitem\xi}
865 \pgfmathtruncatemacro\actualnumitem{\maxsmitem-1}
866 \foreach \smitem [count=\xi] in {\#2}{%
867 \ifnumequal{\xi}{1}{ %true
868 \node[planet,smvisible on=<\xi->](planet){\smitem};
869 }{%false
870 \pgfmathtruncatemacro{\xj}{\xi-1}
871 \pgfmathtruncatemacro{\angle}{360/\actualnumitem*\xj}
872 \edef\col{\@nameuse{color@\xj}}
873 \node[satellite,smvisible on=<\xi->] (satellite\xj)
874     at (\angle:\sm@core@distanceplanetsatellite) {\smitem };
875 }%
876 }%
877 \foreach \smitem [count=\xi] in {\#2}{%
878     \ifnumgreater{\xi}{1}{ %true
879         \pgfmathtruncatemacro{\xj}{\xi-1}
880         \edef\col{\@nameuse{color@\xj}}

```

```

881     \pgfmathtruncatemacro{\xk}{mod(\xj,\actualnumitem) +1}
882     \pgfmathtruncatemacro{\smvisible}{\xi+1}
883     \path[connection planet satellite,-,smvisible on=<\smvisible->]
884       (satellite\xj) edge[bend right] (satellite\xk);
885   }{}
886 }%
887 \end{tikzpicture}
888 }{}%end-connected constellation diagram
889 \IfStrEq{\diagramtype}{priority descriptive diagram}{% true-priority descriptive diagram
890   \pgfmathparse{subtract(\sm@core@priorityarrowwidth,\sm@core@priorityarrowheadextend)}
891   \pgfmathsetmacro{\sm@core@priorityticksize}{\pgfmathresult/2}
892   \pgfmathsetmacro{\arrowtickxshift}{(\sm@core@priorityarrowwidth-\sm@core@priorityticksize)/2}
893   \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
894     \foreach \smitem [count=\xi] in {#2}{\global\let\maxsmitem\xi}
895     \foreach \smitem [count=\xi] in {#2}{%
896       \edef\col{\@nameuse{color@\xi}}
897       \pgfmathtruncatemacro{\smvisible}{\xi+1}
898       \node[description,drop shadow={smvisible on=<\smvisible->},smvisible on=<\smvisible->]
899         (module\xi) at (0,0+\xi*\sm@core@descriptiveitemsysep) {\smitem};
900 \draw[line width=\sm@core@prioritytick,\col,smvisible on=<\smvisible->]
901   ([xshift=-\arrowtickxshift pt]module\xi.base west)--
902   ($([xshift=-\arrowtickxshift pt]module\xi.base west)-(\sm@core@priorityticksize pt,0)$);
903   }%
904   \coordinate (A) at (module1);
905   \coordinate (B) at (module\maxsmitem);
906   \CalcHeight(A,B){heightmodules}
907   \pgfmathadd{\heightmodules}{\sm@core@priorityarrowheightadvance}
908   \pgfmathsetmacro{\distancemodules}{\pgfmathresult}
909   \pgfmathsetmacro{\arrowxshift}{\sm@core@priorityarrowwidth/2}
910   \begin{pgfonlayer}{background}
911     \node[priority arrow] at ([xshift=-\arrowxshift pt]module1.south west){};
912   \end{pgfonlayer}
913   \end{tikzpicture}
914 }{}% end-priority descriptive diagram
915 \IfStrEq{\diagramtype}{sequence diagram}{% true-sequence diagram
916   \begin{tikzpicture}[every node/.style={align=center,let hypenation}]
917     \foreach \x[count=\xi, count=\prevx from 0] in {#2}{%
918       \edef\col{\@nameuse{color@\xi}}
919       \ifnum\xi=1
920         \node[sequence item,smvisible on=<\xi->] (sequence-item\xi) {\x};
921       \else
922         \node[sequence item,anchor=west,smvisible on=<\xi->]
923           (sequence-item\xi) at (sequence-item\prevx.east) {\x};
924       \fi
925     }
926   \end{tikzpicture}
927 }{}% end-sequence diagram
928 }% end-no value 1
929 }% end-command

```

8.4 Library Additions

The library at first loads the TikZ library `positioning`.

```
930 \usetikzlibrary{positioning}
```

Key definition:

