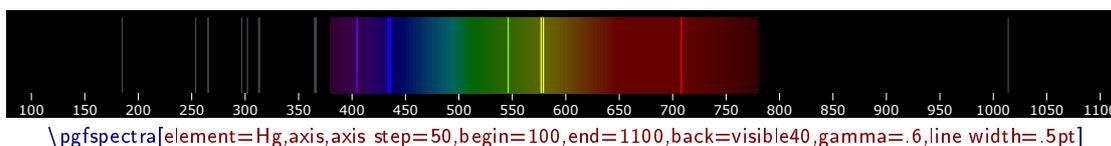


Manual for pgf-spectra 2.1.2

Hugo Gomes
hugo.parelho@gmail.com

3rd July 2021



Abstract

The purpose of this package is to draw the spectrum of elements in a simple way. It's based on the package *pst-spectra*, but with some extra options. It relies on the pgf/TikZ to draw the desired spectrum, continuous or discrete. As in *pst-spectra* there are data available for the spectra of 98 elements and their ions. It also allows the user to draw a spectrum with their own personal data.

In version 2.0.0 the previous data of the visible region was extended to include lines from Extreme UV to Near IR ($10\text{ nm} \leq \lambda \leq 4000\text{ nm}$). See section *The lines data* below for more information.

Also in version 2.0.0 the possibility to redshift the lines of spectra was introduced, by entering directly the redshift value or the velocity and the angle to compute the redshift value (Doppler Redshift).

In version 2.1.0 a new color conversion (correlated color temperature), shadings for use with TikZ and/or PGFPLOTS and a color map for use with PGFPLOTS were introduced.

In version 2.1.1 two new options are available in the shading for use with PGFPLOTS (or TikZ): shade begin and logarithmic (which builds the shading in a logarithmic scale).

In this release – version 2.1.2 – only a small fix was made (wrong char encoding in style definition file – pgf-spectra.sty).

Contents

Installation and usage	1
What's new	2
In version 2.1.1	2
In version 2.1.0	2
In version 2.0.0	3
The lines data	4
1 Data based on pst-spectra	4
2 Data based on NIST	4
The commands	5
\pgfspectra	5
\wcolor{wavelength}	5
\pgfspectraStyle[options]	6
\pgfspectraStyleReset	6
\tempercolor{Kelvin}	7
\pgfspectrashade[<h v>](start,end){name}	7
\pgfspectraplotshade[options]{name}	8
\pgfspectraplotmap[<l h>]{name}	9
\pgfspectrarainbow[<[tikz options]><(rainbow options)>]{radius}	10
The options for \pgfspectra	11
~> use visible shading	11
~> width	11
~> height	11
~> element	11
~> charge	12
~> Imin	12
~> relative intensity	12
~> relative intensity threshold	12
~> line intensity	13
~> gamma	14
~> brightness	14
~> back	15
~> backIRUV	15
~> IRcolor	15
~> UVcolor	16
~> lines	16
~> line width	16
~> begin	16
~> end	17
~> absorption	17
~> axis	17
~> axis step	18
~> axis ticks	18
~> axis unit	18
~> axis unit precision	19
~> axis color	19
~> axis font	20
~> axis font color	20
~> label	20
~> label position	20
~> label font	20

~> label font color	21
~> label before text	21
~> label after text	21
~> redshift	21
~> show redshift value	22
The options for <code>\pgfspectraplotshade</code>	23
~> shade begin	23
~> shade end	23
~> shade opacity	24
~> shade opacity color	24
~> logarithmic	25
The options for <code>\pgfspectrarainbow</code>	26
~> rainbow start	26
~> rainbow knock out	26
~> rainbow fade	27
~> rainbow transparency	27
~> rainbow background	27
Examples	29
Using <code>\pgfspectra</code>	29
Using <code>\pgfspectrashade</code> in <code>TikZ</code>	32
Using <code>\pgfspectraplotshade</code> and <code>\pgfspectraplotmap</code> with <code>PGFPLOTS</code>	33
Using <code>\pgfspectrarainbow</code>	50
Alphabetical list of available options	52
Recommendations and known issues	54
The code	54

Installation and usage

pgf-spectra is placed under the terms of the L^AT_EX Project Public License, version 1.3 or later (<http://www.latex-project.org/lppl.txt>). pgf-spectra loads and only requires the package `TikZ`.

You need to put the style file (pgf-spectra.sty) in a location where PDF^LA_TE_X can find them. According to the TDS conventions this may be a subdirectory named tex/latex/pgfspectra/ or tex/latex/misc/ in your (site specific) installation tree (insert your appropriate directory delimiter instead of /, if needed).

If you are using PDF^LA_TE_X, just can simply include the style file without any option via the `\usepackage` command, `\usepackage{pgf-spectra}`

It can also be loaded with *one option* to select the data source:
`\usepackage[option]{pgf-spectra}`

For more detailed information see section [The lines data](#).

What's new

► In version 2.1.1

- Code rewrote for the command `\pgfspectraplotshade`, supporting two new keys:
 - shade begin
 - logarithmic

► In version 2.1.0

- The continuous visible region is now drawn via TikZ shading, improving a little bit the speed of the whole process.
- Minor fix: the width of the emission/absorption lines are now correctly drawn.
- New keys for `\pgfspectra`:
 - use visible shading
 - backVIS
 - axis unit
 - axis unit precision
- New color conversion command, which converts a temperature in Kelvin to the correspondent rgb color, based on correlated color temperature:
 - `\tempercolor{temperature in Kelvin}`
- New commands that provides shadings to use in TikZ:
 - `\pgfspectrashade[<h|v>](start,end){name}`
 - `\pgfspectrarainbow[<tikz options>]><(rainbow options)>{radius}`

The TikZ keys that affect the rainbow are:

 - * color
 - * opacity
 - * scope fading

The specific rainbow options are:

 - * rainbow fade
 - * rainbow start
 - * rainbow knock out
 - * rainbow background
 - * rainbow transparency
- New command that provides a shading to use in PGFPLOTS:
 - `\pgfspectraplotshade[options]{name}` with the following specific keys
 - * shade end
 - * shade opacity
 - * shade opacity color
- New command that builds a color map to use in PGFPLOTS:
 - `\pgfspectraplotmap[<|h>]{name}`

► In version 2.0.0

- The package can now be loaded with one of the following options:
 - `\usepackage[NIST]{pgf-spectra}` (**default**)
 - `\usepackage[LSE]{pgf-spectra}`
- Range of spectral window from 10 nm to 4000 nm (previous version from 380 nm to 780 nm).
- Added the lines data outside the visible range for the 98 elements.
- No more dependency on the package `ifthen` (code rewritten with the `\ifx` $\text{T}_{\text{E}}\text{X}$ primitive).
- Setting/disabling global options to draw the spectra's with the new commands:
 - `\pgfspectraStyle[options]`
 - `\pgfspectraStyleReset`
- New keys:
 - axis ticks
 - backIRUV (only for emission spectrum)
 - IRcolor (for emission lines and for background in absorption spectrum)
 - UVcolor (for emission lines and for background in absorption spectrum)
 - redshift
 - show redshift value
- The issues with the zooming of the pdf viewer sometimes introducing blank lines in the spectra have been fixed:



The rendition should now be working for every zoom (I hope!):

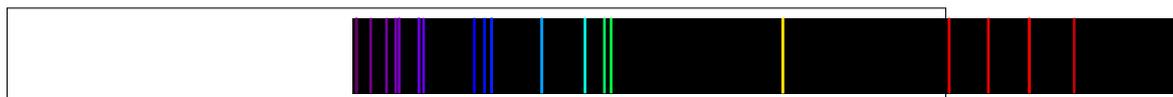


Many thanks to *Daniel García's* suggestion to solve this problem!

- Fixed the problem when putting the spectra inside any horizontal $\text{T}_{\text{E}}\text{X}$ box, like `\makebox`, `\mbox` or `\hbox`.

For instance, the code `\makebox[\textwidth][c]{\pgfspectra[element=He]}`:

- had as a result in the previous version (version 1.0):



- and will result in current version (2.0.0) at:



The lines data

There are two data sets available for drawing the spectra: one based in the previous version, whose data was initially obtained from the package `pst-spectra` and the other obtained from `NIST`.

In both cases are included the lines for 98 elements, from hydrogen ($Z = 1$) to einsteinium ($Z = 99$), except for francium ($Z = 87$). For each element there are lines between 10 nm and 4000 nm (obtained from the referred pages at February 2021).

1 Data based on `pst-spectra`

This set was obtained from <http://cdsarc.u-strasbg.fr/viz-bin/Cat?VI/16>

According to the information on the page the listed lines are based on "Line Spectra of the Elements", Joseph Reader and Charles H. Corliss CRC Handbook of Chemistry and Physics. This book refers that «The table contains the outstanding spectral lines of neutral (*I*) and singly ionized (*II*) atoms of the elements from hydrogen through plutonium ($Z = 1 - 94$); selected strong lines from doubly ionized (*III*), triply ionized (*IV*), and quadruply ionized (*V*) atoms are also included.»

Note: `pst-spectra` documentation refers "*Line Spectra of the Elements from the Astronomical Data Center of NASA*" as the source material, but I'm assuming the original source is "*Line Spectra of the Elements*", Joseph Reader and Charles H. Corliss CRC Handbook of Chemistry and Physics, obtained from <http://cdsarc.u-strasbg.fr/viz-bin/Cat?VI/16>.

To use this data set load the package `pgf-spectra` with the option `LSE` (acronym to Line Spectra of the Elements):

```
\usepackage[LSE]{pgf-spectra}
```

Number of lines provided: 46065 (see file `pgf-spectraDataLSE.pdf`)

2 Data based on `NIST`

This set was obtained from <https://physics.nist.gov/PhysRefData/Handbook/Tables/findinglist.htm>

According to the information on the page the listed lines «includes data for the neutral and singly-ionized atoms».

Note: **This set is loaded by default.** Because the data to search is slightly smaller (only neutral and singly-ionized atoms) the time consumption when building the spectra could be a bit lower.

To use this data set load the package `pgf-spectra` without options:

```
\usepackage{pgf-spectra}
```

Number of lines provided: 11980 (see file `pgf-spectraDataNIST.pdf`);

The commands

The four *main* commands, those related with this package itself, are:

- `\pgfspectra` or `\pgfspectra[options list]`
- `\wcolor{wavelength}`
- `\pgfspectraStyle[options]`
- `\pgfspectraStyleReset`

There are other four commands to use with TikZ and/or PGFPLOTS:

- `\tempercolor{Kelvin}`
- `\pgfspectrashade[<h|v>](start,end){name}`
- `\pgfspectraplotshade[options]{name}`
- `\pgfspectraplotmap[<|h>]{name}`

And finally *just for fun* a command that draws a rainbow:

- `\pgfspectrarainbow[<tikz options>]><(rainbow options)>{radius}`

► Utilization of `\pgfspectra`

This command is used without options to draw the visible continuous spectrum:

```
\pgfspectra
```



When using options, a continuous or discrete spectra in the visible region can be drawn, for instance:

```
\pgfspectra[width=.5\textwidth,height=1.25cm]
```



```
\pgfspectra[width=\textwidth,element=Ne]
```

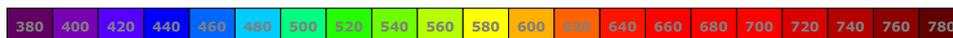


The options available for `\pgfspectra` are described in section [The options for \pgfspectra](#).

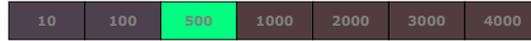
► Utilization of `\wcolor{wavelength}`

A command to convert a wavelength from 380 to 780 nanometres (or other value in the range $10\text{ nm} \leq \lambda \leq 4000\text{ nm}$) to the respective color available as 'wcolor':

```
\tikz{\foreach \x in {380,400,...,780}{\wcolor{\x}
  \draw[fill=wcolor] (.03*\x,0) rectangle ++(.6,.5)
  node[midway,font=\tiny\bfseries,text=black!50] {\x};}}
```



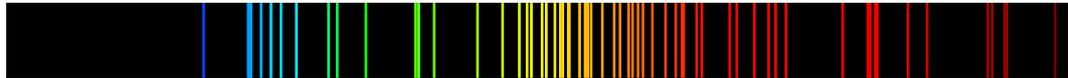
```
\tikz{\foreach \x/\y in {10/0,100/1,500/2,1000/3,2000/4,3000/5,4000/6}{\wlcolor{\x}
\draw[fill=wlcolor] (\y,0) rectangle ++(1,.5)
node[midway,font=\tiny\bfseries,text=black!50] {\x};}}
```



► Utilization of `\pgfspectraStyle[options]`

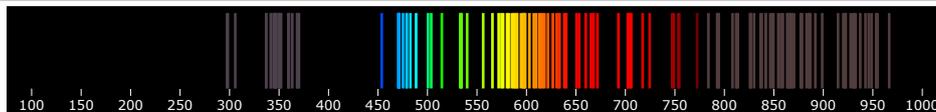
Use this command to set the global style of all the subsequent drawn spectra:

```
\pgfspectra[element=Ne] (before defining the global style)
```

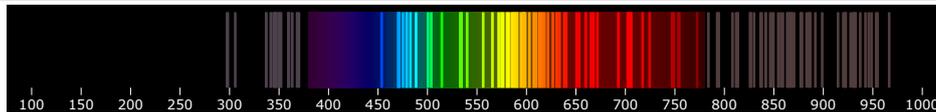


```
\pgfspectraStyle[width=.75\textwidth,axis,begin=100,end=1000,axis step=50]
```

```
\pgfspectra[element=Ne] (after defining the global style)
```

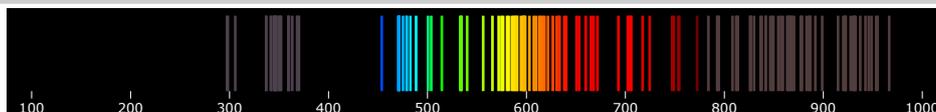


```
\pgfspectra[element=Ne,back=visible40,gamma=.6] (adding other options)
```



Note you can locally override the settings defined in the global style:

```
\pgfspectra[element=Ne,axis step=100] (overriding a global option)
```

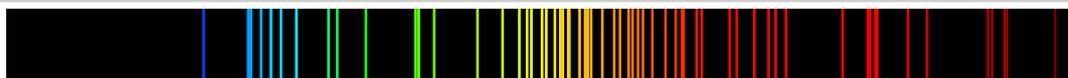


► Utilization of `\pgfspectraStyleReset`

Used to reset all the options of the spectra to their default values:

```
\pgfspectraStyleReset
```

```
\pgfspectra[element=Ne]
```

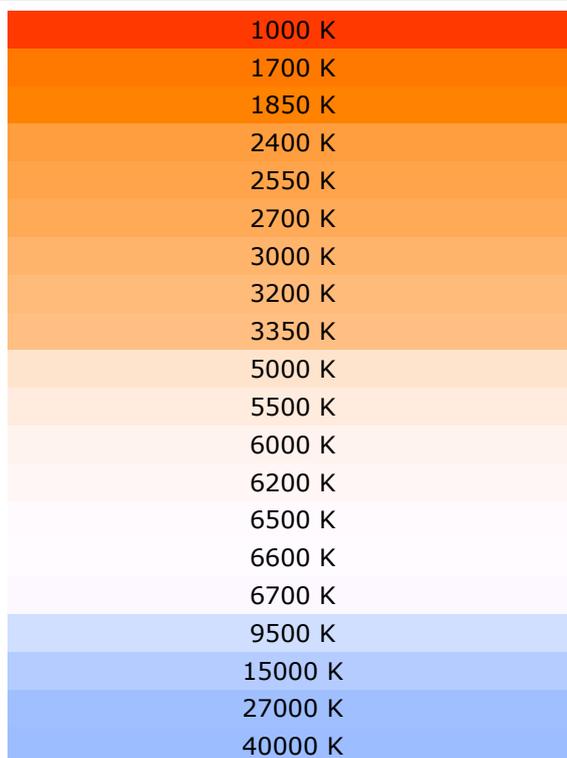


► Utilization of `\tempercolor{Kelvin}`

A command that uses the CIE 1964 10-degree color matching function to convert a given temperature, in Kelvin ($1000\text{ K} \leq T \leq 40000\text{ K}$), to the respective correlated color. For more information on the implemented algorithm, please see:

- <https://tannerhelland.com/2012/09/18/convert-temperature-rgb-algorithm-code.html>
- <https://www.zombieprototypes.com/?p=210>
- <https://github.com/neilbartlett/color-temperature>

```
\foreach \T in {1000,1700,1850,2400,2550,2700,3000,3200,%
  3350,5000,5500,6000,6200,6500,6600,6700,9500,15000,%
  27000,40000}
{\tempercolor{\T}\tikz{
  \fill[tempercolor,font=\small] (0,0) rectangle (7.5,.5) %
  node[midway] {\color{black}\T K};}\ [-1pt]%
}%
```



► Utilization of `\pgfspectrashade[<h|v>](start,end){name}`

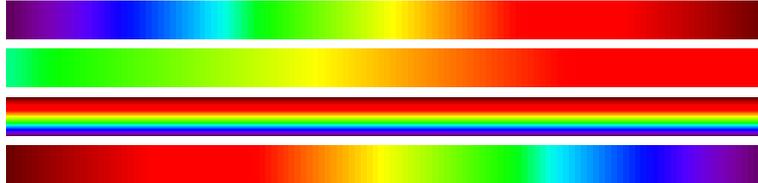
This command builds and makes available a **h**orizontal or a **v**ertical shading, between the `'start'` and `'end'` wavelengths (in nanometres), to use in TikZ pictures with the provided `'name'`.

Note that, in this command, the 'start' wavelength needs to be smaller than the 'end' wavelength and is in the visible region: $\lambda_{\text{start}} < \lambda_{\text{end}}$ and $380 \leq \lambda \leq 780$.

The optional parameter takes the value **h** or **v** and has the default value of **h**.

```
\pgfspectrashade(380,780){myShadeA}
\pgfspectrashade(500,700){myShadeB}
\pgfspectrashade[v](380,780){myShadeC}
```

```
\tikz{\fill[shading=myShadeA] (0,0) rectangle (10,.5);}
\\ [3pt]\tikz{\fill[shading=myShadeB] (0,0) rectangle (10,.5);}
\\ [3pt]\tikz{\fill[shading=myShadeC] (0,0) rectangle (10,.5);}
\\ [3pt]\tikz{\fill[shading=myShadeA,shading angle=180] (0,0) rectangle (10,.5);}
```

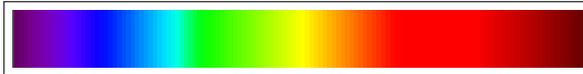


► Utilization of `\pgfspectraplotshade[options]{name}`

This command, without any options, builds and makes available a shading in the wavelength range from 380 nm to 780 nm to use in PGFPLOTS with the provided 'name'.

```
\pgfspectraplotshade{myPlotShadeA}
```

```
\fbox{\tikz{\fill[shading=myPlotShadeA] (0,0) rectangle (7.5,.75);}}
```



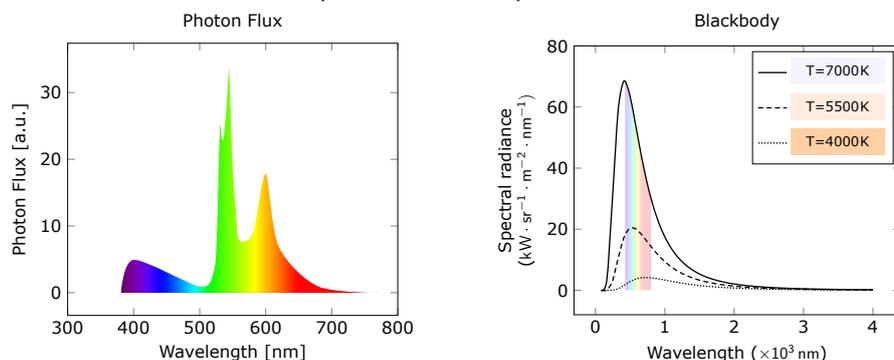
The optional argument can receive specific options for the shade or `\pgfspectra` options:

```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=white,UVcolor=white,
shade opacity=.2,gamma=.6]{myPlotShadeB}
```

```
\fbox{\tikz{\fill[shading=myPlotShadeB] (0,0) rectangle (7.5,.75);}}
```



The specific options available are `shade end`, `shade opacity` and `shade opacity color`. See section [The options for \pgfspectraplotshade](#) for detailed information on using these options. When used in PGFPLOTS it's possible to do plots like:



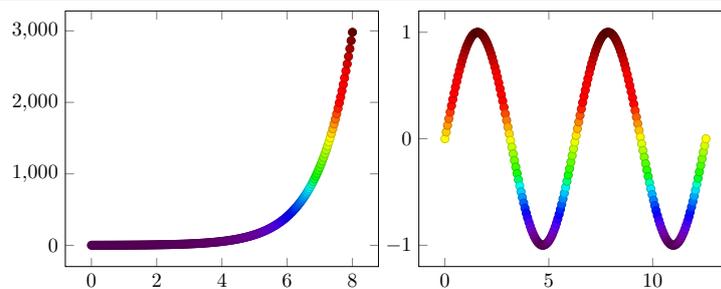
For these plots and other ones see [Using \pgfspectraplotshade and \pgfspectraplotmap with PGFPLOTS](#).

► Utilization of `\pgfspectraplotmap[<l|h>]{name}`

This command builds and makes available a **low** or **high resolution** color map in the wavelength range from 380 nm to 780 nm to use in PGFPLOTS with the provided 'name':

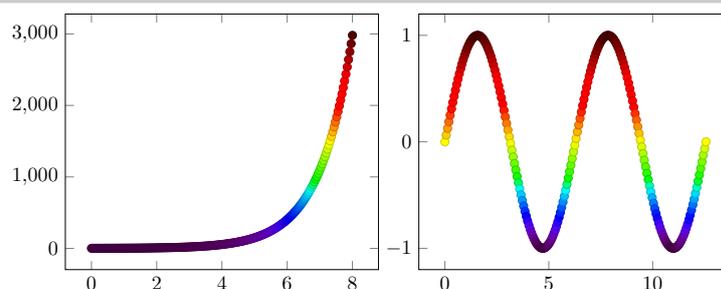
```
\pgfspectraplotmap{myColorMap}% low resolution (default value for optional parameter)
```

```
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMap]
\addplot+[scatter,only marks,domain=0:8,samples=200] {exp(x)};
\end{axis}
\end{tikzpicture}
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMap]
\addplot+[scatter,only marks,domain=0:4*pi,samples=200] {sin(deg(x))};
\end{axis}
\end{tikzpicture}
```



```
\pgfspectraplotmap[h]{myColorMapH}% high resolution ('h' value in optional parameter)
```

```
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMapH]
\addplot+[scatter,only marks,domain=0:8,samples=200] {exp(x)};
\end{axis}
\end{tikzpicture}
\begin{tikzpicture}
\begin{axis}[colormap name=myColorMapH]
\addplot+[scatter,only marks,domain=0:4*pi,samples=200] {sin(deg(x))};
\end{axis}
\end{tikzpicture}
```



Actually using high or low resolution produces the same effect on plot. The difference resides on the number of colors available to the 'color of colormap' feature. For more information see [Using \pgfspectraplotshade](#) and [\pgfspectraplotmap with PGFPLOTS](#).

The above commands – `\pgfspectrashade`, `\pgfspectraplotshade` and `\pgfspectraplotmap` – were inspired in the [TeX - LaTeX Stack Exchange](#) questions, [Filling optical spectrum curve with color gradient](#) and [How to create an electromagnetic spectrum using pgfplots package \(together with colormaps\)](#), which were referred by Stefan Pinnow, as examples, in a features request for the `pgf-spectra` package.

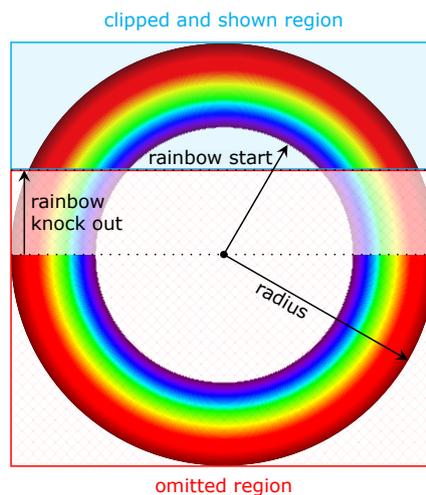
► **Utilization of `\pgfspectrarainbow`**`<[tikz options]><(rainbow options)>{radius}`

Without options this command draws a rainbow with the specified radius:

```
\pgfspectrarainbow{2cm}
```



The rainbow is designed with the following schema resulting in a clipped and shown region:



The options available could be specific options for the rainbow or *common TikZ* options:

- the rainbow specific options:
 - rainbow start
 - rainbow knock out
 - rainbow fade
 - rainbow transparency
 - rainbow background
- the *TikZ* options: any option known by *TikZ* and/or *TikZ* libraries.

For detailed information on using this command see [The options for `\pgfspectrarainbow`](#).

The options for `\pgfspectra`

For the commands `\pgfspectra` and `\pgfspectraStyle` there are a set of options available to draw the spectrum as described below.

The list of options is of the form `'key'` or `'key=value'` separated by commas.

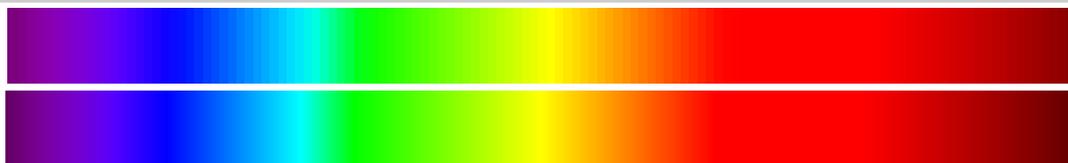
use visible shading

default: *true*

The visible region of the spectra is drawn using a *TikZ* shading instead of line by line, resulting in a faster drawing of that region. When set to `'false'` the visible region is drawn line by line: this value could be useful for printers that tend to be problematic when printing the shadings.

(new in v2.1.0)

```
\pgfspectra
\\ \pgfspectra[use visible shading=false]
```



width

default: *0.9\textwidth*

Sets the width of the spectrum.

```
\pgfspectra[width=10cm]
```



height

default: *1cm*

Sets the height of the spectrum.

```
\pgfspectra[height=40pt]
```



element

default: *NONE*

A single chemical symbol of an element or a list of chemical symbols.

```
\pgfspectra[element=H]
```



```
\pgfspectra[element={H,He}]
```



charge

default: 0

The charge of the *particle* to draw the spectrum. Use 'all' to get all available lines for the element, i.e, for the atom and all the positive ions that exist in the database.

```
\pgfspectra[element=He]
```



```
\pgfspectra[element=He,charge=1]
```



```
\pgfspectra[element=He,charge=2]
```

Element "He" with charge "2" have no lines to display.

```
\pgfspectra[element=He,charge=all]
```

**Imin**

default: 0

The minimum intensity of the lines to put in the spectrum. Value from 0 to 1.

```
\pgfspectra[element=He,Imin=.5]
```



```
\pgfspectra[element=He,Imin=.05]
```

**relative intensity**

default: false

Draws the lines respecting the intensity of the observed spectrum.

```
\pgfspectra[element=He,relative intensity]
```

**relative intensity threshold**

default: 0.25

Sets the minimum intensity for the lines in the spectrum when using relative intensities. When set to 0.25 a line with real intensity 0 will have a spectral intensity of 0.25 and a line with intensity equal to the max intensity observed in that spectrum will have an intensity in the computed spectrum of 1, assuming of course, an overall intensity in the range between 0 and 1.

```
\pgfspectra[element=He,relative intensity,relative intensity threshold=0]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=.25]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=.5]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=.75]
```



```
\pgfspectra[element=He,relative intensity,relative intensity threshold=1]
```



In fact, setting the relative intensity threshold to 1 is equivalent to the spectrum without relative intensities:

```
\pgfspectra[element=He]
```



line intensity

default: 100

Draws all the lines with the specified intensity between 0 and 100 (as a percentage of the maximum intensity).

```
\pgfspectra[element=He,line intensity=0]
```



```
\pgfspectra[element=He,line intensity=50]
```



```
\pgfspectra[element=He,line intensity=100]
```



```
\pgfspectra[element=He]
```



gamma

default: 0.8

Gamma color correction: any positive value.

`\pgfspectra[gamma=.1]``\pgfspectra[gamma=.8]``\pgfspectra[gamma=1]``\pgfspectra[gamma=2]``\pgfspectra[gamma=5]``\pgfspectra[gamma=10]`**brightness**

default: 1

Brightness color correction as in the CMYK color model. Value between 0 and 1. Zero stands for black and one for the maximum bright. *This option only works for the continuous component of the spectra, to change the "brightness" of spectral lines use the option 'line intensity'.*

`\pgfspectra[brightness=.1]``\pgfspectra[brightness=.5]``\pgfspectra[brightness=1]`

backdefault: *black*

Sets the background color of the spectrum. Only useful when there are spectral lines. Some shorthand are defined to put the visible region in the background: `'visible5'`, `'visible10'`, `'visible15'`, ... , `'visible100'`.

Note: this labels combined with the `'brightness'` option makes it possible to achieve other values on the background, since the visible amount (5%,10%,...) is multiplied by the value of brightness.

```
\pgfspectra[element=He,back=white]
```



```
\pgfspectra[element=He,back=black!50]
```



```
\pgfspectra[element=He,back=visible50]
```



```
\pgfspectra[element=He,back=visible50,brightness=.26]
```

**backIRUV**default: *black*

Sets the background color, *only for the emission spectrum*, outside the visible region

($10nm \leq \lambda < 380nm$ and $780nm < \lambda \leq 4000nm$)

(new in v2.0.0)

```
\pgfspectra[element=He,back=visible50,begin=100,end=1000,backIRUV=white]
```

**IRcolor**default: *rgb(0.3157,0.2373,0.2373)*

Sets the color for emission lines and for background in absorption spectrum in the IR region

($780nm < \lambda \leq 4000nm$)

(new in v2.0.0)

```
\pgfspectra[element=He,back=visible50,begin=100,end=1000,IRcolor=white]
```



```
\pgfspectra[element=He,begin=100,end=1000,IRcolor=white,absorption]
```



UVcolordefault: `rgb(0.3,0.2568,0.3)`Sets the color for emission lines and for background in absorption spectrum in the UV region ($10nm \leq \lambda < 380nm$)*(new in v2.0.0)*`\pgfspectra[element=He,back=visible50,begin=100,end=1000,UVcolor=white]``\pgfspectra[element=He,begin=100,end=1000,UVcolor=white,absorption]`**lines**default: `{}`A comma separated list of wavelengths in the interval $[10; 4000]$ nm.*(Interval updated in v2.0.0)*`\pgfspectra[lines={400,500,550,700}]``\pgfspectra[lines={200,205,400,500,550,700,850,950,980},begin=100,end=1000]`**line width**default: `1pt`

The width of each individual line in the spectrum.

`\pgfspectra[line width=2pt]``\pgfspectra[line width=2pt,element=He]`**begin**default: `380`The starting wavelength in nanometres of the spectrum ($10 \leq \lambda \leq 4000$).*(Interval updated in v2.0.0)*`\pgfspectra[begin=500]`

end default: 740

The finishing wavelength in nanometres of the spectrum ($10 \leq \lambda \leq 4000$).

(Interval updated in v2.0.0)

```
\pgfspectra[end=500]
```



Remark: *it's obviously possible to set 'begin' and 'end' at the same time and if desired change the order of the wavelengths.*

```
\pgfspectra[begin=500,end=700]
```



```
\pgfspectra[begin=700,end=500]
```



```
\pgfspectra[begin=780,end=380]
```



```
\pgfspectra[begin=780,end=380,element=He]
```



absorption default: *false*

Draws the absorption spectrum instead of the emission one.

```
\pgfspectra[element=H,absorption]
```



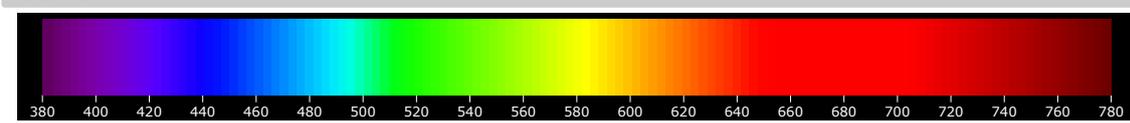
```
\pgfspectra[element={H,He},absorption]
```



axis default: *false*

By default draws a nanometric axis below the spectrum.
In v2.1.0 is now possible to *change the unit* of the axis.

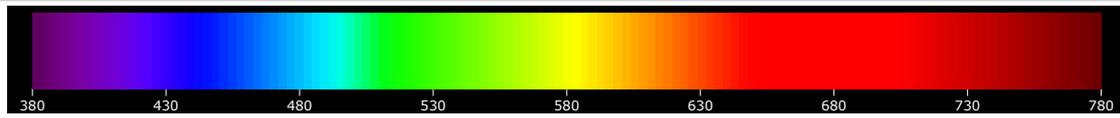
```
\pgfspectra[axis]
```



axis step

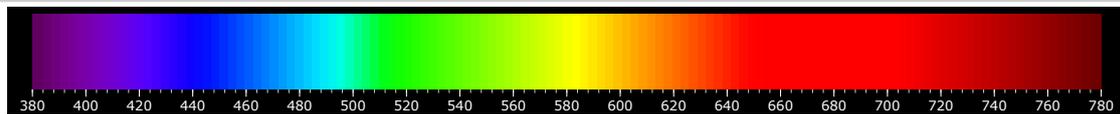
default: 20

The increment (in nanometres) to use in the axis scale.

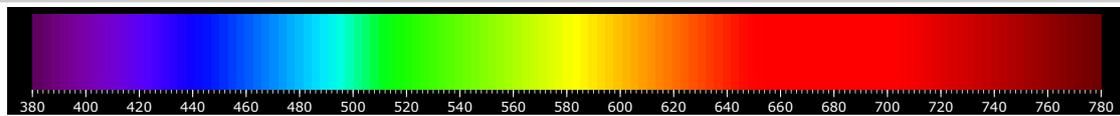
`\pgfspectra[axis,axis step=50]`**axis ticks**

default: 0

The number of minor ticks between two consecutive ticks in the axis.

(new in v2.0.0)`\pgfspectra[axis,axis ticks=4]`

Keep in mind, if you desire to divide two consecutive ticks into 10 equal parts set `'axis ticks=9'`:

`\pgfspectra[axis,axis ticks=9]`**axis unit**default: *nm*

Sets the unit to use in the displayed values of wavelenghts in the axis below the spectrum.

Available units are:

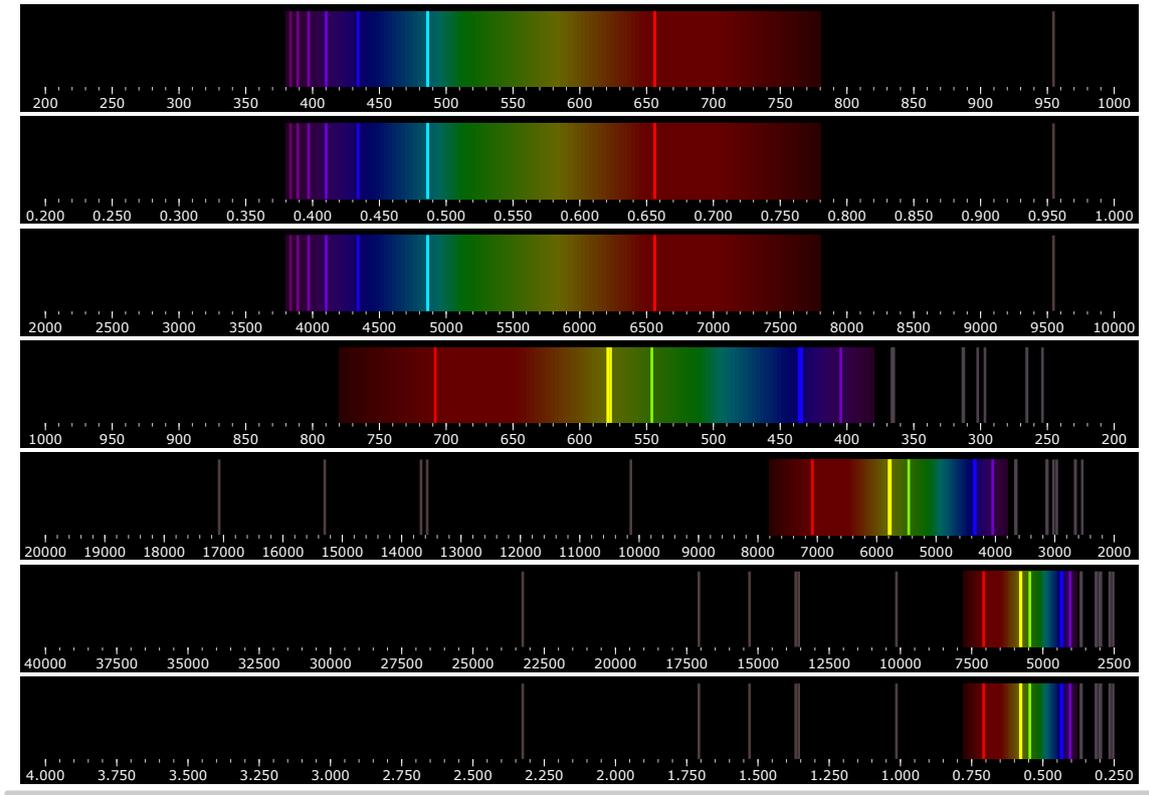
- nanometre (nm): `axis unit=nm`
- micrometre (μm): `axis unit=micron`
- angstrom (\AA): `axis unit=A`

(new in v2.1.0)

```

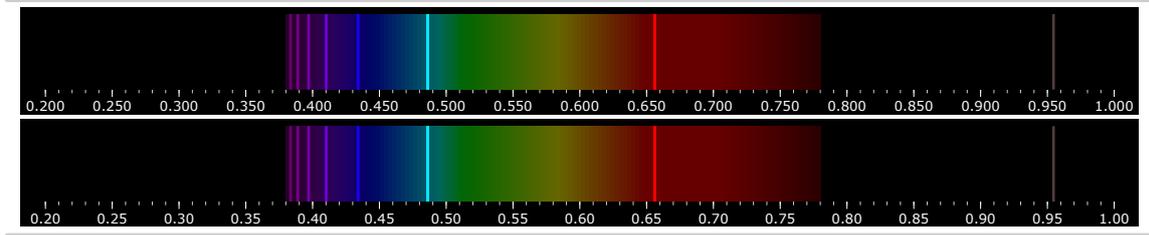
\pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=micron,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=A,back=visible40]
\\ \pgfspectra[element=Hg,begin=1000,end=200,axis,axis step=50,axis
ticks=4,back=visible40]
\\ \pgfspectra[element=Hg,begin=2000,end=200,axis,axis step=100,axis
ticks=4,axis unit=A,back=visible40]
\\ \pgfspectra[element=Hg,begin=4000,end=250,axis,axis step=250,axis
ticks=4,axis unit=A,back=visible40]
\\ \pgfspectra[element=Hg,begin=4000,end=250,axis,axis step=250,axis
ticks=4,axis unit=micron,back=visible40]

```



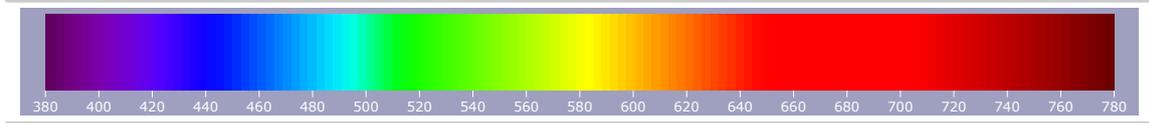
axis unit precision default: 3
 Sets the precision of the displayed wavelenghts in the axis below the spectrum. *(new in v2.1.0)*

```
\pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis ticks=4,axis
unit=micron,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=micron,axis unit precision=2,back=visible40]
```



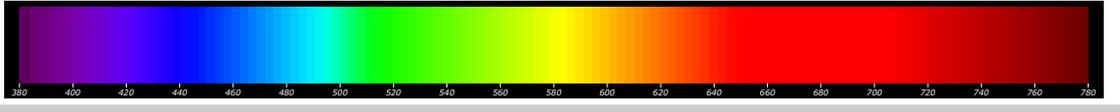
axis color default: black
 The color of the axis.

```
\pgfspectra[axis,axis color=red!50!green!50!blue!50]
```



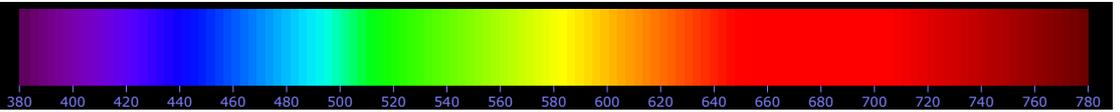
axis font default: `\tiny`
 The font specs to use in the axis.

```
\pgfspectra[axis,axis font=\fontsize{3}{3}\itshape\selectfont]
```



axis font color default: `white`
 The color of the font used in the axis.

```
\pgfspectra[axis,axis font color=blue!50!white]
```



label default: `false`
 Puts a label for the spectrum.

```
\pgfspectra[label]
```



```
\pgfspectra[label,element=He]
```



label position default: `west`
 Sets the position of the label according to:

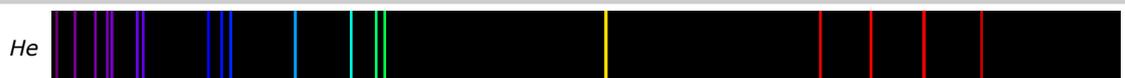
north west	north	north east
west	<i>spectrum</i>	east
south west	south	south east

```
\pgfspectra[label,label position=east,element=He]
```



label font default: `\bfseries\small`
 The font specs for the label.

```
\pgfspectra[label,label font=\footnotesize\itshape,element=He]
```



label font colordefault: *black*

The color of the font used in the label.

`\pgfspectra[label,label font color=blue!50!white,element=He]`

He

**label before text**default: `{}`

Inserts text before the value stored in the label: if chemical symbols were provided, the label has them stored, otherwise it is empty.

`\pgfspectra[label,label before text=text\ ,element=He]`

text He



Remark: The `_` is to insert a space between the text entered by user and the text stored in label.

label after textdefault: `{}`

Inserts text after the value stored in the label: if chemical symbols were provided, the label has them stored, otherwise it is empty.

`\pgfspectra[label,label after text=\ text,element=He]`

He text

**redshift**default: `{}`

Redshift (or blueshift) the spectral lines:

The redshift value (z) is *defined* as $1 + z = \lambda_{obs} / \lambda_E$ which leaves the observed wavelength to $\lambda_{obs} = (1 + z)\lambda_E$, given the emitted wavelength of the source (λ_E).

- Use `'redshift=<numeric value>'` to directly enter the redshift value
- or use `'redshift={D=<numeric value 1>/<numeric value 2>}'` to compute the Relativistic Doppler redshift with $\bar{v} = \text{<numeric value 1>}$ and $\theta = \text{<numeric value 2>}^\circ$.

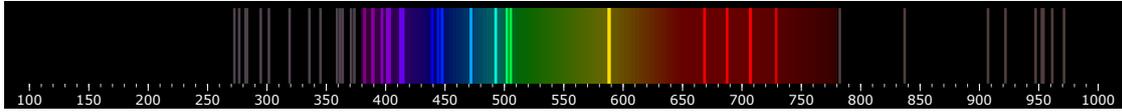
The Relativistic Doppler redshift ($1 + z$) is calculated accordingly:

$$1 + z = \frac{1 + \bar{v} \cos \theta}{\sqrt{1 - \bar{v}^2}} \quad \bar{v} = \frac{v}{c}$$

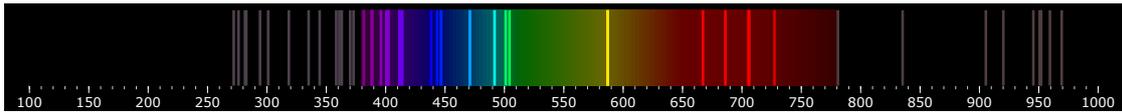
where \bar{v} is the *normalized velocity* (in units of the speed of light in vacuum, c) of the emitter and θ is the angle between the direction of relative motion and the direction of emission in the observer's frame (zero angle is directly away from the observer). So, if the source of light is moving away from an observer, then redshift occurs ($z > 0$), but, if the source moves towards the observer, then blueshift occurs ($z < 0$).