```
931 \pgfkeys{/smart diagram/additions/.cd,
932   additional item shape/.initial=\pgfkeysvalueof{/smart diagram/module shape},
933   additional item shape/.get=\sm@additions@additionalitemshape,
934   additional item shape/.store in=\sm@additions@additionalitemshape,
935   additional item border color/.initial=none,
936   additional item border color/.get=\sm@additions@additionalitembordercolor,
937   additional item border color/.store in=\sm@additions@additionalitembordercolor,
938   additional item bottom color/.initial=white,
939   additional item bottom color/.get=\sm@additions@additionalitembottomcolor,
940   additional item bottom color/.store in=\sm@additions@additionalitembottomcolor,
941   additional item fill color/.initial=none,
942   additional item fill color/.get=\sm@additions@additionalitemfillcolor,
943   additional item fill color/.store in=\sm@additions@additionalitemfillcolor,
944   additional item text width/.initial=1.75cm,
945   additional item text width/.get=\sm@additions@additionalitemtextwidth,
946   additional item text width/.store in=\sm@additions@additionalitemtextwidth,
947   additional item width/.initial=2cm,
948   additional item width/.get=\sm@additions@additionalitemwidth,
949   additional item width/.store in=\sm@additions@additionalitemwidth,
950   additional item height/.initial=1cm,
951   additional item height/.get=\sm@additions@additionalitemheight,
952   additional item height/.store in=\sm@additions@additionalitemheight,
953   additional item font/.initial=\normalfont,
954   additional item font/.get=\sm@additions@additionalitemfont,
955   additional item font/.store in=\sm@additions@additionalitemfont,
956   additional item border decoration/.initial={},
957   additional item border decoration/.get=\sm@additions@additionalitemdecoration,
958   additional item border decoration/.store in=\sm@additions@additionalitemdecoration,
959   additional item offset/.initial={0.25cm},
960   additional item offset/.get=\sm@additions@additionalitemoffset,
961   additional item offset/.store in=\sm@additions@additionalitemoffset,
962   additional item fill opacity/.initial={1},
963   additional item fill opacity/.get=\sm@additions@additionalitemfilloperacy,
964   additional item fill opacity/.store in=\sm@additions@additionalitemfilloperacy,
965   additional item text opacity/.initial={1},
966   additional item text opacity/.get=\sm@additions@additionalitemtextopacity,
967   additional item text opacity/.store in=\sm@additions@additionalitemtextopacity,
968   additional arrow tip/.initial={stealth},
969   additional arrow tip/.get=\sm@additions@additionalarrowtip,
970   additional arrow tip/.store in=\sm@additions@additionalarrowtip,
971   additional arrow line width/.initial={0.1cm},
972   additional arrow line width/.get=\sm@additions@additionalarrowlinewidth,
973   additional arrow line width/.store in=\sm@additions@additionalarrowlinewidth,
974   additional arrow color/.initial={gray},
```

```

975 additional arrow color/.get=\sm@additions@additionalarrowcolor,
976 additional arrow color/.store in=\sm@additions@additionalarrowcolor,
977 additional arrow style/.initial={->},
978 additional arrow style/.get=\sm@additions@additionalarrowstyle,
979 additional arrow style/.store in=\sm@additions@additionalarrowstyle,
980 additional item shadow/.initial={},
981 additional item shadow/.get=\sm@additions@additionalitemshadow,
982 additional item shadow/.store in=\sm@additions@additionalitemshadow,
983 }
984
985 \newif\ifconnectionsdisabled
986 \pgfkeys{/smart diagram/additions/.cd,
987   additional connections disabled/.is if=connectionsdisabled,
988   additional connections disabled=true,
989 }
990
991 \pgfkeys{/smart diagram/.cd,
992   additions/.style={/smart diagram/additions/.cd,#1}%
993 }

```

Style definition; the additional item style comprises lot of usual TikZ options: it possible to select a coloring with a vertical shading or an uniform filling.

```

994 \tikzset{additional item/.style={
995   align=center,
996   \sm@additions@additionalitemshape,
997   thick,
998   draw=\sm@additions@additionalitembordercolor,
999   top color=white,
1000  bottom color=\sm@additions@additionalitembottomcolor,
1001  postaction={fill=\sm@additions@additionalitemfillcolor},
1002  text width=\sm@additions@additionalitemtextwidth,
1003  minimum width=\sm@additions@additionalitemwidth,
1004  minimum height=\sm@additions@additionalitemheight,
1005  font=\sm@additions@additionalitemfont,
1006  fill opacity=\sm@additions@additionalitemfillopacity,
1007  text opacity=\sm@additions@additionalitemtextopacity,
1008  \sm@additions@additionalitemshadow,
1009  \sm@additions@additionalitemdecoration
1010 },
1011  additional item arrow type/.style={
1012    \sm@additions@additionalarrowstyle,
1013    >=\sm@additions@additionalarrowtip,
1014    line width=\sm@additions@additionalarrowlinewidth,
1015    \sm@additions@additionalarrowcolor
1016  },
1017 }

```

Command definition; at first the diagram is created with the usual command, then the foreach iterates in order to get additions. The additions' strings are cut by means of the package xstring and its macro \StrCut. Of course, to all the tikzpictures, the option remember picture is added.

```

1018 \NewDocumentCommand{\smartdiagramadd}{r[] m m}{
1019 \tikzstyle{every picture}+=[remember picture]
1020 \smartdiagram[#1]{#2}
1021 \begin{tikzpicture}[remember picture,overlay,
1022   every node/.style={align=center,let hyphenation}]
1023 \foreach \smitem [count=\xi] in {#2} {\global\let\numitems\xi}
1024 \foreach \smitem[count=\xi] in {#3}{
1025 \StrCut{\smitem}{/}\pos\textitem
1026 \StrCut{\pos}{\space of\space}\point\modulenum
1027 \node[additional item,
1028   \point=\sm@additions@additionalitemoffset of \modulenum]
1029   (additional-module\xi) {\textitem};
1030 \ifconnectionsdisabled
1031   \relax
1032 \else
1033   \begin{pgfonlayer}{smart diagram arrow back}
1034     \draw[additional item arrow type]
1035       (additional-module\xi) -- (\modulenum);
1036   \end{pgfonlayer}
1037 \fi
1038 }
1039 \end{tikzpicture}
1040 }

```

Definition of the command to connect additions with diagram modules:

```

1041 \NewDocumentCommand{\smartdiagramconnect}{m m}{%
1042   \begin{tikzpicture}[remember picture,overlay]
1043     \foreach \start/\end in {#2}
1044       \draw[additional item arrow type,#1]
1045         (\start) -- (\end);
1046   \end{tikzpicture}
1047 }

```