*(new in v2.0.0)*

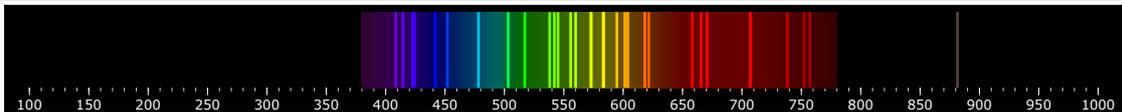
```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift={D=.001/0}]
```



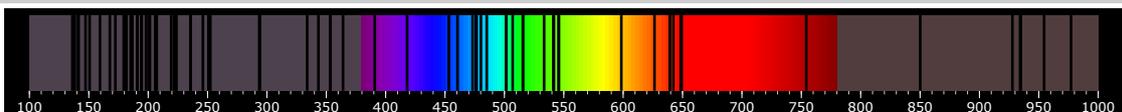
```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift={D=.001/180}]
```



```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift=.5]
```

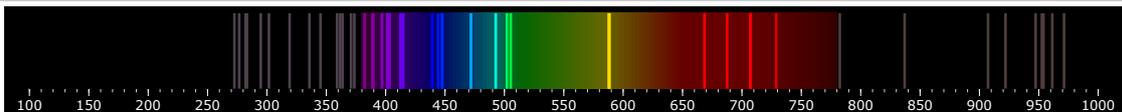


```
\pgfspectra[element=He,absorption,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift=-.5]
```



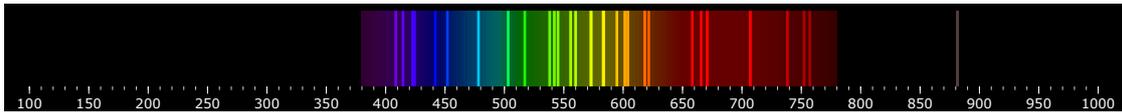
show redshift value default: *false*
 Writes the value of the redshift (left below the spectrum). *(new in v2.0.0)*

```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift={D=.001/0},show redshift value]
```



Relativistic Doppler redshift $z=0.001$ ($v=.001c$; $\theta=0^\circ$)

```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis ticks=4,begin=100,end=1000,redshift=.5,show redshift value]
```



redshift $z=.5$

The options for `\pgfspectraplotshade`

This command creates a shade to use with the `\addplot` command provided by the PGF-PLOTS package. The shade starts at `shade begin` and finishes at `shade end`. The shading could be adjusted using the following options:

- `shade begin`
- `shade end`
- `shade opacity`
- `shade opacity color`
- `logarithmic`

`shade begin`

default: 380

This value determines the start wavelength of the computed shading. It should be set equal to the minimum value of the plotted data. It could be different from the inferior limit of the domain provided to the plot (see the [PGFPLLOTS package documentation](#) for more information). The range of accepted values goes from 0 nm to `(shade end-1)`. *(new in v2.1.1)*

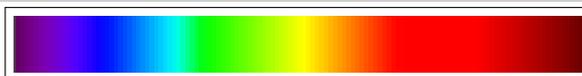
`shade end`

default: 780

This value determines the end wavelength of the computed shading. It should be set equal to the maximum value of the plotted data and could be different from the superior limit of the domain provided to the plot. The range of accepted values goes from `(shade begin+1)` to 16000 nm. *(new in v2.1.0)*

```
\pgfspectraplotshade{shadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade begin=600]{shadeBegin600}
```

```
\fbox{\tikz{\fill[shading=shadeEnd600] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade end=600]{shadeEnd600}
```

```
\fbox{\tikz{\fill[shading=shadeEnd1500] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[begin=300,shade end=600]{shade300to600}
```

```
\fbox{\tikz{\fill[shading=shade300to600] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[begin=600,shade end=900]{shade600to900}
```

```
\fbox{\tikz{\fill[shading=shade600to900] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[begin=300,shade end=900]{shade300to900}
```

```
\fbox{\tikz{\fill[shading=shade300to900] (0,0) rectangle (7.5,.75);}}
```



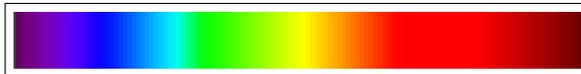
shade opacity

default: 1

The opacity of the computed shade. '0' stands for 0% and the shading is totally transparent; '1' stands for 100% and the shading isn't transparent at all. *(new in v2.1.0)*

```
\pgfspectraplotshade{shadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity=.5]{shadeOpacity50}
```

```
\fbox{\tikz{\fill[shading=shadeOpacity50] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity=0]{shadeOpacity0}
```

```
\fbox{\tikz{\fill[shading=shadeOpacity0] (0,0) rectangle (7.5,.75);}}
```



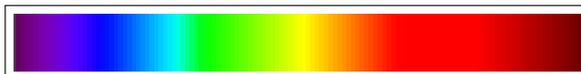
shade opacity color

default: white

The background color of the computed shading. Only visible when `shade opacity` is lesser then 1. *(new in v2.1.0)*

```
\pgfspectraplotshade{shadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity color=black]{shadeOpacityBlack}
```

```
\fbox{\tikz{\fill[shading=shadeOpacityBlack] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity color=black, shade opacity=.5]{shadeOpacityBlack50}
```

```
\fbox{\tikz{\fill[shading=shadeOpacityBlack50] (0,0) rectangle (7.5,.75);}}
```

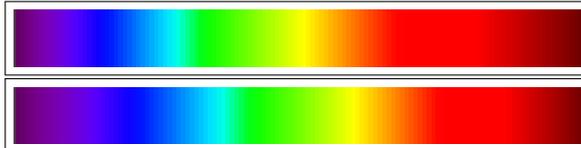


logarithmicdefault: *false*

When set to true the shading is build in a logarithmic scale. The smaller wavelengths are *wided* and the longer ones are *shortened* in the displayed region. *(new in v2.1.1)*

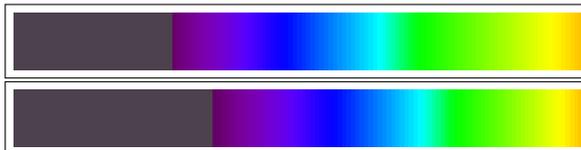
```
\pgfspectraplotshade[logarithmic]{logshadeDefault}
```

```
\fbox{\tikz{\fill[shading=shadeDefault] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshadeDefault] (0,0) rectangle (7.5,.75);}}
```



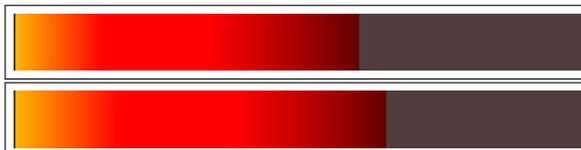
```
\pgfspectraplotshade[logarithmic,begin=300,shade end=600]{logshade300to600}
```

```
\fbox{\tikz{\fill[shading=shade300to600] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade300to600] (0,0) rectangle (7.5,.75);}}
```



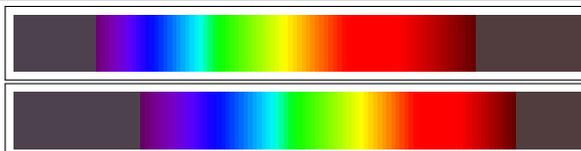
```
\pgfspectraplotshade[logarithmic,begin=600,shade end=900]{logshade600to900}
```

```
\fbox{\tikz{\fill[shading=shade600to900] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade600to900] (0,0) rectangle (7.5,.75);}}
```



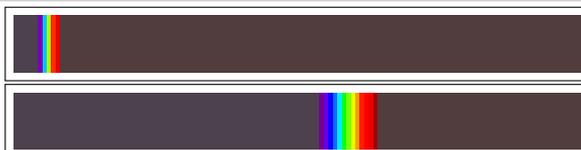
```
\pgfspectraplotshade[logarithmic,begin=300,shade end=900]{logshade300to900}
```

```
\fbox{\tikz{\fill[shading=shade300to900] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade300to900] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[logarithmic,begin=10,shade
end=10000]{logshade10to10000}
```

```
\fbox{\tikz{\fill[shading=shade10to10000] (0,0) rectangle (7.5,.75);}}
\\ \fbox{\tikz{\fill[shading=logshade300to900] (0,0) rectangle (7.5,.75);}}
```



The options for `\pgfspectrarainbow`

For the command `\pgfspectrarainbow` there are a set of options that control the rainbow drawn.

The specific rainbow options are:

- `rainbow start`
- `rainbow knock out`
- `rainbow fade`
- `rainbow transparency`
- `rainbow background`

Some `TikZ` keys that affect the rainbow are:

- `'color'`
- `opacity`
- `scope fading`

The default rainbow drawn is:

```
\pgfspectrarainbow{2cm}
```



rainbow start

default: `.6`

The fraction from which the rainbow colors begin, relative to the center of a circle with radius 1. This value should be in the interval $[0,1]$. *(new in v2.1.0)*

```
\pgfspectrarainbow(rainbow start=.8){2cm}% the rainbow colors starts at 1.6cm
\hspace{1cm}%
\pgfspectrarainbow(rainbow start=.4){2cm}% the rainbow colors starts at .8cm
```

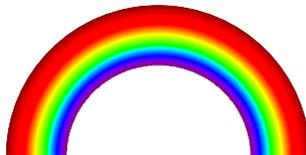


rainbow knock out

default: `.4`

The relative distance from the half-circle base to perform the clip. This value should be in the interval $[-1,1]$. *(new in v2.1.0)*

```
\pgfspectrarainbow(rainbow knock out=0){2cm}% the full half circle
```



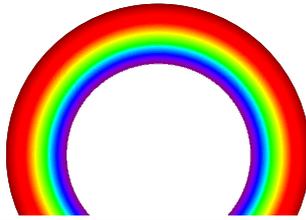
```
\pgfspectrarainbow(rainbow knock out=.4){2cm}% the default value
```



```
\pgfspectrarainbow(rainbow knock out=.8){2cm}% only 80% of the half circle is shown
```



```
\pgfspectrarainbow(rainbow knock out=-.4){2cm}% «extending» the half-circle
```

**rainbow fade**default: `{}`

Applies a scope fading in the clipped region (requires loading the TikZ fadings library). For more information about the fadings see the TikZ manual. *(new in v2.1.0)*

```
%\usetikzlibrary{fadings}
...
\pgfspectrarainbow(rainbow fade=south){2cm}
\hspace{1cm}%
\pgfspectrarainbow(rainbow fade=west){2cm}
```

rainbow transparencydefault: `0`

The overall transparency of the rainbow. `'0'` (0%) stands for the fill colors in the rainbow without transparency ; `'1'` (100%) represents a totally transparent rainbow. *(new in v2.1.0)*

```
\pgfspectrarainbow(rainbow transparency=.5){2cm}
```

**rainbow background**default: `white`

The background color below the rainbow (only visible with transparency). *(new in v2.1.0)*

```
\pgfspectrarainbow(rainbow background=blue){2cm}
\hspace{1cm}%
\pgfspectrarainbow(rainbow background=blue,rainbow transparency=.5){2cm}
```



Some of the TikZ keys that affect the rainbow:

```
\pgfspectrarainbow[blue]{2cm}% Setting only the fill color only takes no effect
```



```
\pgfspectrarainbow[blue,scope fading=south]{2cm}
```



```
\pgfspectrarainbow[blue,opacity=.5]{2cm}
```



More examples in [Using `\pgfspectrarainbow`](#) `<[tikz options]><(rainbow options)>{radius}`.

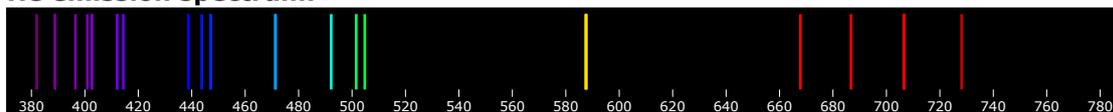
Examples

► Using `\pgfspectra`

Here are some examples for drawing some *eventually useful* spectra:

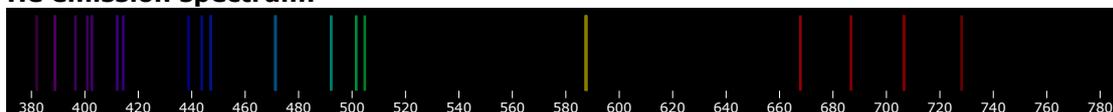
```
\pgfspectra[element=He,axis,label,label position=north west,
label after text=\ emission spectrum:]
```

He emission spectrum:



```
\pgfspectra[element=He,axis,label,label position=north west,label after text=
\ emission spectrum:,relative intensity,relative intensity threshold=.5]
```

He emission spectrum:

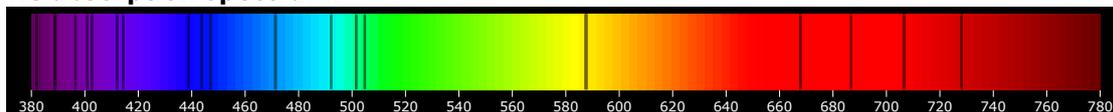


```
\pgfspectra[element=He,charge=all,line intensity=50,Imin=.05]
```

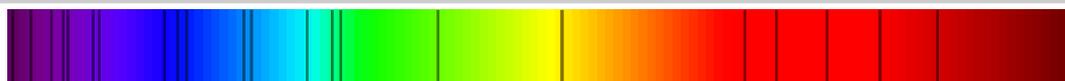


```
\pgfspectra[element=He,absorption,axis,label,label position=north west,label after
text=\ absorption spectrum:,relative intensity,relative intensity threshold=.5]
```

He absorption spectrum:



```
\pgfspectra[element=He,charge=all,absorption,line intensity=50]
```



```
\pgfspectra[element=He,charge=all,relative intensity,back=visible75,gamma=2]
```



When the lines are manually inserted it's possible to use 'label before text' only with personalized text. In the next three examples 'label before text' is used to make labels for a multiple choice problem, omitting evidently the type of luminous font.

✓ Laser He-Ne

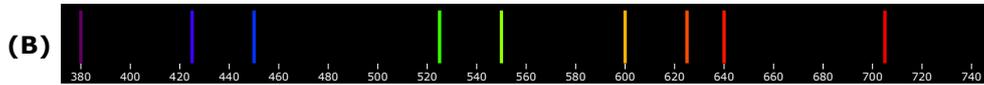
```
\pgfspectra[height=.7cm,end=740,lines={633},line
width=1.25pt,width=.75\linewidth,label,axis,label before text=(A),
axis font=\fontsize{4pt}{6pt}\selectfont]
```

(A)



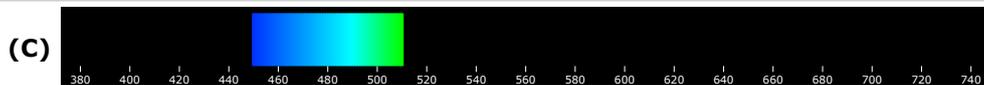
✓ Fluorescent lamp

```
\pgfspectra[height=.7cm,end=740,lines={380,425,450,525,550,600,625,640,705},
line width=1.25pt,width=.75\linewidth,label,axis,label before text=(B),
axis font=\fontsize{4pt}{6pt}\selectfont]
```



✓ Blue LED

```
\pgfspectra[height=.7cm,end=740,lines={450,451,452,453,454,455,456,457,458,
459,460,461,462,463,464,465,466,467,468,469,470,471,472,473,474,475,476,
477,478,479,480,481,482,483,484,485,486,487,488,489,490,491,492,493,494,
495,496,497,498,499,500,501,502,503,504,505,506,507,508,509,510},
line width=1.25pt,width=.75\linewidth,label,axis,label before text=(C),
axis font=\fontsize{4pt}{6pt}\selectfont]
```



✓ Sun like spectrum

```
\pgfspectra[element={H,Fe,Mg,Na},absorption,line intensity=40,Imin=.05]
```



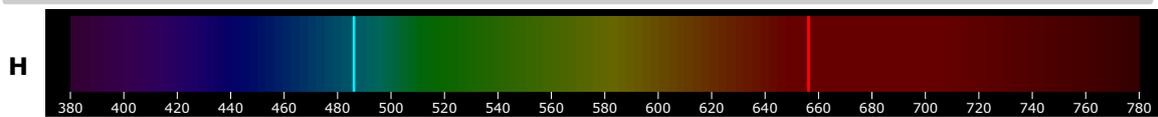
✓ Sirius like spectrum

```
\pgfspectra[element={H,He},absorption,line intensity=40,Imin=.05]
```

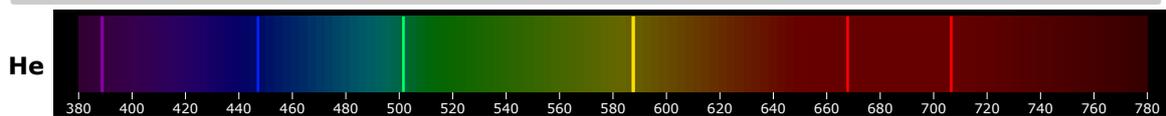


✓ "Classical" emission spectra of elements:

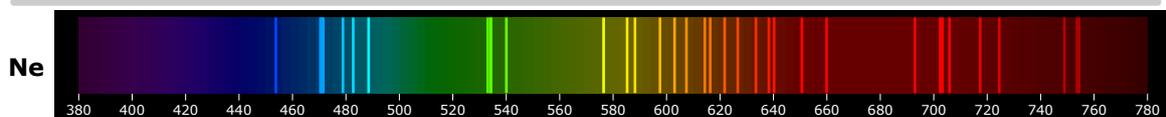
```
\pgfspectra[element=H,back=visible40,gamma=.6,label,axis,Imin=.05]
```



```
\pgfspectra[element=He,back=visible40,gamma=.6,label,axis,Imin=.05]
```

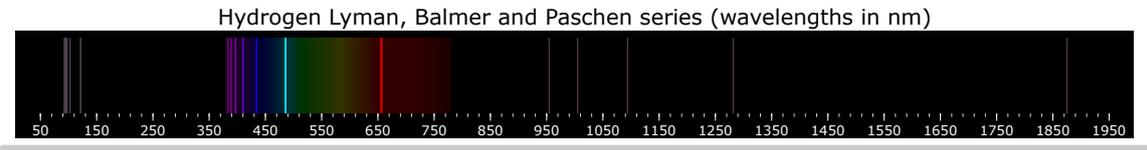


```
\pgfspectra[element=Ne,back=visible40,gamma=.6,label,axis,Imin=.05]
```



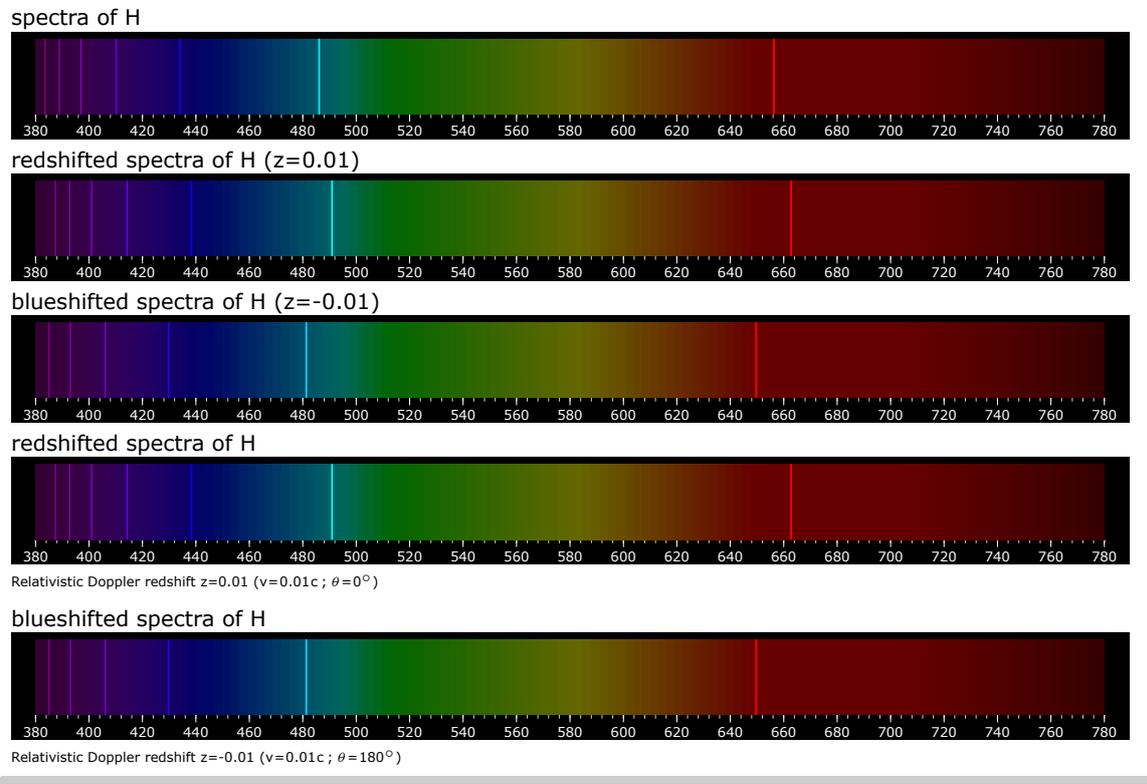
✓ Series of hydrogen:

```
\pgfspectra[element=H,line width=.5pt,begin=50,end=1950,axis,axis
step=100,axis ticks=4,back=visible40,gamma=.6,brightness=.5,label,label
position=north,label font=\footnotesize,label after text={ydrogen Lyman, Balmer
and Paschen series (wavelengths in nm)}]
```



✓ Redshifted & Blueshifted lines of hydrogen using the \foreach statement:

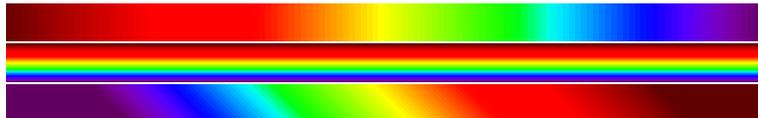
```
\pgfspectraStyle[axis,axis ticks=4,back=visible40,gamma=.6,line width=.5pt]
\pgfspectra[element=H,label,label position=north west,label
font=\footnotesize,label before text={spectra of \ }]
\foreach \SQ/\z/\shift in {H/0.01/redshifted,H/-0.01/blueshifted}{
  \pgfspectra[element=\SQ,label,label position=north west,label
font=\footnotesize,label before text={\shift\ spectra of \ },label after
text={ \ (z=\z)},redshift=\z]
}
\foreach \SQ/\z/\shift in {H/{D=0.01/0}/redshifted,H/{D=0.01/180}/blueshifted}{
  \pgfspectra[element=\SQ,label,label position=north west,label
font=\footnotesize,label before text={\shift\ spectra of \ },redshift=\z,show
redshift value]
}
```



► Using `\pgfspectrashade` in TikZ

Obviously, the *normal* TikZ keys used to control the shadings apply to the shading generated via `\pgfspectrashade`:

```
\pgfspectrashade(380,780){myShadeA}
\\ \tikz{\fill[shading=myShadeA,shading angle=180] (0,0) rectangle (10,.5);}
\\ \tikz{\fill[shading=myShadeA,shading angle=90] (0,0) rectangle (10,.5);}
\\ \tikz{\fill[shading=myShadeA,shading angle=45] (0,0) rectangle (10,.5);}
```



Providing an opacity to the drawing and applying a shade works well too:

```
\pgfspectrashade(380,780){myShadeA}
• on black background:
  \\ \tikz{\fill[shading=myShadeA,opacity=.5] (0,0) rectangle (10,.5);}
• on white background:
  \\ \tikz{\fill[white,shading=myShadeA,opacity=.5] (0,0) rectangle (10,.5);}
• on red background:
  \\ \tikz{\fill[red,shading=myShadeA,opacity=.5] (0,0) rectangle (10,.5);}
```

- on black background:



- on white background:



- on red background:



The gamma in the generated shade (via `\pgfspectrashade`) could be modified using the `'gamma'` key of `\pgfspectra` set by the command `\pgfspectraStyle`:

```
\pgfspectrashade(380,780){myShadeA}
\tikz{\fill[myShadeA] (0,0) rectangle (10,.5);}
```



```
\pgfspectraStyle[gamma=2]
\pgfspectrashade(380,780){myShadeGammaII}
\tikz{\fill[myShadeGammaII] (0,0) rectangle (10,.5);}
```



```
\pgfspectraStyle[gamma=10]
\pgfspectrashade(380,780){myShadeGammaX}
\tikz{\fill[myShadeGammaX] (0,0) rectangle (10,.5);}
\pgfspectraStyleReset
```



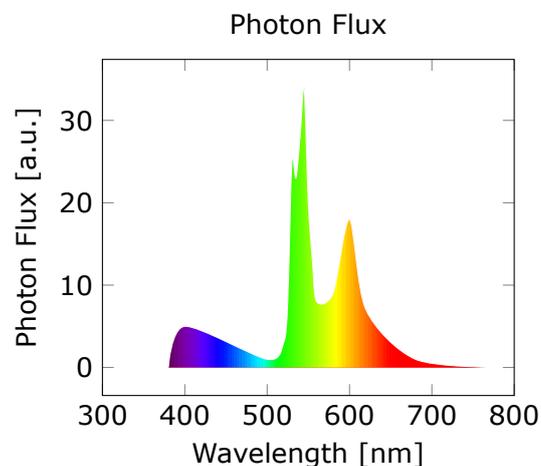
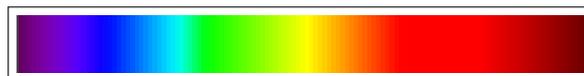
► Using `\pgfspectraplotshade` and `\pgfspectraplotmap` with PGFPLOTS

The command `\pgfspectraplotshade` is designed to build a shading to use with PGFPLOTS. Next examples show a few possibilities of how it could be used regarding two *sources*: a source of light and their photon flux and the blackbody spectral radiance.

In order to correctly make the filling between the path at axis and the plotted curve, the path should begin at 'shade begin' and end at 'shade end':
`\path[name path=axis] (shade begin,0) – (shade end,0);`

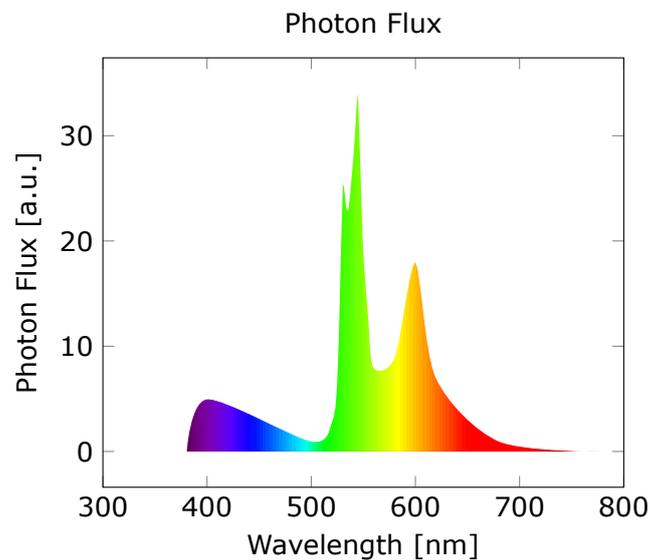
```
\pgfspectraplotshade{visiblespectrum}% default shading [380;780]nm
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrum] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=800,%
]%
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
};
\path[name path=axis] (380,0) -- (780,0);
\addplot+ [thick,shading=visiblespectrum]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```



The above example could be obtained with the following *improved* code, based on a suggestion made by Stefan Pinnow:

```
\begin{tikzpicture}
  \pgfmathsetmacro{\xmin}{300}
  \pgfmathsetmacro{\xmax}{800}
  \pgfmathsetmacro{\shbegin}{380}
  \pgfmathsetmacro{\shend}{780}
  \pgfspectraplotshade[shade begin=\shbegin,shade end=\shend]{visiblespectrum}
  \begin{axis}[
    title=Photon Flux,
    xlabel={Wavelength in nm},
    ylabel={Photon Flux in a.u.},
    xmin=\xmin,
    xmax=\xmax,
  ]
  \addplot [smooth, name path=spectrum,white] coordinates {
    (380,0) (400,5) (500,1) (520,3) (525,8) (530,25)
    (535,23) (540,28) (545,34) (550,20) (555,13)
    (560,8) (580,9) (600,18) (620,7) (680,1) (780,0)
  };
  \path [name path=axis] (\shbegin,0) -- (\shend,0);
  \addplot+ [thick,shading=visiblespectrum]
    fill between [of=spectrum and axis];
  \end{axis}
\end{tikzpicture}
```

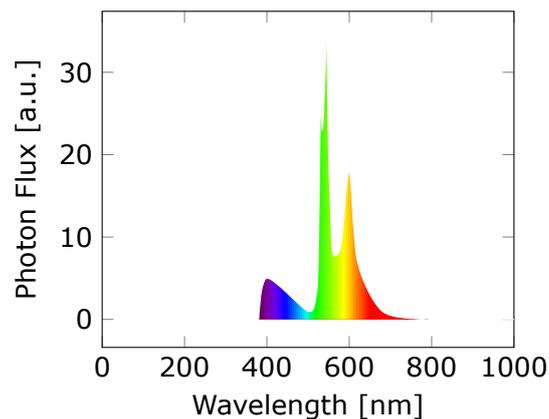


```
\pgfspectraplotshade[shade end=1000]{visiblespectrum}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrum] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=0,%
xmax=1000,%
]
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
( 800, 0 ) ( 900, 0 ) ( 1000, 0 )
};
\path[name path=axis] (380,0) -- (1000,0);
\addplot+ [thick,shading=visiblespectrum]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```

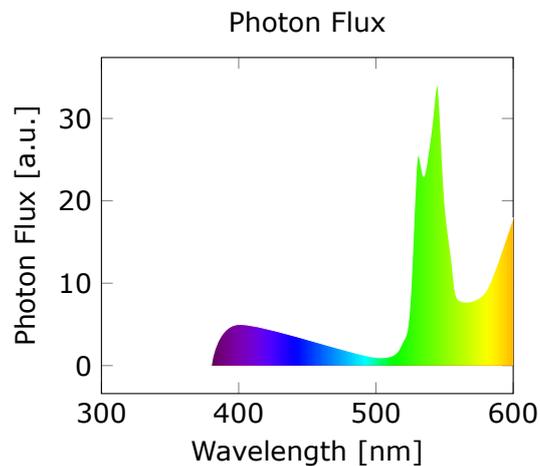
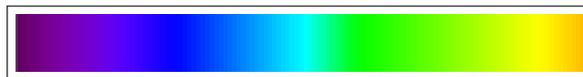


Photon Flux



```
\pgfspectraplotshade[shade end=600]{visiblespectrum}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrum] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=600,%
]
\addplot[smooth, name path=spectrum,draw=none] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 )
};
\path[name path=axis] (380,0) -- (600,0);
\addplot+ [thick,shading=visiblespectrum]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```

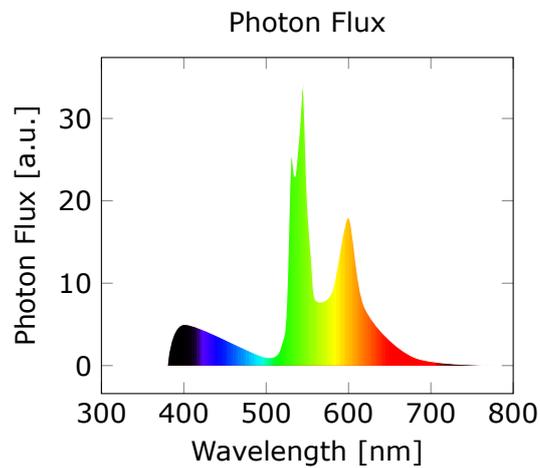
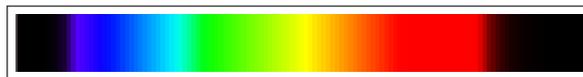


`\pgfspectraplotshade[gamma=10]{visiblespectrumGammaX}`

```

\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrumGammaX] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}{%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=800,%
]
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 380, 0 ) (400,5) ( 500, 1 ) ( 520, 3 ) ( 525, 8 ) ( 530, 25 )
( 535, 23 ) ( 540, 28 ) ( 545, 34 ) ( 550, 20 ) ( 555, 13 )
( 560, 8 ) ( 580, 9 ) ( 600, 18 ) ( 620, 7 ) ( 680, 1 ) ( 780, 0 )
};
\path[name path=axis] (380,0) -- (780,0);
\addplot+ [thick,shading=visiblespectrumGammaX]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%

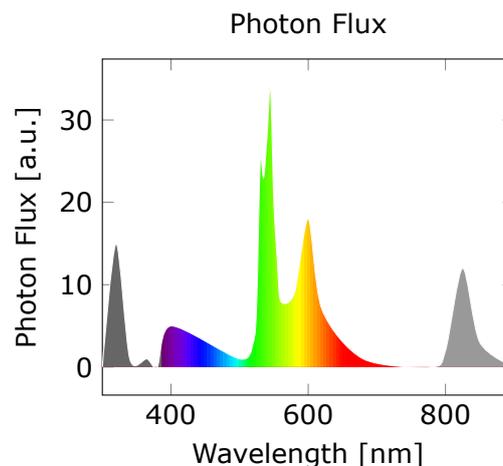
```



Note: when setting the color for IR or UV within `\pgfspectraplotshade` make sure it doesn't end with '`!<number>`' like '`black!40`'; use '`black!40!white`' instead.

```
\pgfspectraplotshade[IRcolor=black!40!white,UVcolor=black!60!white,shade begin=300,shade end=900]{visiblespectrumIRUV}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=visiblespectrumIRUV] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[%
title= Photon Flux,%
xlabel={Wavelength [nm]},%
ylabel={Photon Flux [a.u.]},%
xmin=300,%
xmax=900,%
]
\addplot[smooth, name path=spectrum,white] plot[] coordinates{%
( 300 , 0) (320, 15) (340, 1) (365, 1)
( 380, 2.5) (400,5) ( 500, 1) ( 520, 3) ( 525, 8) ( 530, 25)
( 535, 23) ( 540, 28) ( 545, 34) ( 550, 20) ( 555, 13)
( 560, 8) ( 580, 9) ( 600, 18) ( 620, 7) ( 680, 1) ( 780, 0)
( 800, 2) (825, 12) (850, 3) ( 900, 0)
};
\path[name path=axis] (300,0) -- (900,0);
\addplot+ [thick,shading=visiblespectrumIRUV]
fill between[of=spectrum and axis];
\end{axis}
\end{tikzpicture}%
```



For the *blackbody spectral radiance*, the Planck's distribution is used with the values:

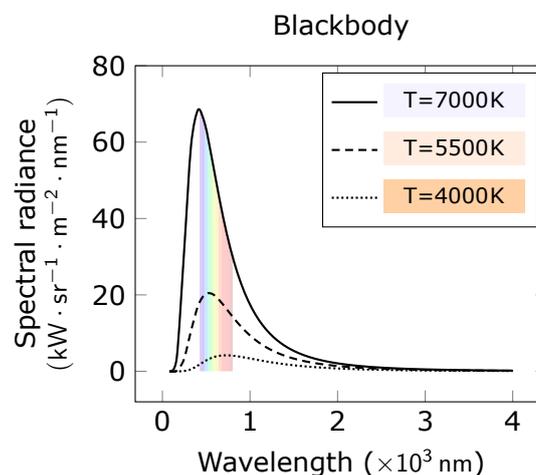
- $c = 3 \times 10^{14}$ microns \cdot s⁻¹ – speed of light
- $h = 6.626 \times 10^{22}$ kg \cdot microns² \cdot s⁻¹ – Planck constant
- $k_B = 1.38 \times 10^{-11}$ kg \cdot microns² \cdot s⁻² \cdot K⁻¹ – Boltzmann constant
- λ – wavelength (microns)
- T – temperature (K)
- Planck distribution: $B_\lambda = 2hc^2 \frac{1}{\lambda^5 e^{\frac{hc}{\lambda k_B T} - 1}}$ (kW \cdot sr⁻¹ \cdot m⁻¹ \cdot nm⁻¹)

The legend of the plots is created with the following definition:

```
\def\myentry#1{\tempercolor{#1}%
\tikz{\fill[tempercolor] (0,-.5pt) rectangle (40pt,.5pt)
node[midway,font=\footnotesize,anchor=mid] {\color{black} T=#1\hspace{.1ex}K};}}
```

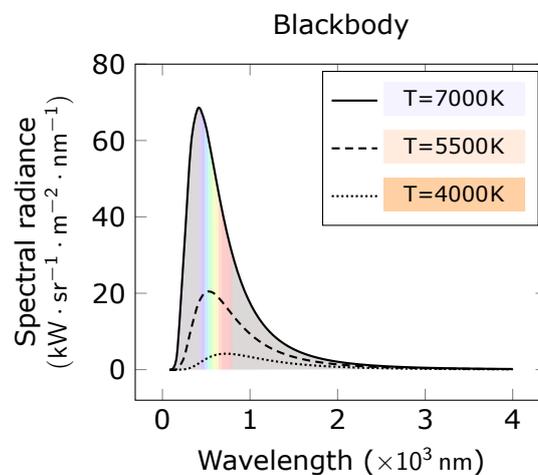
```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=white,UVcolor=white,
gamma=.6,shade opacity=.2]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[title=Blackbody,xlabel={Wavelength (\mathsf{\times 10^3}\,nm)}],%
ylabel={Spectral radiance\mathsf{(kW\cdot sr^{-1}\cdot m^{-2}\cdot nm^{-1})}},%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [white,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```



```
\pgfspectraplotshade[shade begin=0,shade end=4000,shade opacity=.2,
gamma=.6]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[title=Blackbody,xlabel={Wavelength ( $\mathsf{\times 10^3}$ ,nm)}],%
ylabel={Spectral radiance\
 $\mathsf{\text{kW}\cdot\text{sr}^{-1}\cdot\text{m}^{-2}\cdot\text{nm}^{-1}}$ },%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [white,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```

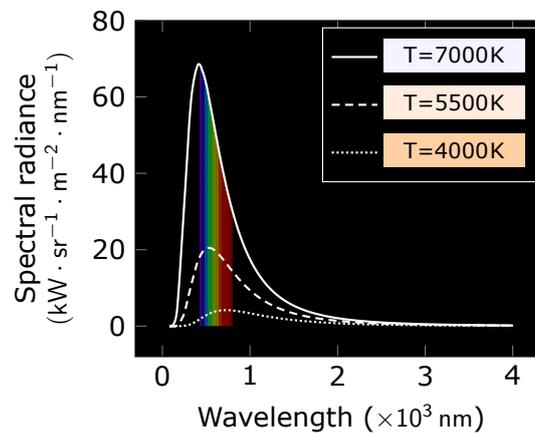


```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=black,UVcolor=black,
gamma=.6,shade opacity=.5,shade opacity color=black]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[
axis background/.style={fill=black},%
legend style={fill=black,draw=white},%
title=Blackbody,xlabel={Wavelength ( $\times 10^3$  nm)},%
ylabel={Spectral radiance\
 $\text{kW} \cdot \text{sr}^{-1} \cdot \text{m}^{-2} \cdot \text{nm}^{-1}$ },%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [black,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```



Blackbody

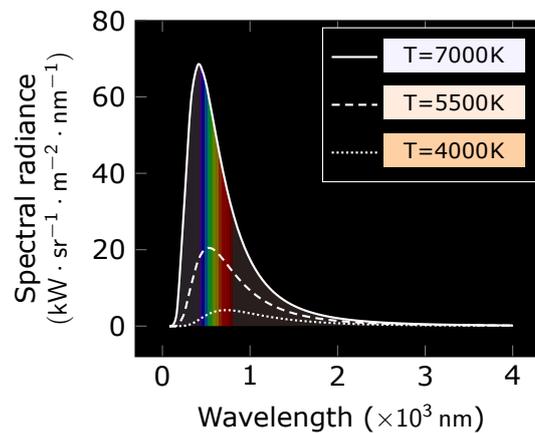


```
\pgfspectraplotshade[shade begin=0,shade end=4000,shade opacity=.5,
gamma=.6,shade opacity color=black]{BBody}
```

```
\makebox[\linewidth][c]{%
\fbbox{\tikz{\fill[shading=BBody] (0,0) rectangle (7.5,.75);}}%
}%
\\ [10pt]\\ \begin{tikzpicture}
\begin{axis}[
axis background/.style={fill=black},%
legend style={fill=black,draw=white},%
title=Blackbody,xlabel={Wavelength ( $\times 10^3$  nm)},%
ylabel={Spectral radiance\
 $\text{kW} \cdot \text{sr}^{-1} \cdot \text{m}^{-2} \cdot \text{nm}^{-1}$ },%
ylabel style={align=center},ymax=80,domain=0:4]%
\addplot[smooth, name path=spectrum,black,samples=50,thick] plot[]
{119.268/(x^5*(exp(14404/(x*7000))-1))};\addlegendentry{\myentry{7000}}%
\addplot[smooth,black,samples=50,densely dashed,thick] plot[]
{119.268/(x^5*(exp(14404/(x*5500))-1))};\addlegendentry{\myentry{5500}}%
\addplot[smooth,black,samples=50,densely dotted,thick] plot[]
{119.268/(x^5*(exp(14404/(x*4000))-1))};\addlegendentry{\myentry{4000}}%
\path[name path=axis] (axis cs:0,0) -- (axis cs:1,0);
\addplot+ [black,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```

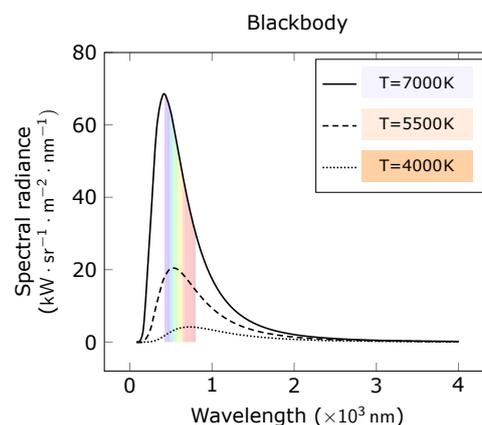


Blackbody



The above examples could be obtained with a much functional and prettier code, also proposed by Stefan Pinnow:

```
\begin{tikzpicture}[
  /pgf/declare function={
    BlackBodySpectralRadiance(\x,\T) = 119.268/(x^5*(exp(14404/(\x*\T))-1));
  },
]
  \pgfspectraplotshade[
    shade begin=0,
    shade end=4000,
    IRcolor=white,
    UVcolor=white,
    gamma=.6,
    shade opacity=.2,
  ]{BBody}
  \def\myentry#1{\tempercolor{#1}%
    \tikz{\fill [tempercolor] (0,-.5pt) rectangle (40pt,.5pt)
      node [midway,font=\footnotesize,anchor=mid]
        {\color{black} $T = #1\,\mathrm{K}$};}%
  }
  \begin{axis}[
    title=Blackbody,
    xlabel={Wavelength in $\mathrm{10^3\,nm}$},
    ylabel={%
      Spectral radiance in\
      $\mathrm{kW\cdot sr^{-1}\cdot m^{-2}\cdot nm^{-1}}$},
    ylabel style={align=center},
    ymax=80,
    cycle list name=linestyles,
    domain=0:4,
    samples=51,
    smooth,
  ]
  \pgfplotsinvokeforeach{7000,5500,4000}{
    \addplot+ [thick,name path=spectrum-#1] {BlackBodySpectralRadiance(x,#1)};
    \addlegendentry{\myentry{#1}}
  }
  \path [name path=axis] (axis cs:0,0) -- (axis cs:4,0);
  \addplot [shading=BBody] fill between [of=spectrum-7000 and axis];
  \end{axis}
\end{tikzpicture}
```



The `logarithmic` option of the `\pgfspectraplotshade` command could be used as a possible solution for the [TeX - LaTeX Stack Exchange](#) question, [How to create a electromagnetic spectrum using pgfplots package \(together with colormaps\)](#).

Filling optical spectrum curve with color gradient (first answer)

The original code lines that was replaced, in this possible answer, are commented.

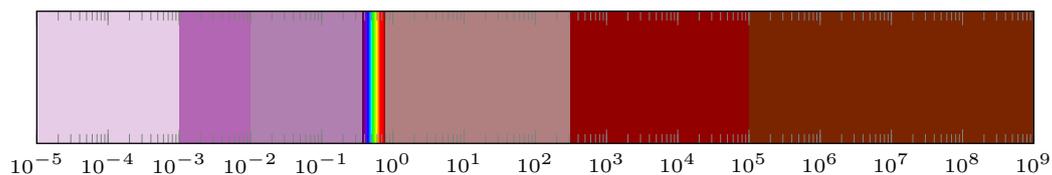
```
\documentclass[12pt]{article}
\usepackage[dvipsnames,table]{xcolor}
\usepackage{siunitx} % SI-units
\usepackage{pgf-spectra}
\usepackage{pgfplots}
\usepgfplotslibrary{units} % to add units easily to axis
\usepgfplotslibrary{fillbetween} % to fill inbetween curves
\usepgfplotslibrary{colormaps} % to create colormaps
\pgfplotsset{width=12.2cm, height=7cm}
\pgfplotsset{compat=newest} %(making it only compatalbe with
    %new releases of pgfplots)
%\pgfdeclarehorizontalshading{visiblelight}{50bp}{
%color(0.000000000000000bp)=(violet);
%color(8.333333333333333bp)=(blue);
%color(16.666666666666670bp)=(cyan);
%color(25.000000000000000bp)=(green);
%color(33.333333333333330bp)=(yellow);
%color(41.666666666666670bp)=(orange);
%color(50.000000000000000bp)=(red)
%}%
%
% make the horizontal shading and set the UV and IR colors -->
%\pgfspectraStyle[gamma=.6]% uncomment to change the gamma
\wlcolor{380}\colorlet{UV}{wlcolor}%
\wlcolor{780}\colorlet{IR}{wlcolor}%
\pgfspectraplotshade[logarithmic, UVcolor=UV]{logvisiblelight}
\pgfspectraplotshade{visiblelight}
%\pgfspectraStyleReset% uncomment to reset the style
\begin{document}
\begin{tikzpicture}[fill between/on layer={axis grid}]
\begin{axis}[
    xlabel={Wavelength},
    xticklabel style = {font=\tiny,yshift=0.2ex},
    xmin=10^-5,
    xmax=10^9,
    x unit=\si{\micro\meter},
    xmode=log,
    ymin=0,
    ymax=1,
    height=3cm,
    yticklabels={},
    ytick=\empty,
    legend cell align=left,
    legend style={at={(0.85,-0.77)},anchor=north}
]

```

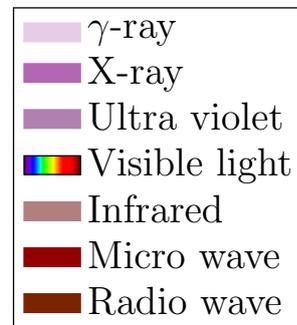
```

\addplot[draw=none, name path=start, forget plot] coordinates{(10^-5,0)(10^-5,1)};
\addplot[draw=none, name path=gamma, forget plot] coordinates{(10^-3,0)(10^-3,1)};
\addplot[draw=none, name path=xrays, forget plot] coordinates{(10^-2,0)(10^-2,1)};
%\addplot[draw=none, name path=uv, forget plot] coordinates{(0.4,0)(0.4,1)};
\addplot[draw=none, name path=uv, forget plot] coordinates{(0.38,0)(0.38,1)};
%\addplot[draw=none, name path=visible, forget plot] coordinates{(0.7,0)(0.7,1)};
\addplot[draw=none, name path=visible, forget plot] coordinates{(0.78,0)(0.78,1)};
\addplot[draw=none, name path=ir, forget plot] coordinates{(10^2.5,0)(10^2.5,1)};
\addplot[draw=none, name path=microwave, forget plot] coordinates{(10^5,0)(10^5,1)};
\addplot[draw=none, name path=radiowave, forget plot] coordinates{(10^9,0)(10^9,1)};
\addplot[violet!20, area legend] fill between[of=start and gamma];
\addlegendentry{\gamma-ray}
\addplot[violet!60, area legend] fill between[of=gamma and xrays];
\addlegendentry{X-ray}
%\addplot[violet, area legend] fill between[of=xrays and uv];
\addplot[UV!50, area legend] fill between[of=xrays and uv];
\addlegendentry{Ultra violet}
\addplot[shading=visiblelight, area legend] fill between[of=uv and visible];
\addlegendentry{Visible light}%makes the correct legend (not logarithmic)
\addplot[shading=logvisiblelight, forget plot] fill between[of=uv and visible];
%\addplot[red, area legend] fill between[of=visible and ir];
\addplot[IR!50, area legend] fill between[of=visible and ir];
\addlegendentry{Infrared}
%\addplot[Bittersweet, area legend] fill between[of=ir and microwave];
\addplot[IR!50!Bittersweet, area legend] fill between[of=ir and microwave];
\addlegendentry{Micro wave}
\addplot[Brown, area legend] fill between[of=microwave and radiowave];
\addlegendentry{Radio wave}
\end{axis}
\end{tikzpicture}
\end{document}

```



Wavelength [μm]



Filling optical spectrum curve with color gradient (second answer)

The original code lines that was replaced, in this possible answer, are commented and the code without changes was omitted.

```

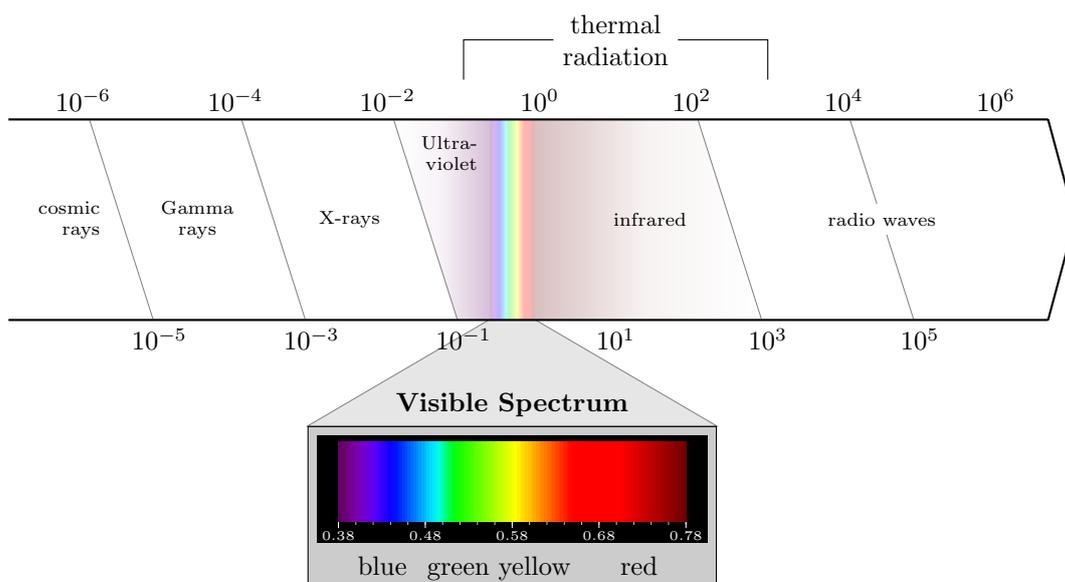
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{calc, positioning, shapes, backgrounds, fit, arrows}
\usepackage{pgf-spectra}
\usepackage{siunitx}
\usepackage{contour}
\begin{document}
%\pgfdeclarehorizontalshading{visiblelight}{50bp}{%
%   \url{https://tex.stackexchange.com/a/348492/120853}
%   color(0bp)=(violet!25);
%   color(8.33bp)=(blue!25);
%   color(16.67bp)=(cyan!25);
%   color(25bp)=(green!25);
%   color(33.33bp)=(yellow!25);
%   color(41.5bp)=(orange!25);
%   color(50bp)=(red!25)
%}%
%
% make the horizontal shading and set the UV and IR colors -->
%\pgfspectraStyle[gamma=.6]% uncomment to change the gamma
\wcolor{380}\colorlet{UV}{wcolor}%
\wcolor{780}\colorlet{IR}{wcolor}%
\pgfspectraplotshade[logarithmic,shade opacity=.3]{visiblelight}%
%\pgfspectraStyleReset% uncomment to reset the style
\begin{tikzpicture}[%
  raylabel/.style={font=\scriptsize}
]
% ... code omitted ... %
% On background layer so already drawn arrow and scale lines cover it up nicely
\begin{scope}[on background layer]
  \node[
    inner sep=0pt,
    outer sep=0pt,
    %fit={{[xshift=-2.2em]WAVELENGTH_0|-ARROW.after tail}
    %([xshift=-2.2em]WAVELENGTH_1|-ARROW.before tail)}, shading=visiblelight]
    fit={{[xshift=-1.9em]WAVELENGTH_0|-ARROW.after tail}
    ([xshift=-3em]WAVELENGTH_1|-ARROW.before tail)}, shading=visiblelight}%
    (SMALL_VISIBLE_LIGHT) {}];
  \shade[
    left color=white,
    %right color=violet!25,
    right color=UV!25,
    %middle color=violet!5,
    middle color=UV!5,
    outer sep=0pt
  ]
  % ... code omitted ... %
  \shade[
    %left color=red!25,
    left color=IR!25,
    right color=white,
    %middle color=red!5,
    middle color=IR!5,
    outer sep=0pt,
  ]
  % ... code omitted ... %
\end{scope}
\end{tikzpicture}

```

```

% Some labels can be drawn automatically at the designated label coordinates:
\foreach [count=\i] \label in {
  {Gamma\\rays},
  {X-rays},
  {},%Skip this one
  {infrared}
}{
  \node[raylabel, align=center] at (LABEL_\i) {\label};
}
% These do not fit the loop and are drawn manually:
\node[raylabel, align=right, anchor=north] at
  ([yshift=-1em,xshift=-2.5pt]$(WAVELENGTH_-2)!0.45!(WAVELENGTH_0)$)
  {Ultra-\\violet};
\node[raylabel, fill=white] at (CONNECTION_6) {radio waves};
\node[raylabel, left=0.1em of CONNECTION_1, align=right] {cosmic\\rays};
\node[
  draw,
  fill=black!20,
  below=4em of SMALL_VISIBLE_LIGHT,
  align=center,
  label=above:{\textbf{Visible Spectrum}}
] (FULL_VISIBLE_LIGHT) {%
%\pgfspectra[width=13em,height=3em]\\
\pgfspectra[width=13em,height=3em,axis,axis unit=micron,axis step=100,
  axis ticks=4,axis unit precision=2]\\%
%\pgfspectra also has a builtin axis which of course much better than
%this terrible approach, but it is in nanometer
{\num{0.38} \hfill\num{0.48} \hfill\num{0.58}%
  \hfill \num{0.68} \hfill\num{0.78}}\\
};
% ... code omitted ... %
\end{tikzpicture}
\end{document}

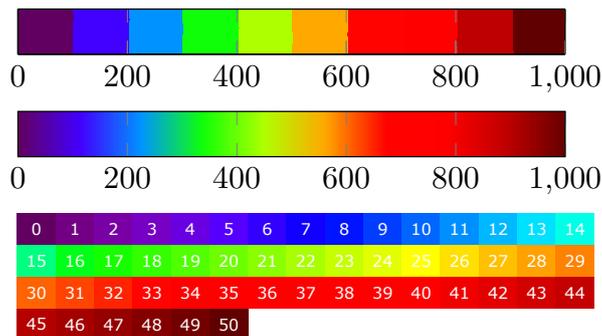
```



Next examples show possible usage of color maps feature of PGFPLOTS with the color map build with the `\pgfspectraplotmap` command:

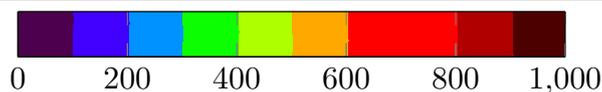
```
\pgfspectraplotmap{myColorMap}% default resolution (51 colors)
```

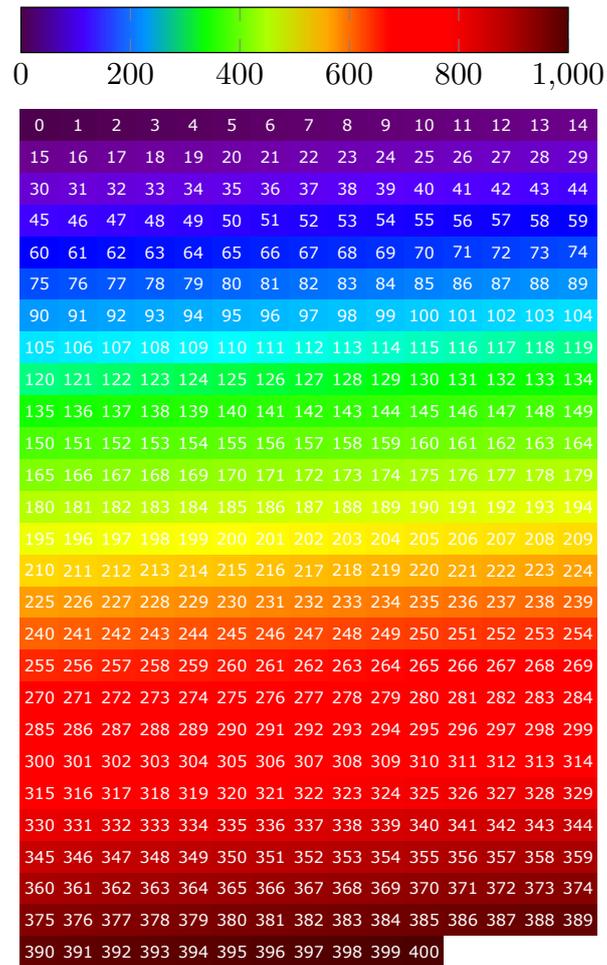
```
\pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMap)},
colorbar horizontal,colormap access=const]
\\ \pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMap)},
colorbar horizontal,colormap access=map]
% --- code improved by Stefan Pinnow --->
\begin{tikzpicture}
  \foreach \i [
    evaluate=\i as \x using {int(mod(\i,15))},
    evaluate=\i as \y using {floor(\i/15)},
  ] in {0,...,\pgfplotscolormaplastindexof{myColorMap}}{
    \fill [index of colormap={\i of myColorMap}]
      (\x*12pt,-\y*10pt) rectangle ++(12pt,10pt)
      node [inner sep=0pt,midway,font=\tiny,text=white] {\i};
  }
\end{tikzpicture}
```



```
\pgfspectraplotmap[h]{myColorMap}% high resolution (401 colors)
% color(0) -> 380nm color(1) -> 381nm ... color(60) -> 380+60=440nm ...
% ... color(400) -> 780nm
```

```
\pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMapH)},
colorbar horizontal,colormap access=const]
\\ \pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMapH)},
colorbar horizontal,colormap access=map]
% --- code improved by Stefan Pinnow --->
\begin{tikzpicture}
  \foreach \i [
    evaluate=\i as \x using {int(mod(\i,15))},
    evaluate=\i as \y using {floor(\i/15)},
  ] in {0,...,\pgfplotscolormaplastindexof{myColorMap}}{
    \fill [index of colormap={\i of myColorMap}]
      (\x*12pt,-\y*10pt) rectangle ++(12pt,10pt)
      node [inner sep=0pt,midway,font=\tiny,text=white] {\i};
  }
\end{tikzpicture}
```





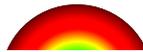
► Using `\pgfspectrarainbow`

Here are some examples of rainbows:

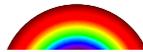
```
\pgfspectrarainbow{1cm}
```



```
\pgfspectrarainbow(rainbow start=0){1cm}
```



```
\pgfspectrarainbow(rainbow start=.4){1cm}
```



```
\pgfspectrarainbow(rainbow start=.8){1cm}
```



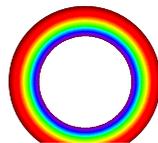
```
\pgfspectrarainbow(rainbow knock out=.8){1cm}
```



```
\pgfspectrarainbow(rainbow knock out=0){1cm}
```



```
\pgfspectrarainbow(rainbow knock out=-.8){1cm}
```



```
\pgfspectrarainbow(rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow background=white){1cm}
```



```
\pgfspectrarainbow(rainbow background=blue,rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow background=black,rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow background=white,rainbow transparency=.5){1cm}
```



```
\pgfspectrarainbow(rainbow fade=south){1cm}
```



```
\pgfspectrarainbow(rainbow fade=north){1cm}
```



```
\pgfspectrarainbow[white,path fading=south]{1cm}
```



```
\pgfspectrarainbow[white](rainbow fade=south){1cm}
```



```
\pgfspectrarainbow[orange,path fading=west](rainbow fade=south){1cm}
```



```
\pgfspectrarainbow[blue,xslant=.1,opacity=.2]{1cm}
```



Alphabetical list of available options

\pgfspectra

key	description	type	default	value(s)
absorption	minimum intensity for the lines in the spectrum when using their relative intensities	boolean	false	{true, false}
axis	show or hide the axis	boolean	false	{true, false}
axis color	color of the axis	color	black	any named color or user defined color
axis font	font of the axis labels	font commands	{\tiny}	T _E X font commands
axis font color	color of the axis labels	color	white	any named color or user defined color
axis step	interval in nanometres between two major axis ticks	integer	20	[0; end-begin]nm
axis ticks	number of minor ticks	integer	0	{0,1,2,3,...}
axis unit	unit of the axis labels	text	nm	nm or micron or A
axis unit precision	number of significant digits (for values in nanometres) shown in axis labels	integer	3	{0,1,2,3,...}
back	spectrum background color	color	black	any named color or user defined color
backIRUV	IR and UV emission lines color in emission spectrum or background color of IR and UV regions in absorption spectrum	color	black	any named color or user defined color
backVIS	visible region background color in emission spectrum or emission lines color in absorption spectrum	color	black	any named color or user defined color
begin	first wavelength, in nanometres	integer	380	[10;4000]nm
brightness	brightness color correction as in the CMYK color model	decimal	1	[0;1]
charge	charge of the element(s)	integer	0	LSE Data: {0,1,2,3,4} NIST Data: {0,1}
element	chemical symbol of one element or comma sparated list of chemical symbols elements	text	NONE	H to Es except Fr
end	last wavelength, in nanometres	integer	780	[10;4000]nm
gamma	gamma color correction at the edges of the visible region	decimal	0.8	[0;1]
height	spectrum height	length	1cm	up to maximum T _E X dimension (16384pt)
Imin	minimum intensity of the lines	decimal	0	[0;1]
IRcolor	IR emission lines color in emission spectrum or background color of IR region in absorption spectrum	color	rgb(.3157,.2373,.2373)	any named color or user defined color
label	show or hide the axis labels	boolean	false	{true, false}
label after text	extra text to place after the label of the spectrum	text	{}	
label before text	extra text to place before the label of the spectrum	text	{}	
label font	font of the spectrum label	font commands	{\bfseries\small}	
label font color	color of the font of the spectrum label	color	black	any named color or user defined color
label position	position of the label of the spectrum	text	{west}	{west, north west, north, north east, east, south east, south, south west}
line intensity	draw all lines with the same intensity value	integer	100	{0,1,2,...,99,100}
line width	width of each line drawn in the spectrum	length	1pt	up to maximum T _E X dimension (16384pt)

\pgfspectra (continuation)

key	description	type	default	value(s)
lines	number or comma sparated list of numbers	integer or decimal	{}	[10;4000]nm
redshift	computes and draws the redshifted (or blueshifted) lines	text	{}	numeric value or {numeric value 1/numeric value 2}
relative intensity	draws the lines using their relative intensities	boolean	false	{true, false}
relative intensity threshold	all lines with intensity	decimal	0.25	[0;1]
show redshift value	show or hide the redshift value	boolean	false	{true, false}
use visible shading	visible region is drawn using a shading (instead of line by line)	boolean	true	{true, false}
UVcolor	UV emission lines color in emission spectrum or background color of UV region in absorption spectrum	color	rgb(.3,.2568,.3)	any named color or user defined color
width	spectrum width	length	{0.9\textwidth}	up to maximum \TeX dimension (16384pt)

\pgfspectraplotshade

key	description	type	default	value(s)
shade begin	first wavelength, in nanometres	integer	380	[0;15999]nm
shade end	last wavelength, in nanometres	integer	780	[1;16000]nm
shade opacity	opacity of the computed shade	decimal	1	[0;1]
shade opacity color	background color of the computed shading	color	white	any named color or user defined color
logarithmic	the shading is build in a logarithmic scale	boolean	false	{true, false}

\pgfspectrarainbow

key	description	type	default	value(s)
rainbow background	background color below the rainbow	color	white	any named color or user defined color
rainbow fade	scope fading in the clipped region	text	{}	any named <i>TikZ</i> fading or user defined fading
rainbow knock out	relative distance from the half-circle base to perform the clip	decimal	.4	[-1;1]
rainbow start	fraction from which the rainbow colors begin, relative to the center of a circle with radius 1	decimal	.6	[0;1]
rainbow transparency	overall transparency of the rainbow	decimal	0	[0;1]

Recommendations and known issues

The code could be a bit slow, so if there are many spectra to draw, the time consumption to get them could be high. In that case it's preferable to compile individual spectrum via the *preview* package, for later inclusion with `\includegraphics{<filename>.pdf}`:

```

1 % <filename >.tex
2 \documentclass{article}
3 \usepackage{pgf-spectra}
4 \usepackage[active,tightpage]{preview}
5 \PreviewEnvironment{tikzpicture}
6 \setlength\PreviewBorder{1pt}%
7 %%%%%%%%%%%
8 \begin{document}
9 \pgfspectra[element=H,width=15cm]
10 \end{document}

```

Hint for T_EX 'limits':

If tex capacity exceeded when running...

«! TeX capacity exceeded, sorry [main memory size=2000001].»

just make a `\newpage` at the point of origin of the message (ejecting the page releases the T_EX memory!)

The code

```

1 % Hugo Gomes @ 15/04/2016 (v1.0)
2 % Hugo Gomes @ 15/03/2021 (v2.0.0)
3 % Hugo Gomes @ 12/05/2021 (v2.1.0)
4 % Hugo Gomes @ 27/06/2021 (v2.1.1)
5 % Hugo Gomes @ 03/07/2021 (v2.1.2)
6 %%%%%%%%%%%
7 \NeedsTeXFormat{LaTeX2e}%
8 \ProvidesPackage{pgf-spectra}[03/07/2021 pgf-spectra v2.1.2]%
9 \RequirePackage{tikz}%
10 \DeclareOption{LSE}{\input{spectra.data.LSE.tex}}%
11 \DeclareOption{NIST}{\input{spectra.data.NIST.tex}}%
12 \ExecuteOptions{NIST}%
13 \ProcessOptions\relax%
14 %%%%%%%%%%%
15 \definecolor{wlRcolor}{rgb}{.3157,.2373,.2373}% NEW v2.0.0
16 \definecolor{wlUVcolor}{rgb}{.3,.2568,.3}% NEW v2.0.0
17 %
18 \newcount\wl@counta% tmp counter
19 \newcount\wl@countb% tmp counter
20 \newcount\wl@countc% tmp counter
21 \newcount\wl@countd% tmp counter
22 %
23 \newif\ifwl@absorption%
24 \newif\ifwl@elem@exist%
25 \newif\ifwl@drawaxis%
26 \newif\ifwl@axislabel%
27 \newif\ifwl@intensity%
28 \newif\ifwl@redshift% NEW v2.0.0
29 \newif\ifwl@RSvalue% NEW v2.0.0
30 \newif\ifwl@usevisibleshade% NEW v2.1.0
31 % defining PGF keys
32 \pgfkeys{/wl/.cd,%
33 element/.get=\wl@element,%
34 element/.store in=\wl@element,%
35 element/.default=NONE,%

```

```

36 width/.get=\wl@width,%
37 width/.store in=\wl@width,%
38 width/.default={0.9\textwidth},%
39 height/.get=\wl@height,%
40 height/.store in=\wl@height,%
41 height/.default=1cm,%
42 back/.get=\wl@back,%
43 back/.store in=\wl@back,%
44 back/.default=black,%
45 backIRUV/.get=\wl@backnotvisible,% NEW v2.0.0
46 backIRUV/.store in=\wl@backnotvisible,% NEW v2.0.0
47 backIRUV/.default=black,% NEW v2.0.0
48 backVIS/.get=\wl@backvisible,% NEW v2.1.0
49 backVIS/.store in=\wl@backvisible,% NEW v2.1.0
50 backVIS/.default=black,% NEW v2.1.0
51 IRcolor/.get=\wl@IRcolor,% NEW v2.0.0
52 IRcolor/.store in=\wl@IRcolor,% NEW v2.0.0
53 IRcolor/.default=wIRcolor,% NEW v2.0.0
54 UVcolor/.get=\wl@UVcolor,% NEW v2.0.0
55 UVcolor/.store in=\wl@UVcolor,% NEW v2.0.0
56 UVcolor/.default=wUVcolor,% NEW v2.0.0
57 charge/.get=\wl@charge,%
58 charge/.store in=\wl@charge,%
59 charge/.default=0,%
60 lmin/.get=\wl@lmin,%
61 lmin/.store in=\wl@lmin,%
62 lmin/.default=0,%
63 lines/.get=\wl@lines,%
64 lines/.store in=\wl@lines,%
65 lines/.default={},%
66 line width/.get=\wl@linewidth,%
67 line width/.store in=\wl@linewidth,%
68 line width/.default=1pt,%
69 begin/.get=\wl@begin,%
70 begin/.store in=\wl@begin,%
71 begin/.default=380,%
72 end/.get=\wl@end,%
73 end/.store in=\wl@end,%
74 end/.default=780,%
75 axis step/.get=\wl@axisstep,%
76 axis step/.store in=\wl@axisstep,%
77 axis step/.default=20,%
78 axis unit/.get=\wl@axisunit,% NEW v2.1.0
79 axis unit/.store in=\wl@axisunit,% NEW v2.1.0
80 axis unit/.default=nm,% NEW v2.1.0
81 axis unit precision/.get=\wl@axisunitprecision,% NEW v2.1.0
82 axis unit precision/.store in=\wl@axisunitprecision,% NEW v2.1.0
83 axis unit precision/.default=3,% NEW v2.1.0
84 axis ticks/.get=\wl@axisticks,% NEW v2.0.0
85 axis ticks/.store in=\wl@axisticks,% NEW v2.0.0
86 axis ticks/.default=0,% NEW v2.0.0
87 axis color/.get=\wl@axiscolor,%
88 axis color/.store in=\wl@axiscolor,%
89 axis color/.default=black,%
90 axis font/.get=\wl@axisfont,%
91 axis font/.store in=\wl@axisfont,%
92 axis font/.default={\tiny},%
93 axis font color/.get=\wl@axisfontcolor,%
94 axis font color/.store in=\wl@axisfontcolor,%
95 axis font color/.default=white,%
96 label position/.get=\wl@labelposition,%
97 label position/.store in=\wl@labelposition,%
98 label position/.default={west},%
99 label before text/.get=\wl@labelbtext,%
100 label before text/.store in=\wl@labelbtext,%
101 label before text/.default={},%
102 label after text/.get=\wl@labelatext,%
103 label after text/.store in=\wl@labelatext,%
104 label after text/.default={},%
105 label font/.get=\wl@labelfont,%

```

```

106 label font/.store in=\wl@labelfont,%
107 label font/.default={\bfseries\small},%
108 label font color/.get=\wl@labelfontcolor,%
109 label font color/.store in=\wl@labelfontcolor,%
110 label font color/.default=black,%
111 gamma/.get=\wl@gamma,%
112 gamma/.store in=\wl@gamma,%
113 gamma/.default=0.8,%
114 brightness/.get=\wl@brightness,%
115 brightness/.store in=\wl@brightness,%
116 brightness/.default=1,%
117 line intensity/.get=\wl@lineint,%
118 line intensity/.store in=\wl@lineint,%
119 line intensity/.default=100,%
120 relative intensity threshold/.get=\wl@relintthresh,%
121 relative intensity threshold/.store in=\wl@relintthresh,%
122 relative intensity threshold/.default=0.25,%
123 absorption/.is if=\wl@absorption,%
124 axis/.is if=\wl@drawaxis,%
125 label/.is if=\wl@axislabel,%
126 relative intensity/.is if=\wl@intensity,%
127 redshift/.get=\wl@redshift,% NEW v2.0.0
128 redshift/.store in=\wl@redshift,% NEW v2.0.0
129 redshift/.default={},% NEW v2.0.0
130 show redshift value/.is if=\wl@RSvalue,% NEW v2.0.0
131 use visible shading/.is if=\wl@usevisibleshade% NEW v2.1.0
132 }%
133 % setting keys with default values
134 \pgfkeys{/wl/.cd,element,width,height,back,backIRUV,IRcolor,UVcolor,charge,lmin,lines,line
width,begin,end,% NEW v2.0.0 → backIRUV,IRcolor,UVcolor
135 axis color,axis font,axis font color,axis step,axis unit,axis unit precision,axis ticks,
label position,label before text,label after text,label font,label font color,gamma,
brightness,line intensity,% NEW v2.1.0 → axis unit,axis unit precision
136 relative intensity threshold,absorption=false,axis=false,label=false,relative intensity=
false,redshift,show redshift value=false,% NEW v2.0.0 → redshift, show redshift value
137 use visible shading,backVIS}% NEW v2.1.0 → use visible shading
138 % strings for \ifx tests
139 \def\wl@NE{NONE}%
140 \def\wl@ll{all}%
141 \def\wl@visible{visible}%
142 \def\wl@visible@list{visible,visible 5,visible 10,visible 15,visible 20,visible 25,visible 30,
visible 35,visible 40,visible 45,visible 50,visible 55,visible 60,visible 65,visible 70,visible 75,
visible 80,visible 85,visible 90,visible 95,visible 100}%
143 \def\wl@label@position@list{west,north west,north,north east,east,south east,south,south
west}%
144 \def\wl@redshift@D{D}% NEW v2.0.0
145 \def\wl@axisunit@nm{nm}% NEW v2.1.0
146 \def\wl@axisunit@um{micron}% NEW v2.1.0
147 \def\wl@axisunit@A{A}% NEW v2.1.0
148 %%% COMMANDS
149 % commands #####
150 \newif\ifpgfspectra@StyleIsDef\pgfspectra@StyleIsDeffalse% NEW v2.0.0
151 % \pgfspectraStyle[options]% NEW v2.0.0
152 \def\pgfspectraStyle[#1]{\pgfspectraStyleReset\pgfspectra@StyleIsDeftrue\relax\tikzset{/wl/.
cd,#1}%
153 \def\pgfspectra@DoStyle{\tikzset{/wl/.cd,#1}}% applies storing user style for future use
154 }% NEW v2.0.0
155 % \pgfspectraStyleReset% NEW v2.0.0
156 \def\pgfspectraStyleReset{\pgfspectra@StyleIsDeffalse\tikzset{/wl/.cd,%
157 element=NONE,width=0.9\textwidth,height=1cm,back=black,backIRUV=black,IRcolor=wlIRcolor,
UVcolor=wlUVcolor,%
158 charge=0,lmin=0,lines={},line width=1pt,begin=380,end=780,axis color=black,axis font=\tiny,
axis font color=white,axis step=20,axis unit=nm,axis unit precision=3,axis ticks=0,label
position=west,% NEW v2.1.0 → axis unit,axis unit precision
159 label before text={},label after text={},label font=\bfseries\small,label font color=black,
gamma=0.8,brightness=1,line intensity=100,%
160 relative intensity threshold=0.25,absorption=false,axis=false,label=false,relative intensity
=false,redshift={},show redshift value=false,% NEW v2.0.0 → redshift, show redshift value
161 use visible shading,backVIS=black}}% NEW v2.1.0 → use visible shading

```

```

162 % ————— The main command to draw the spectra
163 % \pgfspectra[options]
164 \def\pgfspectra{\@ifnextchar[\wl@pgfspectra@withoptions{\wl@pgfspectra@nooptions}}%
165 \def\wl@pgfspectra@nooptions{\wl@pgfspectra@continuous(0.9\textwidth,1cm)}%
166 % #####
167 \def\wl@pgfspectra@continuous(#1,#2){\ignorespaces%
168 \ifwl@usevisibleshade% NEW v2.1.0
169 \pgfspectrashade(380,780){\wl@visibleshade}%
170 \tikz{\fill[shading=\wl@visibleshade](0,0) rectangle (#1,#2);}%
171 \else%
172 \begin{tikzpicture}%
173 \pgfmathparse{#1/400}\edef\xscale{\pgfmathresult}
174 \pgfmathparse{1.4*\xscale+.09*\linewidth/\wl@width}\edef\wl@linewidth{\pgfmathresult}% NEW v
175 2.0.0 \xscale} → {1.4*\xscale+.09*\linewidth/\wl@width}
176 \foreach \x in {380,...,780}%
177 {%
178 \wlcolor{\x}%
179 \pgfmathparse{(\x-380)*\xscale}\edef\wl@currentx{\pgfmathresult pt}%
180 \draw[\wl@temp,line width=\wl@linewidth](\wl@currentx,0) — ++(0,#2);%
181 }%
182 \end{tikzpicture}%
183 \fi%
184 % #####
185 \def\wl@pgfspectra@withoptions[#1]{\ignorespaces%
186 % setting default values or user style
187 \ifpgfspectra@StyleIsDef\pgfspectraStyleReset\pgfspectra@DoStyle\pgfspectra@StyleIsDeftrue
188 \relax\else\pgfspectraStyleReset\relax\fi% NEW v2.0.0
189 % process options (key values)
190 \pgfkeys{/wl/.cd,#1}%
191 % axis height
192 \setbox0=\hbox{\wl@axisfont\selectfont380}\edef\wl@axis@height{\the\ht0}%
193 % process visible background (visible+opacity)
194 \wl@counta=0%
195 \wl@countb=-1%
196 \@for\@myarg:=\wl@visible@list\do{%
197 \ifx\wl@back\@myarg\wl@countb=\wl@counta\fi%
198 \advance\wl@counta by1%
199 }%
200 \ifnum\wl@countb=-1\edef\@visible@opacity{1}\else% NEW v2.0.0
201 \ifnum\wl@countb=0\let\wl@back\wl@visible\edef\@visible@opacity{.5}\else%
202 \ifnum\wl@countb>0\let\wl@back\wl@visible\pgfmathparse{.05*\wl@countb}\edef\@visible@opacity
203 {\pgfmathresult}\fi\fi\fi%
204 %
205 % check limits ... % NEW v2.0.0
206 \ifnum\wl@end<10\relax\def\wldesz{10}\let\wl@end\wldesz\fi%
207 \ifnum\wl@end>4000\relax\def\wlquatomil{4000}\let\wl@end\wlquatomil\fi%
208 \ifnum\wl@begin<10\relax\def\wldesz{10}\let\wl@begin\wldesz\fi%
209 \ifnum\wl@begin>4000\relax\def\wlquatomil{4000}\let\wl@begin\wlquatomil\fi%
210 %
211 % verifying redshift key
212 \ifx\wl@redshift\@empty\relax%
213 \wl@redshiftfalse%
214 \else%
215 \wl@processredshiftkey\wl@redshift\relax%
216 \fi%
217 %
218 % if no element provided draws continuous spectrum with options or user list of lines
219 \ifx\wl@element\wlN@NE%no element by the user
220 \ifx\wl@elt@chemsym\undefined\else\let\wl@elt@chemsym\undefined\fi%
221 \ifx\wl@lines\@empty%no lines by the user => continuous spectrum
222 % draws the continuous spectrum width options (default or by the user)
223 \begin{tikzpicture}%

```

```

222     \pgfkeys{/wl/.cd,#1}% NEW v2.0.0
223     \pgfmathparse{\wl@width/(abs(\wl@end-\wl@begin))}\edef\xscale{\pgfmathresult}%
224     \ifwl@drawaxis%draws the axis
225     \wl@utils@draw@axis%
226     \fi%\ifwl@drawaxis
227     \ifwl@axislabel%put the label
228     \wl@utils@put@label%
229     \fi%\ifwl@axislabel
230     \let\wl@back\wl@visible%
231     \let\wl@background@UVcolor\wl@backnotvisible\let\wl@background@IRcolor
        \wl@backnotvisible\relax%
232     \wl@utils@drawbackground{\@visible@opacity*\wl@brightness}%
233 \end{tikzpicture}%
234 \let\wl@list@@\@empty%
235 \else% lines by the user
236 \edef\wl@list@@{\wl@lines}%
237 \let\wl@background@UVcolor\wl@backnotvisible\let\wl@background@IRcolor\wl@backnotvisible
        \relax%
238 \fi%\wl@lines\@empty
239 \else%\wl@element\wN@NE
240 % else get element(s) data
241 \wl@countc=0%
242 \wl@countd=1%
243 \@for\@myarg:=\wl@element\do{\advance\wl@countc by1}%count number of elements
244 \wl@addt@list{,}%
245 \@for\@myarg:=\wl@element\do{%
246     \cur@elem@existtrue%
247     \def\wl@elt@chemsym{NOT FOUND!}%
248     \def\@search@result@err{NOT FOUND!}%
249     \wl@elt@data{\@myarg}\relax%
250     % check if element provided exists
251     \ifx\@search@result@err\wl@elt@chemsym Element\ '@@myarg' with charge ''
        \wl@charge'' not found!\cur@elem@existfalse\else%
252     % if exists, set the wavelength's list
253     \wl@set@element@list{\wl@elt@elemdata}{\wl@elt@lmax}%
254     \fi%\@search@result@err\wl@elt@chemsym
255     \ifcur@elem@exist\ifnum\wl@countd<\wl@countc\wl@addt@list{\wl@list@@}{,}\fi\fi%
256     \advance\wl@countd by1%
257     }%end do
258 \fi%\wl@element\wN@NE
259 % check if there are lines to draw and make the spectrum
260 \ifx\wl@list@@\@empty\ifx\wl@element\wN@NE\else Element\ '@@wl@element' with charge ''
        \wl@charge'' have no lines to display.\fi\else%
261     \ifwl@absorption%absortion spectrum
262     \begin{tikzpicture}%
263         \pgfkeys{/wl/.cd,#1}% NEW v2.0.0
264         \pgfmathparse{\wl@width/(abs(\wl@end-\wl@begin))}\edef\xscale{\pgfmathresult}
        }%
265         \ifwl@drawaxis%draws the axis
266         \wl@utils@draw@axis%
267         \fi%\ifwl@drawaxis
268         \ifwl@axislabel%put the label
269         \wl@utils@put@label%
270         \fi%\ifwl@axislabel
271         \let\wl@back\wl@visible%
272         \let\wl@background@UVcolor\wl@UVcolor\let\wl@background@IRcolor\wl@IRcolor
            \relax%
273         \wl@utils@drawbackground{\wl@brightness}%
274         % draws the lines
275         \wl@utils@drawabsorptionlines%
276     \end{tikzpicture}%
277     \else%emission spectrum
278     % draws the spectrum
279     \ifx\wl@back\wl@visible%visible background
280     \begin{tikzpicture}%
281         \pgfkeys{/wl/.cd,#1}% NEW v2.0.0
282         \pgfmathparse{\wl@width/(abs(\wl@end-\wl@begin))}\edef\xscale{\pgfmathresult}
        }%
283         \ifwl@drawaxis%draws the axis
284         \wl@utils@draw@axis%

```

```

285         \fi%\ifwl@drawaxis
286         \ifwl@axislabel%put the label
287         \wl@utils@put@label%
288         \fi%\ifwl@axislabel
289         \let\wl@background@UVcolor\wl@backnotvisible \let \wl@background@IRcolor
           \wl@backnotvisible \relax%
290         \wl@utils@drawbackground{\@visible@opacity*\wl@brightness}%
291         \wl@utils@drawemissionlines% emission lines
292     \end{tikzpicture}%
293     \else%without visible background
294     \begin{tikzpicture}%
295         \pgfkeys{/wl/.cd,#1}% NEW v2.0.0
296         \pgfmathparse{\wl@width/(abs(\wl@end-\wl@begin))}\edef\xscale{\pgfmathresult}
           }%
297         \ifwl@drawaxis%draws the axis
298         \wl@utils@draw@axis%
299         \fi%\ifwl@drawaxis
300         \ifwl@axislabel%put the label
301         \wl@utils@put@label%
302         \fi%\ifwl@axislabel
303         \let\wl@background@UVcolor\wl@backnotvisible \let \wl@background@IRcolor
           \wl@backnotvisible \relax%
304         \wl@utils@drawbackground{0}% dummy argument
305         \wl@utils@drawemissionlines% emission lines
306     \end{tikzpicture}%
307     \fi%\wl@back\@visible
308     \fi%\ifwl@absorption
309     \fi% \wl@list@\@empty
310 }%
311 % #####
312 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
313 % get individual line data from one element of the array data
314 \def\wl@get@line@info[#1 #2 #3]{%
315 \def\@currentline@wl{#1}% return
316 \def\@currentline@int{#2}% return
317 \def\@currentline@charge{#3}% return
318 }%
319 % #####
320 % \wl@set@element@list \wl@elt@elemdata}{\wl@elt@lmax}
321 % #####
322 %
323 % \wl@set@element@list{\wl@elt@elemdata}{\wl@elt@lmax}
324 % RETURN: \wl@list@@ -> (wl1,wl2,...)
325 % or if relative intensity true (between 0 and 1)
326 % \wl@list@@ -> (wl1/int 1,wl2/int 2,...)
327 %
328 \newif\ifwl@first% for first occurrence of lmin
329 \def\wl@set@element@list#1#2{\ignorespaces \wl@elt@lmax
330 \wl@firsttrue%
331 \wl@counta=0%
332 \wl@countb=1%
333 \pgfmathparse{int(\wl@intmin*100)}\edef\@wl@intmin{\pgfmathresult}%intensity percentage
334 \ifnum\@wl@intmin=0% include all intensities
335 \ifx\wl@l\wl@charge%ALL lines
336 \@for\@myarg:=#1\do{\advance\wl@counta by1}%count all entries
337 \ifwl@intensity%
338 \@for\@myarg:=#1%
339 \do{%
340 \expandafter\wl@get@line@info\@myarg%
341 \pgfmathparse{\wl@relintthresh+(1-\wl@relintthresh)*\@currentline@int/#2}\edef
           \wl@intensity@to@list{\pgfmathresult}%
342 \ifnum\wl@countb<\wl@counta\wl@addt@list{\wl@list@@}{\@currentline@wl/
           \wl@intensity@to@list,} \else%
343 \wl@addt@list{\wl@list@@}{\@currentline@wl/\wl@intensity@to@list} \fi%
344 \advance\wl@countb by1%
345 }%END do
346 \else%
347 \@for\@myarg:=#1%
348 \do{%
349 \expandafter\wl@get@line@info\@myarg%

```

```

350     \ifnum\wl@countb<\wl@counta\wl@addt@list{\wl@list@@}{\@currentline@wl,} \else%
351     \wl@addt@list{\wl@list@@}{\@currentline@wl} \fi%
352     \advance\wl@countb by 1%
353   }%END do
354   \fi%
355 \else% lines for one specific charge
356 \@for\@myarg:=#1\do{\expandafter\wl@get@line@info\@myarg\ifx\@currentline@charge\wl@charge
  \advance\wl@counta by 1\fi}%count only if is the desired charge
357   \ifwl@intensity%
358   \@for\@myarg:=#1%
359   \do{%
360     \expandafter\wl@get@line@info\@myarg%
361     \pgfmathparse{\wl@relintthresh+(1-\wl@relintthresh)*\@currentline@int/#2}\edef
      \wl@intensity@to@list{\pgfmathresult}%
362     \ifx\@currentline@charge\wl@charge%add to list if is the desired charge
363     \ifnum\wl@countb<\wl@counta\wl@addt@list{\wl@list@@}{\@currentline@wl/
      \wl@intensity@to@list,} \else%
364     \wl@addt@list{\wl@list@@}{\@currentline@wl/\wl@intensity@to@list} \fi%
365     \advance\wl@countb by 1%
366     \fi%
367   }%END do
368   \else%
369   \@for\@myarg:=#1%
370   \do{%
371     \expandafter\wl@get@line@info\@myarg%
372     \ifx\@currentline@charge\wl@charge%add to list if is the desired charge
373     \ifnum\wl@countb<\wl@counta\wl@addt@list{\wl@list@@}{\@currentline@wl,} \else%
374     \wl@addt@list{\wl@list@@}{\@currentline@wl} \fi%
375     \advance\wl@countb by 1%
376     \fi%
377   }%END do
378   \fi%
379 \fi%
380 \else%\wl@intmin>0 & \wl@intmin<1
381 \ifnum\@wl@intmin>100\else%
382 \pgfmathparse{\wl@intmin*#2}\edef\wl@actual@int{\pgfmathresult}%
383 \ifx\wl@ll\wl@charge%ALL lines
384 \@for\@myarg:=#1\do{\advance\wl@counta by 1}%count all entries
385   \ifwl@intensity%
386   \@for\@myarg:=#1%
387   \do{%
388     \expandafter\wl@get@line@info\@myarg%
389     \pgfmathparse{notless(\@currentline@int, \wl@actual@int)} \relax\edef\wl@int@result{
      \pgfmathresult}%
390     \ifnum\wl@int@result=1%
391     \pgfmathparse{\wl@relintthresh+(1-\wl@relintthresh)*\@currentline@int/#2}\edef
      \wl@intensity@to@list{\pgfmathresult}%
392     \ifwl@first\wl@addt@list{\wl@list@@}{\@currentline@wl/\wl@intensity@to@list}
      \else%
393     \wl@addt@list{\wl@list@@}{, \@currentline@wl/\wl@intensity@to@list} \fi%
394     \ifwl@first\wl@firstfalse \fi%
395     \fi%
396     \advance\wl@countb by 1%
397   }%END do
398   \else%
399   \@for\@myarg:=#1%
400   \do{%
401     \expandafter\wl@get@line@info\@myarg%
402     \pgfmathparse{notless(\@currentline@int, \wl@actual@int)} \relax\edef\wl@int@result{
      \pgfmathresult}%
403     \ifnum\wl@int@result=1%
404     \ifwl@first\wl@addt@list{\wl@list@@}{\@currentline@wl} \else%
405     \wl@addt@list{\wl@list@@}{, \@currentline@wl} \fi%
406     \ifwl@first\wl@firstfalse \fi%
407     \fi%
408     \advance\wl@countb by 1%
409   }%END do
410   \fi%
411 \else% lines for one specific charge

```

```

412 \@for\@myarg:=#1\do{\expandafter\wl@get@line@info\@myarg\ifx\@currentline@charge\wl@charge
  \advance\wl@counta by 1\fi}%count only if is the desired charge
413   \ifwl@intensity%
414   \@for\@myarg:=#1%
415   \do{%
416   \expandafter\wl@get@line@info\@myarg%
417   \ifx\@currentline@charge\wl@charge%add to list if is the desired charge
418     \pgfmathparse{notless(\@currentline@int,\wl@actual@int)}\edef\wl@int@result{
       \pgfmathresult}%
419     \ifnum\wl@int@result=1%
420     \pgfmathparse{\wl@relintthresh+(1-\wl@relintthresh)*\@currentline@int/#2}\edef
       \wl@intensity@to@list{\pgfmathresult}%
421     \ifwl@first\wl@addt@list{\wl@list@@}{\@currentline@wl/\wl@intensity@to@list}
       \else%
422     \wl@addt@list{\wl@list@@}{,\@currentline@wl/\wl@intensity@to@list}\fi%
423     \ifwl@first\wl@firstfalse\fi%
424   \fi%
425   \advance\wl@countb by 1%
426   \fi%
427 }%END do
428 \else%
429 \@for\@myarg:=#1%
430 \do{%
431 \expandafter\wl@get@line@info\@myarg%
432 \ifx\@currentline@charge\wl@charge%add to list if is the desired charge
433   \pgfmathparse{notless(\@currentline@int,\wl@actual@int)}\edef\wl@int@result{
     \pgfmathresult}%
434   \ifnum\wl@int@result=1%
435   \ifwl@first\wl@addt@list{\wl@list@@}{\@currentline@wl}\else%
436   \wl@addt@list{\wl@list@@}{,\@currentline@wl}\fi%
437   \ifwl@first\wl@firstfalse\fi%
438 \fi%
439 \advance\wl@countb by 1%
440 \fi%
441 }%END do
442 \fi%
443 \fi%
444 \fi%
445 \fi%
446 }%
447 % add to list
448 \def\wl@addt@list#1#2{\edef\wl@list@@{#1#2}}%
449 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
450 % internal utils
451 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
452 \def\wl@utils@draw@axis{\ignorespaces%
453   % axis unit -> NEW v2.1.0
454   \pgfkeys{/pgf/number format/.cd, fixed, precision=\wl@axisunitprecision, set
     thousands separator={}, assume math mode=true}
455   \ifx\wl@axisunit\wl@axisunit@nm\relax%
456     \def\wl@axisunit@scale{1}\def\wl@axisunit@addzeros{}%
457   \else\ifx\wl@axisunit\wl@axisunit@um\relax%
458     \def\wl@axisunit@scale{1000}\def\wl@axisunit@addzeros{\pgfkeys{/pgf/
       number format/.cd, fixed zerofill}}%
459   \else\ifx\wl@axisunit\wl@axisunit@A\relax%
460     \def\wl@axisunit@scale{1}\def\wl@axisunit@addzeros{0}% probably not the
       best solution to overcome the TeX dimension limit values (16384pt)...
       but works!
461   \fi\fi\fi%
462   \ifnum\wl@begin>\wl@end%
463   % New xshift={-2.5*\wl@axis@height} to hold bigger numbers, e.g. 2500
464   \draw[draw=none, fill=\wl@axiscolor] ([xshift={2.5*\wl@axis@height}]0,
     \wl@height+2.5pt) rectangle ([xshift={-2.5*\wl@axis@height}]-\wl@width
     ,-2.5*\wl@axis@height);%
465     % minor ticks -> NEW v2.0.0
466     \ifnum\wl@axisticks>0\relax%
467     \pgfmathparse{\wl@end+\wl@axisstep/(\wl@axisticks+1)}\pgfmathparse{
       int(\pgfmathresult)}%
468     \edef\@axis@list{\wl@end,\pgfmathresult,...,\wl@begin}%
469     \foreach \x in \@axis@list%

```

```

470         {%
471         \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult
         pt}%
472         \draw[\wl@axisfontcolor!80!transparent,line width=.25pt] (
         \wl@currentx,-.375*\wl@axis@height) — ++(0,.375*\wl@axis@height
         );%
473         }%
474         \fi%
475         \pgfmathparse{\wl@end+\wl@axisstep}\pgfmathparse{int(\pgfmathresult)}%
476         \edef\@axis@list{\wl@end,\pgfmathresult,...,\wl@begin}%
477         \foreach \x in \@axis@list%
478         {%
479         \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult pt}%
480         \draw[\wl@axisfontcolor,line width=.25pt] (\wl@currentx,-.75*
         \wl@axis@height) — ++(0,.75*\wl@axis@height);%
481         \pgfmathparse{\x/\wl@axisunit@scale}\edef\wl@xscaledvalue{\pgfmathresult}%
         NEW v2.1.0
482         \node[\wl@axisfontcolor,font=\wl@axisfont,above,inner sep=0pt] at (
         \wl@currentx,-2.25*\wl@axis@height) {\pgfmathprintnumber{\wl@xscaledvalue
         }\wl@axisunit@addzeros};%
483         }%
484         \else%
485         \draw[draw=none,fill=\wl@axiscolor] ([xshift={-2.5*\wl@axis@height}]0,
         \wl@height+2.5pt) rectangle ([xshift={2.5*\wl@axis@height}]\wl@width
         ,-2.5*\wl@axis@height);%
486         % minor ticks -> NEW v2.0.0
487         \ifnum\wl@axisticks>0\relax%
488         \pgfmathparse{\wl@begin+\wl@axisstep/(\wl@axisticks+1)}\pgfmathparse
         {int(\pgfmathresult)}%
489         \edef\@axis@list{\wl@begin,\pgfmathresult,...,\wl@end}%
490         \foreach \x in \@axis@list%
491         {%
492         \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{
         \pgfmathresult pt}%
493         \draw[\wl@axisfontcolor!80!transparent,line width=.25pt] (
         \wl@currentx,-.375*\wl@axis@height) — ++(0,.375*\wl@axis@height
         );%
494         }%
495         \fi%
496         \pgfmathparse{\wl@begin+\wl@axisstep}\pgfmathparse{int(\pgfmathresult)}%
497         \edef\@axis@list{\wl@begin,\pgfmathresult,...,\wl@end}%
498         \foreach \x in \@axis@list%
499         {%
500         \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{\pgfmathresult pt}%
501         \draw[\wl@axisfontcolor,line width=.25pt] (\wl@currentx,-.75*
         \wl@axis@height) — ++(0,.75*\wl@axis@height);%
502         \pgfmathparse{\x/\wl@axisunit@scale}\edef\wl@xscaledvalue{\pgfmathresult}%
         NEW v2.1.0
503         \node[\wl@axisfontcolor,font=\wl@axisfont,above,inner sep=0pt] at (
         \wl@currentx,-2.25*\wl@axis@height) {\pgfmathprintnumber{\wl@xscaledvalue
         }\wl@axisunit@addzeros};%
504         }%
505         \fi%
506     }%
507 \def\wl@utils@put@label{\ignorespaces%
508     \ifx\wl@elt@chemsym\undefined\def\wl@elt@chemsym{}\fi%
509     \wl@get@label@position%
510     \ifnum\wl@begin>\wl@end%
511         \ifcase\wl@label@position%
512             %west
513             \ifwl@drawaxis%\ifwl@axislabel%
514             \node[\wl@labelfontcolor,font=\wl@labelfont,left,minimum width=2
         em,align=right] at (-2.5*\wl@axis@height-\wl@width,0.5*
         \wl@height-1.25*\wl@axis@height) {\wl@labelbtext
         \wl@elt@chemsym\wl@labelatext};%
515             \else%
516             \node[\wl@labelfontcolor,font=\wl@labelfont,left,minimum width=2
         em,align=right] at (-\wl@width,0.5*\wl@height) {\wl@labelbtext
         \wl@elt@chemsym\wl@labelatext};%
517             \fi%

```

```

518 \or%north west
519 \ifwl@drawaxis%
520 \node[\wl@labelfontcolor, font=\wl@labelfont, above right, inner
xsep=0pt] at (-2.5*\wl@axis@height-\wl@width, \wl@height) {
\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
521 \else%
522 \node[\wl@labelfontcolor, font=\wl@labelfont, above right, inner
xsep=0pt] at (-\wl@width, \wl@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
523 \fi%
524 \or%north
525 \node[\wl@labelfontcolor, font=\wl@labelfont, above] at (-0.5*
\wl@width, \wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
};%
526 \or%north east
527 \ifwl@drawaxis%
528 \node[\wl@labelfontcolor, font=\wl@labelfont, above left, inner xsep=0
pt] at (2.5*\wl@axis@height, \wl@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
529 \else%
530 \node[\wl@labelfontcolor, font=\wl@labelfont, above left, inner xsep=0
pt] at (0, \wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
};%
531 \fi
532 \or%east
533 \ifwl@drawaxis%
534 \node[\wl@labelfontcolor, font=\wl@labelfont, right] at (2.5*
\wl@axis@height, 0.5*\wl@height-1.25*\wl@axis@height) {
\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
535 \else%
536 \node[\wl@labelfontcolor, font=\wl@labelfont, right] at (0, 0.5*
\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
537 \fi%
538 \or%south east
539 \ifwl@drawaxis%
540 \node[\wl@labelfontcolor, font=\wl@labelfont, below left, inner
xsep=0pt] at (2.5*\wl@axis@height, -2.5*\wl@axis@height) {
\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
541 \else%
542 \node[\wl@labelfontcolor, font=\wl@labelfont, below left, inner
xsep=0pt] at (0, 0) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%
543 \fi%
544 \or%south
545 \ifwl@drawaxis%
546 \node[\wl@labelfontcolor, font=\wl@labelfont, below] at (-0.5*
\wl@width, -2.5*\wl@axis@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
547 \else%
548 \node[\wl@labelfontcolor, font=\wl@labelfont, below] at (-0.5*
\wl@width, 0) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
549 \fi%
550 \or%south west
551 \ifwl@drawaxis%
552 \node[\wl@labelfontcolor, font=\wl@labelfont, below right, inner
xsep=0pt] at (-2.5*\wl@axis@height-\wl@width, -2.5*
\wl@axis@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
};%
553 \else%
554 \node[\wl@labelfontcolor, font=\wl@labelfont, below right, inner
xsep=0pt] at (-\wl@width, 0) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%
555 \fi%
556 \else%
557 \ifcase \wl@label@position%
558 %west
559 \ifwl@drawaxis%
560 \node[\wl@labelfontcolor, font=\wl@labelfont, left, minimum width=2
em, align=right] at (-2.5*\wl@axis@height, 0.5*\wl@height-1.25*

```

```

        \wl@axis@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
    };%
562     \else%
563     \node[\wl@labelfontcolor,font=\wl@labelfont,left,minimum width=2
        em,align=right] at (0,0.5*\wl@height) {\wl@labelbtext
        \wl@elt@chemsym\wl@labelatext};%
564     \fi%
565     \or%north west
566     \ifwl@drawaxis%
567     \node[\wl@labelfontcolor,font=\wl@labelfont,above right,inner
        xsep=0pt] at (-2.5*\wl@axis@height,\wl@height) {
        \wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
568     \else%
569     \node[\wl@labelfontcolor,font=\wl@labelfont,above right,inner
        xsep=0pt] at (0,\wl@height) {\wl@labelbtext\wl@elt@chemsym
        \wl@labelatext};%
570     \fi%
571     \or%north
572     \node[\wl@labelfontcolor,font=\wl@labelfont,above] at (0.5*\wl@width
        ,\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
573     \or%north east
574     \ifwl@drawaxis%
575     \node[\wl@labelfontcolor,font=\wl@labelfont,above left,inner
        xsep=0pt] at (\wl@width+2.5*\wl@axis@height,\wl@height) {
        \wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
576     \else%
577     \node[\wl@labelfontcolor,font=\wl@labelfont,above left,inner
        xsep=0pt] at (\wl@width,\wl@height) {\wl@labelbtext
        \wl@elt@chemsym\wl@labelatext};%
578     \fi%
579     \or%east
580     \ifwl@drawaxis%
581     \node[\wl@labelfontcolor,font=\wl@labelfont,right] at ([xshift
        ={2.5*\wl@axis@height}]\wl@width,0.5*\wl@height-1.25*
        \wl@axis@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
        };%
582     \else%
583     \node[\wl@labelfontcolor,font=\wl@labelfont,right] at (\wl@width
        ,0.5*\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
        };%
584     \fi%
585     \or%south east
586     \ifwl@drawaxis%
587     \node[\wl@labelfontcolor,font=\wl@labelfont,below left,inner
        xsep=0pt] at (\wl@width+2.5*\wl@axis@height,-2.5*
        \wl@axis@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
        };%
588     \else%
589     \node[\wl@labelfontcolor,font=\wl@labelfont,below left,inner
        xsep=0pt] at (\wl@width,0) {\wl@labelbtext\wl@elt@chemsym
        \wl@labelatext};%
590     \fi%
591     \or%south
592     \ifwl@drawaxis%
593     \node[\wl@labelfontcolor,font=\wl@labelfont,below] at (0.5*
        \wl@width,-2.5*\wl@axis@height) {\wl@labelbtext
        \wl@elt@chemsym\wl@labelatext};%
594     \else%
595     \node[\wl@labelfontcolor,font=\wl@labelfont,below] at (0.5*
        \wl@width,0) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
596     \fi%
597     \or%south west
598     \ifwl@drawaxis%
599     \node[\wl@labelfontcolor,font=\wl@labelfont,below right,inner
        xsep=0pt] at (-2.5*\wl@axis@height,-2.5*\wl@axis@height) {
        \wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
600     \else%
601     \node[\wl@labelfontcolor,font=\wl@labelfont,below right,inner
        xsep=0pt] at (0,0) {\wl@labelbtext\wl@elt@chemsym
        \wl@labelatext};%

```

```

602             \fi%
603         \fi%
604     \fi%
605 }%
606 \def\wl@utils@drawbackground#1{\ignorespaces% NEW v2.0.0 – in this version draws from UV,
607     VIS to IV → replaces \wl@utils@visiblespectrum#1
608 % reprocess visible background (only visible) → needed because of the override in keys
609     \wl@counta=0%
610     \wl@countb=-1%
611     \@for\@myarg:=\wl@visible@list\do{%
612         \ifx\wl@back\@myarg\wl@countb=\wl@counta\fi%
613         \advance\wl@counta by1%
614     }%
615     \ifnum\wl@countb>-1\let\wl@back\wl@visible\fi%
616     \ifwl@usevisibleshade\relax% NEW v2.1.0
617         \pgfmathparse{int(#1*100)}\edef\wl@bright{\pgfmathresult}%
618         \ifnum\wl@begin>\wl@end% 0
619             \ifnum\wl@end<380\relax% 1
620                 \pgfmathparse{(\wl@end-380)*\xscale}\edef\wl@pointA{\pgfmathresult
621                     pt}%
622                 \ifnum\wl@begin>780\relax% 2
623                     \pgfmathparse{(\wl@end-780)*\xscale}\edef\wl@pointB{
624                         \pgfmathresult pt}%
625                     \draw[draw=none, fill=\wl@background@UVcolor] (0,0) rectangle (
626                         \wl@pointA,\wl@height);%
627                     \draw[draw=none, fill=\wl@background@IRcolor] (\wl@pointB,0)
628                         rectangle (-\wl@width,\wl@height);%
629                     \ifx\wl@back\wl@visible% 3 visible background
630                         \pgfspectrashade(380,780){\wl@visibleshade}%
631                         \fill[shading=\wl@visibleshade, shading angle=180] (
632                             \wl@pointA,0) rectangle (\wl@pointB,\wl@height);%
633                     \else% 3
634                         \draw[draw=none, fill=\wl@back] (\wl@pointA,0) rectangle
635                             (\wl@pointB,\wl@height);%
636                     \fi% 3
637                 \else% 2
638                     \pgfmathparse{(\wl@end-\wl@begin)*\xscale}\edef\wl@pointB{
639                         \pgfmathresult pt}%
640                     \draw[draw=none, fill=\wl@background@UVcolor] (0,0) rectangle (
641                         \wl@pointA,\wl@height);%
642                     \ifx\wl@back\wl@visible% 3 visible background
643                         \pgfspectrashade(380,\wl@begin){\wl@visibleshade}%
644                         \fill[shading=\wl@visibleshade, shading angle=180] (
645                             \wl@pointA,0) rectangle (\wl@pointB,\wl@height);%
646                     \else% 3
647                         \draw[draw=none, fill=\wl@back] (\wl@pointA,0) rectangle
648                             (\wl@pointB,\wl@height);%
649                     \fi% 3
650                 \else% 1
651                     \ifnum\wl@begin>780\relax% 2
652                         \pgfmathparse{(\wl@end-780)*\xscale}\edef\wl@pointB{
653                             \pgfmathresult pt}%
654                         \draw[draw=none, fill=\wl@background@IRcolor] (\wl@pointB,0)
655                             rectangle (-\wl@width,\wl@height);%
656                         \ifx\wl@back\wl@visible% 3 visible background
657                             \pgfspectrashade(\wl@end,780){\wl@visibleshade}%
658                             \fill[shading=\wl@visibleshade, shading angle=180] (0,0)
659                                 rectangle (\wl@pointB,\wl@height);%
660                         \else% 3
661                             \draw[draw=none, fill=\wl@back] (0,0) rectangle (
662                                 \wl@pointB,\wl@height);%
663                         \fi% 3
664                     \else% 2
665                         \pgfmathparse{(\wl@end-\wl@begin)*\xscale}\edef\wl@pointB{
666                             \pgfmathresult pt}%
667                         \ifx\wl@back\wl@visible% 3 visible background
668                             \pgfspectrashade(\wl@end,\wl@begin){\wl@visibleshade}%
669                             \fill[shading=\wl@visibleshade, shading angle=180] (0,0)
670                                 rectangle (\wl@pointB,\wl@height);%

```

```

655         \else% 3
656             \draw[draw=none, fill=\wl@back] (0,0) rectangle (
657                 \wl@pointB, \wl@height);%
658         \fi% 3
659     \fi% 2
660 \else% 1
661     \ifnum\wl@begin<380\relax% 1
662         \pgfmathparse{(380-\wl@begin)*\xscale}\edef\wl@pointA{\pgfmathresult
663             pt}%
664         \ifnum\wl@end>780\relax% 2
665             \pgfmathparse{(780-\wl@begin)*\xscale}\edef\wl@pointB{
666                 \pgfmathresult pt}%
667             \draw[draw=none, fill=\wl@background@UVcolor] (0,0) rectangle (
668                 \wl@pointA, \wl@height);%
669             \draw[draw=none, fill=\wl@background@IRcolor] (\wl@pointB,0)
670                 rectangle (\wl@width, \wl@height);%
671             \ifx\wl@back\wl@visible% 3 visible background
672                 \pgfspectrashade(380,780){\wl@visibleshade}%
673                 \fill[shading=\wl@visibleshade] (\wl@pointA,0) rectangle
674                 (\wl@pointB, \wl@height);%
675             \else% 3
676                 \draw[draw=none, fill=\wl@back] (\wl@pointA,0) rectangle
677                 (\wl@pointB, \wl@height);%
678             \fi% 3
679         \else% 2
680             \pgfmathparse{((\wl@end-\wl@begin)*\xscale)\edef\wl@pointB{
681                 \pgfmathresult pt}%
682             \draw[draw=none, fill=\wl@background@UVcolor] (0,0) rectangle (
683                 \wl@pointA, \wl@height);%
684             \ifx\wl@back\wl@visible% 3 visible background
685                 \pgfspectrashade(380,\wl@end){\wl@visibleshade}%
686                 \fill[shading=\wl@visibleshade] (\wl@pointA,0) rectangle
687                 (\wl@pointB, \wl@height);%
688             \else% 3
689                 \draw[draw=none, fill=\wl@back] (\wl@pointA,0) rectangle
690                 (\wl@pointB, \wl@height);%
691             \fi% 3
692         \else% 1
693             \ifnum\wl@end>780\relax% 2
694                 \pgfmathparse{(780-\wl@begin)*\xscale}\edef\wl@pointB{
695                     \pgfmathresult pt}%
696                 \draw[draw=none, fill=\wl@background@IRcolor] (\wl@pointB,0)
697                     rectangle (\wl@width, \wl@height);%
698                 \ifx\wl@back\wl@visible% 3 visible background
699                     \pgfspectrashade(\wl@begin,780){\wl@visibleshade}%
700                     \fill[shading=\wl@visibleshade] (0,0) rectangle (
701                         \wl@pointB, \wl@height);%
702                 \else% 3
703                     \draw[draw=none, fill=\wl@back] (0,0) rectangle (
704                         \wl@pointB, \wl@height);%
705                 \fi% 3
706             \fi% 2
707         \fi% 1
708     \else% 0
709         \pgfmathparse{int(#1*100)}\edef\wl@bright{\pgfmathresult}%

```

```

706 \pgfmathparse{1.4*\xscale+.09*\linewidth/\wl@width}\edef\wl@linewidth{
707 \pgfmathresult}% NEW v2.0.0 {\xscale} -> {1.4*\xscale+.09*\linewidth/
708 \wl@width}
709 \ifnum\wl@begin>\wl@end% 0
710 \ifnum\wl@end<380\relax% 1
711 \pgfmathparse{(\wl@end-380)*\xscale}\edef\wl@pointA{\pgfmathresult
712 pt}%
713 \ifnum\wl@begin>780\relax% 2
714 \pgfmathparse{(\wl@end-780)*\xscale}\edef\wl@pointB{
715 \pgfmathresult pt}%
716 \draw[draw=none,fill=\wl@background@UVcolor](0,0)rectangle(
717 \wl@pointA,\wl@height);%
718 \draw[draw=none,fill=\wl@background@IRcolor](\wl@pointB,0)
719 rectangle(-\wl@width,\wl@height);%
720 \ifx\wl@back\wl@visible% 3 visible background
721 \foreach\xin{780,...,380}{%
722 \wcolor{\x}%
723 \colorlet{\wcolor}{\wl@temp!\wl@bright!\wl@backvisible}%
724 CHANGED v2.1.0
725 \pgfmathparse{\wl@pointB+(780-\x)*\xscale}\edef
726 \wl@currentx{\pgfmathresult pt}%
727 \draw[\wcolor,line width=@\wl@linewidth](\wl@currentx
728 ,0)---++(0,\wl@height);}%
729 \else% 3
730 \draw[draw=none,fill=\wl@back](\wl@pointA,0)rectangle
731 (\wl@pointB,\wl@height);%
732 \fi% 3
733 \else% 2
734 \pgfmathparse{(\wl@end-\wl@begin)*\xscale}\edef\wl@pointB{
735 \pgfmathresult pt}%
736 \draw[draw=none,fill=\wl@background@UVcolor](0,0)rectangle(
737 \wl@pointA,\wl@height);%
738 \ifx\wl@back\wl@visible% 3 visible background
739 \foreach\xin{\wl@begin,...,380}{%
740 \wcolor{\x}%
741 \colorlet{\wcolor}{\wl@temp!\wl@bright!\wl@backvisible}%
742 CHANGED v2.1.0
743 \pgfmathparse{\wl@pointB+(\wl@begin-\x)*\xscale}\edef
744 \wl@currentx{\pgfmathresult pt}%
745 \draw[\wcolor,line width=@\wl@linewidth](\wl@currentx
746 ,0)---++(0,\wl@height);}%
747 \else% 3
748 \draw[draw=none,fill=\wl@back](\wl@pointA,0)rectangle
749 (\wl@pointB,\wl@height);%
750 \fi% 3
751 \fi% 2
752 \else% 1
753 \ifnum\wl@begin>780\relax% 2
754 \pgfmathparse{(\wl@end-780)*\xscale}\edef\wl@pointB{
755 \pgfmathresult pt}%
756 \draw[draw=none,fill=\wl@background@IRcolor](\wl@pointB,0)
757 rectangle(-\wl@width,\wl@height);%
758 \ifx\wl@back\wl@visible% 3 visible background
759 \foreach\xin{780,...,\wl@end}{%
760 \wcolor{\x}%
761 \colorlet{\wcolor}{\wl@temp!\wl@bright!\wl@backvisible}%
762 CHANGED v2.1.0
763 \pgfmathparse{\wl@pointB+(780-\x)*\xscale}\edef
764 \wl@currentx{\pgfmathresult pt}%
765 \draw[\wcolor,line width=@\wl@linewidth](\wl@currentx
766 ,0)---++(0,\wl@height);}%
767 \else% 3
768 \draw[draw=none,fill=\wl@back](0,0)rectangle(
769 \wl@pointB,\wl@height);%
770 \fi% 3
771 \else% 2
772 \pgfmathparse{(\wl@end-\wl@begin)*\xscale}\edef\wl@pointB{
773 \pgfmathresult pt}%
774 \ifx\wl@back\wl@visible% 3 visible background
775 \foreach\xin{\wl@begin,...,\wl@end}{%

```

```

753         \wcolor{\x}%
754         \colorlet{wcolor}{wl@temp!wl@bright!wl@backvisible}%
755         CHANGED v2.1.0
756         \pgfmathparse{\wl@pointB+(\wl@begin-\x)*\xscale}\edef
757         \wl@currentx{\pgfmathresult pt}%
758         \draw[wcolor,line width=\wl@linewidth] (\wl@currentx
759         ,0) — ++(0,\wl@height);}%
760     \else% 3
761         \draw[draw=none,fill=\wl@back] (0,0) rectangle (
762         \wl@pointB,\wl@height);%
763     \fi% 3
764 \fi% 2
765 \else% 1
766 \ifnum\wl@begin<380\relax% 1
767     \pgfmathparse{(380-\wl@begin)*\xscale}\edef\wl@pointA{\pgfmathresult
768     pt}%
769     \ifnum\wl@end>780\relax% 2
770         \pgfmathparse{(780-\wl@begin)*\xscale}\edef\wl@pointB{
771         \pgfmathresult pt}%
772         \draw[draw=none,fill=\wl@background@UVcolor] (0,0) rectangle (
773         \wl@pointA,\wl@height);%
774         \draw[draw=none,fill=\wl@background@IRcolor] (\wl@pointB,0)
775         rectangle (\wl@width,\wl@height);%
776         \ifx\wl@back\wl@visible% 3 visible background
777             \foreach \x in {380,...,780}{%
778                 \wcolor{\x}%
779                 \colorlet{wcolor}{wl@temp!wl@bright!wl@backvisible}%
780                 CHANGED v2.1.0
781                 \pgfmathparse{\wl@pointB-(780-\x)*\xscale}\edef
782                 \wl@currentx{\pgfmathresult pt}%
783                 \draw[wcolor,line width=\wl@linewidth] (\wl@currentx
784                 ,0) — ++(0,\wl@height);}%
785             \else% 3
786                 \draw[draw=none,fill=\wl@back] (\wl@pointA,0) rectangle
787                 (\wl@pointB,\wl@height);%
788             \fi% 3
789         \else% 2
790             \pgfmathparse{(\wl@end-\wl@begin)*\xscale}\edef\wl@pointB{
791             \pgfmathresult pt}%
792             \draw[draw=none,fill=\wl@background@UVcolor] (0,0) rectangle (
793             \wl@pointA,\wl@height);%
794             \ifx\wl@back\wl@visible% 3 visible background
795                 \foreach \x in {\wl@end,...,380}{%
796                     \wcolor{\x}%
797                     \colorlet{wcolor}{wl@temp!wl@bright!wl@backvisible}%
798                     CHANGED v2.1.0
799                     \pgfmathparse{\wl@pointB-(\wl@end-\x)*\xscale}\edef
800                     \wl@currentx{\pgfmathresult pt}%
801                     \draw[wcolor,line width=\wl@linewidth] (\wl@currentx
802                     ,0) — ++(0,\wl@height);}%
803                 \else% 3
804                     \draw[draw=none,fill=\wl@back] (\wl@pointA,0) rectangle
805                     (\wl@pointB,\wl@height);%
806                 \fi% 3
807             \fi% 2
808         \else% 1
809             \ifnum\wl@end>780\relax% 2
810                 \pgfmathparse{(780-\wl@begin)*\xscale}\edef\wl@pointB{
811                 \pgfmathresult pt}%
812                 \draw[draw=none,fill=\wl@background@IRcolor] (\wl@pointB,0)
813                 rectangle (\wl@width,\wl@height);%
814                 \ifx\wl@back\wl@visible% 3 visible background
815                     \foreach \x in {\wl@begin,...,780}{%
816                         \wcolor{\x}%
817                         \colorlet{wcolor}{wl@temp!wl@bright!wl@backvisible}%
818                         CHANGED v2.1.0
819                         \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{
820                         \pgfmathresult pt}%

```

```

800         \draw[wlcolor,line width=\@wl@linewidth] (\wl@currentx
801         ,0) — ++(0,\wl@height);}%
802     \else% 3
803         \draw[draw=none,fill=\wl@back] (0,0) rectangle (
804         \wl@pointB,\wl@height);%
805     \fi% 3
806 \else% 2
807     \pgfmathparse{(\wl@end-\wl@begin)*\xscale}\edef\wl@pointB{
808     \pgfmathresult pt}%
809     \ifx\wl@back\wl@visible% 3 visible background
810         \foreach \x in {\wl@begin,...,\wl@end}{%
811         \wlcolor{\x}%
812         \colorlet{wlcolor}{wl@temp!\wl@bright!\wl@backvisible}%
813         CHANGED v2.1.0
814         \pgfmathparse{\x-\wl@begin)*\xscale}\edef\wl@currentx{
815         \pgfmathresult pt}%
816         \draw[wlcolor,line width=\@wl@linewidth] (\wl@currentx
817         ,0) — ++(0,\wl@height);}%
818     \else% 3
819         \draw[draw=none,fill=\wl@back] (0,0) rectangle (
820         \wl@pointB,\wl@height);%
821     \fi% 3
822 \fi% 2
823 \fi% 1
824 \fi%0
825 \fi% wl@usevisibleshade
826 }%
827 \def\wl@utils@drawabsorptionlines{\ignorespaces%
828 \ifnum\wl@begin>\wl@end%
829 \ifwl@intensity%
830     \ifwl@redshift\wl@utils@redshift\fi NEW v2.0.0
831     \foreach \x/\y in \wl@list@%
832     {%
833     \pgfmathparse{notless(\x,\wl@end)}\edef\wl@x@nl{\pgfmathresult}%
834     \pgfmathparse{notgreater(\x,\wl@begin)}\edef\wl@x@ng{\pgfmathresult}%
835     \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
836     \pgfmathresult}%
837     \ifnum\wl@plot@point=1%
838     \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult
839     pt}%
840     \pgfmathparse{int(\y*100)}\edef\wl@black{\pgfmathresult}%
841     \wlcolor{\x}%
842     \colorlet{wlcolor}{black!\wl@black!\wl@temp}%
843     \draw[wlcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
844     \wl@height);%
845     \fi%
846     }%
847 \else%
848     \ifwl@redshift\wl@utils@redshift\fi NEW v2.0.0
849     \foreach \x in \wl@list@%
850     {%
851     \pgfmathparse{notless(\x,\wl@end)}\edef\wl@x@nl{\pgfmathresult}%
852     \pgfmathparse{notgreater(\x,\wl@begin)}\edef\wl@x@ng{\pgfmathresult}%
853     \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
854     \pgfmathresult}%
855     \ifnum\wl@plot@point=1%
856     \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult
857     pt}%
858     \wlcolor{\x}%
859     \colorlet{wlcolor}{black!\wl@lineint!\wl@temp}%
860     \draw[wlcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
861     \wl@height);%
862     \fi%
863     }%
864 \fi%
865 \else%
866     \ifwl@intensity%
867     \ifwl@redshift\wl@utils@redshift\fi NEW v2.0.0

```

```

855         \foreach \x/\y in \wl@list@@%
856         {%
857         \pgfmathparse{notless(\x,\wl@begin)}\edef\wl@x@nl{\pgfmathresult}%
858         \pgfmathparse{notgreater(\x,\wl@end)}\edef\wl@x@ng{\pgfmathresult}%
859         \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
            \pgfmathresult}%
860         \ifnum\wl@plot@point=1%
861         \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{
            \pgfmathresult pt}%
862         \pgfmathparse{int(\y*100)}\edef\wl@black{\pgfmathresult}%
863         \wcolor{\x}%
864         \colorlet{wcolor}{black!\wl@black!\wl@temp}%
865         \draw[wcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
            \wl@height);%
866         \fi%
867         }%
868     \else%
869     \ifwl@redshift\wl@utils@redshift\fi% NEW v2.0.0
870     \foreach \x in \wl@list@@%
871     {%
872     \pgfmathparse{notless(\x,\wl@begin)}\edef\wl@x@nl{\pgfmathresult}%
873     \pgfmathparse{notgreater(\x,\wl@end)}\edef\wl@x@ng{\pgfmathresult}%
874     \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
        \pgfmathresult}%
875     \ifnum\wl@plot@point=1%
876     \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{
        \pgfmathresult pt}%
877     \wcolor{\x}%
878     \colorlet{wcolor}{black!\wl@lineint!\wl@temp}%
879     \draw[wcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
        \wl@height);%
880     \fi%
881     }%
882     \fi%
883     \fi%
884     \ifwl@RSvalue% NEW v2.0.0
885     \ifnum\wl@begin>\wl@end%
886     \ifwl@drawaxis\pgfmathparse{-\wl@width-2.5*\wl@axis@height}\edef\wl@redshiftinfo@x{
        \pgfmathresult pt}%
887     \else\pgfmathparse{-\wl@width}\edef\wl@redshiftinfo@x{\pgfmathresult pt}\fi%
888     \else%
889     \ifwl@drawaxis\pgfmathparse{-2.5*\wl@axis@height}\edef\wl@redshiftinfo@x{\pgfmathresult pt}
        %
890     \else\edef\wl@redshiftinfo@x{0pt}\fi%
891     \fi%
892     \ifwl@drawaxis\pgfmathparse{-.75*\wl@axis@height-1.3*\ht0-2pt}\edef\wl@redshiftinfo@y{
        \pgfmathresult pt}\else\edef\wl@redshiftinfo@y{0pt}\fi%
893     \node[below right,inner xsep=0pt,font=\wl@axisfont] at (\wl@redshiftinfo@x,
        \wl@redshiftinfo@y) {\wl@redshiftinfo};%
894     \fi% NEW v2.0.0
895     }%
896     \def\wl@utils@drawemissionlines{\ignorespaces%
897     \ifnum\wl@begin>\wl@end%
898     \ifwl@intensity%
899     \ifwl@redshift\wl@utils@redshift\fi% NEW v2.0.0
900     \foreach \x/\y in \wl@list@@%
901     {%
902     \wcolor{\x}%
903     \pgfmathparse{notless(\x,\wl@end)}\edef\wl@x@nl{\pgfmathresult}%
904     \pgfmathparse{notgreater(\x,\wl@begin)}\edef\wl@x@ng{\pgfmathresult}
        %
905     \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
        \pgfmathresult}%
906     \ifnum\wl@plot@point=1%
907     \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult
        pt}%
908     \pgfmathparse{int(\y*100)}\edef\wl@black{\pgfmathresult}%
909     \colorlet{wcolor}{\wl@temp!\wl@black!black}%
910     \draw[wcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
        \wl@height);%

```

```

911         \fi%
912     }%
913 \else%
914     \ifwl@redshift\wl@utils@redshift\fi% NEW v2.0.0
915     \foreach \x in \wl@list@@%
916     {%
917         \wcolor{\x}%
918         \pgfmathparse{notless(\x,\wl@end)}\edef\wl@x@nl{\pgfmathresult}%
919         \pgfmathparse{notgreater(\x,\wl@begin)}\edef\wl@x@ng{\pgfmathresult}%
920
921         \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
922             \pgfmathresult}%
923         \ifnum\wl@plot@point=1%
924         \pgfmathparse{((\wl@end-\x)*\xscale)}\edef\wl@currentx{\pgfmathresult
925             pt}%
926         \colorlet{wcolor}{wl@temp!\wl@lineint!black}%
927         \draw[wcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
928             \wl@height);%
929         \fi%
930     }%
931 \fi%
932 \else%
933 \ifwl@intensity%
934     \ifwl@redshift\wl@utils@redshift\fi% NEW v2.0.0
935     \foreach \x/\y in \wl@list@@%
936     {%
937         \wcolor{\x}%
938         \pgfmathparse{notless(\x,\wl@begin)}\edef\wl@x@nl{\pgfmathresult}%
939         \pgfmathparse{notgreater(\x,\wl@end)}\edef\wl@x@ng{\pgfmathresult}%
940         \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
941             \pgfmathresult}%
942         \ifnum\wl@plot@point=1%
943         \pgfmathparse{((\x-\wl@begin)*\xscale)}\edef\wl@currentx{
944             \pgfmathresult pt}%
945         \pgfmathparse{int(\y*100)}\edef\wl@black{\pgfmathresult}%
946         \colorlet{wcolor}{wl@temp!\wl@black!black}%
947         \draw[wcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
948             \wl@height);%
949         \fi%
950     }%
951 \else%
952     \ifwl@redshift\wl@utils@redshift\fi% NEW v2.0.0
953     \foreach \x in \wl@list@@%
954     {%
955         \wcolor{\x}%
956         \pgfmathparse{notless(\x,\wl@begin)}\edef\wl@x@nl{\pgfmathresult}%
957         \pgfmathparse{notgreater(\x,\wl@end)}\edef\wl@x@ng{\pgfmathresult}%
958         \pgfmathparse{and(\wl@x@nl,\wl@x@ng)}\edef\wl@plot@point{
959             \pgfmathresult}%
960         \ifnum\wl@plot@point=1%
961         \pgfmathparse{((\x-\wl@begin)*\xscale)}\edef\wl@currentx{
962             \pgfmathresult pt}%
963         \colorlet{wcolor}{wl@temp!\wl@lineint!black}%
964         \draw[wcolor,line width=\wl@linewidth] (\wl@currentx,0) — ++(0,
965             \wl@height);%
966         \fi%
967     }%
968 \fi%
969 \fi%
970 \ifwl@RSvalue% NEW v2.0.0
971 \ifnum\wl@begin>\wl@end%
972 \ifwl@drawaxis\pgfmathparse{-\wl@width-2.5*\wl@axis@height}\edef\wl@redshiftinfo@x{
973     \pgfmathresult pt}%
974 \else\pgfmathparse{-\wl@width}\edef\wl@redshiftinfo@x{\pgfmathresult pt}\fi%
975 \else%
976 \ifwl@drawaxis\pgfmathparse{-2.5*\wl@axis@height}\edef\wl@redshiftinfo@x{\pgfmathresult pt}
977 %
978 \else\edef\wl@redshiftinfo@x{0pt}\fi%
979 \fi%

```

```

968 \ifwl@drawaxis\pgfmathparse{-.75*\wl@axis@height-1.3*\ht0-2pt}\edef\wl@redshiftinfo@y{
    \pgfmathresult pt}\else\edef\wl@redshiftinfo@y{0pt}\fi%
969 \node[below right,inner xsep=0pt,font=\wl@axisfont] at (\wl@redshiftinfo@x,
    \wl@redshiftinfo@y) {\wl@redshiftinfo};%
970 \fi% NEW v2.0.0
971 }%
972 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
973 % return: integer with position (e.g. '0' for west, ... )
974 \def\wl@get@label@position{\ignorespaces%
975 \wl@countc=0%
976 \@for\@mylabel:=\wl@label@position@list%
977     \do{%
978     \ifx\@mylabel\wl@labelposition\edef\wl@label@position{\the\wl@countc}\fi%
979     \advance\wl@countc by1%
980     }%
981 }%
982 % NEW v2.0.0 -----
983 % redshift:  $\lambda(\text{obs})=\lambda(\text{emit}) * \{(1+v/c * \cos(\theta)) / \sqrt{1-v^2/c^2}\}$  -----> Relativistic
    Doppler
984 %  $v_{\text{bar}} \rightarrow v/c \rightarrow$  normalized velocity of the source (e.g. '0.9' for  $v=0.9c$ )
985 %  $\theta \rightarrow$  angle between the direction of relative motion of the source and the direction of
    emission in the observer's frame (zero angle is directly away from the observer)
986 \def\wl@processredshiftkey#1{\ignorespaces%
987 \wl@redshiftfalse%
988 \edef\wl@redshiftkey@expand{#1}% to expand a value passed by a macro (not necessary if the
    user provided a number...)
989 \expandafter\wl@redshiftkey@firstchar\wl@redshiftkey@expand\relax%
990 \ifcat1\wl@@RedShift\relax%
991 \pgfmathparse{1+#1}\relax%
992 \edef\wl@UMMAISZ{\pgfmathresult}\wl@redshifttrue%
993 \edef\wl@redshiftinfo{redshift z=#1}%
994 \else%
995 \edef\wl@redshiftkey@expand{#1}%
996 \expandafter\wl@process@redshift\wl@redshiftkey@expand\relax%
997 \fi%
998 }%
999 \def\wl@redshiftkey@firstchar#1#2\relax{\edef\wl@@RedShift{#1}}%
1000 \def\wl@process@redshift#1=#2/#3\relax{%
1001 \edef\wl@redshiffttest{#1}%
1002 \ifx\wl@redshiffttest\wl@redshift@D\relax%
1003 \pgfmathparse{(1+#2*cos(#3))/sqrt(1-#2*#2)}%
1004 \edef\wl@UMMAISZ{\pgfmathresult}\wl@redshifttrue%
1005 \pgfmathparse{#2*cos(#3))/sqrt(1-#2*#2)}%
1006 \edef\wl@redshiftinfo{Relativistic Doppler redshift z=\pgfmathresult\ (\mbox{v\hskip.1ex=
    \hskip.1ex#2\hskip.1exc\hskip.5ex;\hskip.5ex\ensuremath{\theta}\hskip.1ex=\hskip.1ex#3
    \ensuremath{\wedge\circ}})}%
1007 \fi%
1008 }%
1009 % \wl@utils@redshift -----
1010 % returns the wllist with the shift computed
1011 \def\wl@utils@redshift{\ignorespaces%
1012 \let\wt@backlist@@\wl@list@@\relax%
1013 \let\wl@list@@\empty\relax%
1014 \wl@firsttrue\relax%
1015 \ifwl@intensity% list (lambda/intensity)
1016 \foreach \x/\y in \wt@backlist@@{%
1017 \pgfmathparse{\x*\wl@UMMAISZ}\edef\@currentline@wl{\pgfmathresult}%
1018 \ifwl@first\global\wl@addt@list{\wl@list@@}{\@currentline@wl/\y}\else%
1019 \global\wl@addt@list{\wl@list@@}{,\@currentline@wl/\y}\fi%
1020 \ifwl@first\global\wl@firstfalse\fi%
1021 }%
1022 \else% list (lambda)
1023 \foreach \x in \wt@backlist@@{%
1024 \pgfmathparse{\x*\wl@UMMAISZ}\edef\@currentline@wl{\pgfmathresult}%
1025 \ifwl@first\global\wl@addt@list{\@currentline@wl}\else%
1026 \global\wl@addt@list{\wl@list@@}{,\@currentline@wl}\fi%
1027 \ifwl@first\global\wl@firstfalse\fi%
1028 }%
1029 \fi%
1030 }%

```

```

1031 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1032 % NEW v2.1.0 ----->
1033 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1034 % \pgfspectrashade[type=<h|v>](start,end){name} ----- type: h–horizontal (default) | v–
      vertical
1035 \def\pgfspectrashade{\ignorespaces \@ifnextchar[\wl@pgfspectrashade{\wl@pgfspectrashade[h]}%
1036 \def\wl@pgfspectrashade[#1](#2,#3)#4{\ignorespaces%
1037 \ifnum#2>#3\relax\PackageError{pgf-spectra}%
1038 {in \textbackslash pgfspectrashade the starting wavelength (#2) must be lesser then the
      finishing wavelength (#3)}%
1039 {Try typing the wavelengths in the correct order: \textbackslash pgfspectrashade(#3,#2)...}
      \else%
1040 \ifnum#3<381\relax\PackageError{pgf-spectra}{in \textbackslash pgfspectrashade the finishing
      wavelength (#3) must be at least 381...}%
1041 {Please type a finishing wavelength greater then 380...}\else%
1042 \ifnum#2<380\relax\def\wl@shade@begin{380}\else\pgfmathparse{int(#2)}\edef\wl@shade@begin{
      \pgfmathresult}\fi%
1043 \ifnum#3>780\relax\def\wl@shade@end{780}\else\pgfmathparse{int(#3)}\edef\wl@shade@end{
      \pgfmathresult}\fi%
1044 \pgfmathparse{((\wl@shade@end-\wl@shade@begin)/50)}\edef\wl@shadecolor@step{\pgfmathresult}%
1045 %\foreach \n in {1,...,51}{%
1046 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1047           21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,%
1048           39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1049 \pgfmathparse{\wl@shade@begin+(\n-1)*\wl@shadecolor@step}\edef\wl@currentwl{\pgfmathresult}%
1050 \wlcolor{\wl@currentwl}%
1051 \edef\wl@colname{wlshcol\@Roman\n}\relax\colorlet{\wl@colname}{\wlcolor}%
1052 }%
1053 \def\wl@test{#1}\def\wl@v{v}\def\wl@h{h}%
1054 \ifx\wl@test\wl@h\relax%
1055 % the horizontal pgfshading
1056 \pgfdeclarehorizontalshading{#4}{100bp}{color(0bp)=(wlshcolI); color(25bp)=(wlshcolI); color
      (26bp)=(wlshcolII); color(27bp)=(wlshcolIII); color(28bp)=(wlshcolIV); color(29bp)=(
      wlshcolV); color(30bp)=(wlshcolVI); color(31bp)=(wlshcolVII); color(32bp)=(wlshcolVIII);
      color(33bp)=(wlshcolIX); color(34bp)=(wlshcolX); color(35bp)=(wlshcolXI); color(36bp)=(
      wlshcolXII); color(37bp)=(wlshcolXIII); color(38bp)=(wlshcolXIV); color(39bp)=(wlshcolXV);
      color(40bp)=(wlshcolXVI); color(41bp)=(wlshcolXVII); color(42bp)=(wlshcolXVIII); color(43
      bp)=(wlshcolXIX); color(44bp)=(wlshcolXX); color(45bp)=(wlshcolXXI); color(46bp)=(
      wlshcolXXII); color(47bp)=(wlshcolXXIII); color(48bp)=(wlshcolXXIV); color(49bp)=(
      wlshcolXXV); color(50bp)=(wlshcolXXVI); color(51bp)=(wlshcolXXVII); color(52bp)=(
      wlshcolXXVIII); color(53bp)=(wlshcolXXIX); color(54bp)=(wlshcolXXX); color(55bp)=(
      wlshcolXXXI); color(56bp)=(wlshcolXXXII); color(57bp)=(wlshcolXXXIII); color(58bp)=(
      wlshcolXXXIV); color(59bp)=(wlshcolXXXV); color(60bp)=(wlshcolXXXVI); color(61bp)=(
      wlshcolXXXVII); color(62bp)=(wlshcolXXXVIII); color(63bp)=(wlshcolXXXIX); color(64bp)=(
      wlshcolXL); color(65bp)=(wlshcolXLI); color(66bp)=(wlshcolXLII); color(67bp)=(wlshcolXLIII
      ); color(68bp)=(wlshcolXLIV); color(69bp)=(wlshcolXLV); color(70bp)=(wlshcolXLVI); color
      (71bp)=(wlshcolXLVII); color(72bp)=(wlshcolXLVIII); color(73bp)=(wlshcolXLIX); color(74bp)
      =(wlshcolL); color(75bp)=(wlshcolLI); color(100bp)=(wlshcolLI)}%
1057 \else\ifx\wl@test\wl@v\relax%
1058 % the vertical pgfshading
1059 \pgfdeclareverticalshading{#4}{100bp}{color(0bp)=(wlshcolI); color(25bp)=(wlshcolI); color
      (26bp)=(wlshcolII); color(27bp)=(wlshcolIII); color(28bp)=(wlshcolIV); color(29bp)=(
      wlshcolV); color(30bp)=(wlshcolVI); color(31bp)=(wlshcolVII); color(32bp)=(wlshcolVIII);
      color(33bp)=(wlshcolIX); color(34bp)=(wlshcolX); color(35bp)=(wlshcolXI); color(36bp)=(
      wlshcolXII); color(37bp)=(wlshcolXIII); color(38bp)=(wlshcolXIV); color(39bp)=(wlshcolXV);
      color(40bp)=(wlshcolXVI); color(41bp)=(wlshcolXVII); color(42bp)=(wlshcolXVIII); color(43
      bp)=(wlshcolXIX); color(44bp)=(wlshcolXX); color(45bp)=(wlshcolXXI); color(46bp)=(
      wlshcolXXII); color(47bp)=(wlshcolXXIII); color(48bp)=(wlshcolXXIV); color(49bp)=(
      wlshcolXXV); color(50bp)=(wlshcolXXVI); color(51bp)=(wlshcolXXVII); color(52bp)=(
      wlshcolXXVIII); color(53bp)=(wlshcolXXIX); color(54bp)=(wlshcolXXX); color(55bp)=(
      wlshcolXXXI); color(56bp)=(wlshcolXXXII); color(57bp)=(wlshcolXXXIII); color(58bp)=(
      wlshcolXXXIV); color(59bp)=(wlshcolXXXV); color(60bp)=(wlshcolXXXVI); color(61bp)=(
      wlshcolXXXVII); color(62bp)=(wlshcolXXXVIII); color(63bp)=(wlshcolXXXIX); color(64bp)=(
      wlshcolXL); color(65bp)=(wlshcolXLI); color(66bp)=(wlshcolXLII); color(67bp)=(wlshcolXLIII
      ); color(68bp)=(wlshcolXLIV); color(69bp)=(wlshcolXLV); color(70bp)=(wlshcolXLVI); color
      (71bp)=(wlshcolXLVII); color(72bp)=(wlshcolXLVIII); color(73bp)=(wlshcolXLIX); color(74bp)
      =(wlshcolL); color(75bp)=(wlshcolLI); color(100bp)=(wlshcolLI)}%
1060 \fi\fi%
1061 \fi\fi%
1062 }%

```

```

1063 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1064 \newif\ifwl@logshade% NEW v2.1.1
1065 \pgfkeys{/wl/.cd,%
1066 shade begin/.get=\wl@shadebegin,% NEW v2.1.1
1067 shade begin/.store in=\wl@shadebegin,% NEW v2.1.1
1068 shade begin/.default=380,% NEW v2.1.1
1069 shade end/.get=\wl@shadeend,%
1070 shade end/.store in=\wl@shadeend,%
1071 shade end/.default=780,%
1072 shade opacity/.get=\wl@shade@opacity,%
1073 shade opacity/.store in=\wl@shade@opacity,%
1074 shade opacity/.default=1,%
1075 shade opacity color/.get=\wl@shade@opacitycolor,%
1076 shade opacity color/.store in=\wl@shade@opacitycolor,%
1077 shade opacity color/.default=white,%
1078 logarithmic/.is if=wl@logshade% NEW v2.1.1
1079 }%
1080 \pgfkeys{/wl/.cd,shade begin,shade end,shade opacity,shade opacity color,logarithmic=false}%
1081 % \pgfspectraplotshade{name} or
1082 % \pgfspectraplotshade[options]{name}
1083 % — recoded in v2.1.1 —
1084 \def\pgfspectraplotshade{\@ifnextchar[\wl@pgfspectraplotshade{\wl@pgfspectraplotshade []}}%
1085 \def\wl@pgfspectraplotshade[#1]#2{\ignorespaces%
1086 \pgfkeys{/wl/.cd,shade begin=380,shade end=780,shade opacity=1,shade opacity color=white,
1087 logarithmic=false}%
1088 \pgfkeys{/wl/.cd,UVcolor=wUVcolor,IRcolor=wIRcolor,gamma=.8}%
1089 \pgfkeys{/wl/.cd,#1}%
1090 \pgfmathparse{100*\wl@shade@opacity}\edef\@wl@shade@opacity{\pgfmathresult}%
1091 %
1092 % verifying start & end
1093 \edef\wl@shadebegin@pt{\wl@shadebegin pt}%
1094 \edef\wl@shadeend@pt{\wl@shadeend pt}%
1095 \ifdim\wl@shadebegin@pt<0pt\relax%
1096 \PackageError{pgf-spectra}{pgfspectraplotshade – minimum wavelength is 0nm!}{}%
1097 \else\ifdim\wl@shadeend@pt>16000pt\relax%
1098 \PackageError{pgf-spectra}{pgfspectraplotshade – maximum wavelength is 16000nm!}{}%
1099 \else\ifdim\wl@shadeend@pt<\wl@shadebegin@pt\relax%
1100 \PackageError{pgf-spectra}{pgfspectraplotshade – shade end must be greater then shade begin
1101 !}{}%
1102 \else\relax%
1103 \ifdim\wl@shadeend@pt<380pt\relax%
1104 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!\@wl@shade@opacity!
1105 \wl@shade@opacitycolor);color(100bp)=(\wl@UVcolor!\@wl@shade@opacity!
1106 \wl@shade@opacitycolor)}%
1107 \else%
1108 \ifdim\wl@shadebegin@pt>780pt\relax%
1109 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@IRcolor!\@wl@shade@opacity!
1110 \wl@shade@opacitycolor);color(100bp)=(\wl@IRcolor!\@wl@shade@opacity!
1111 \wl@shade@opacitycolor)}%
1112 \else%
1113 \pgfkeys{/pgf/number format/.cd,fixed,precision=4,set thousands separator={},assume math
1114 mode=true}%
1115 %
1116 \ifdim\wl@shadebegin@pt<380pt\relax%
1117 \ifdim\wl@shadeend@pt>780pt\relax%
1118 %
1119 % —————
1120 % \wl@shadebegin<380nm & \wl@shadeend>780nm
1121 % —————
1122 % Shade structure :
1123 % xShade(bp) = f(wl)
1124 % x_from_\wl@shadebegin[bp]@25bp — x_to_\wl@shadeend[bp]@75bp :
1125 % x_UV_from_\wl@shadebegin[bp]@25bp — xUV_to_379.9nm@x_end_UV[bp] —
1126 % — x_VIS_from_380nm@x_start_VIS[bp] — xVIS_to_780nm@x_end_VIS[bp]
1127 % —————
1128 % — x_IR_from_780.1nm@x_start_IR[bp] — — x\wl@shadeend[bp]@75bp

```

```

1119 %
1120 % if wl@shadebegin=0nm let \wl@shadebegin=0.0001 (log only)
1121 % (approximation for supporting shading from 0... not a real situation in a
1122 % logarithmic plot,
1123 % but not detected in the resulting shade if used for other purposes... )
%

1124 \ifwl@logshade% \wl@logshadetrue
1125 \ifnum\wl@shadebegin=0\relax\edef\wl@shadebegin{0.00001}\fi%
1126 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1127 \pgfmathprintnumber to {\pgfmathresult}{\wl@shadestep}%
1128 \pgfmathparse{50/(\log 10(\wl@shadeend)-\log 10(\wl@shadebegin))}%
1129 \pgfmathprintnumber to {\pgfmathresult}{\wl@shade@logscale@factor}%
1130 \pgfmathparse{25+(\log 10(379.9)-\log 10(\wl@shadebegin))*
1131 \wl@shade@logscale@factor}%
1132 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1133 \edef\wl@X@UV{\wl@X@tmp bp}%
1134 \pgfmathparse{25+(\log 10(780.1)-\log 10(\wl@shadebegin))*
1135 \wl@shade@logscale@factor}%
1136 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1137 \edef\wl@X@IR{\wl@X@tmp bp}%
1138 \pgfmathparse{25+(\log 10(380)-\log 10(\wl@shadebegin))*
1139 \wl@shade@logscale@factor}%
1140 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1141 \edef\wl@x@start@VIS@log{\wl@X@tmp}%
1142 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1143 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1144 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1145 \pgfmathparse{372+8*\n}%
1146 \pgfmathprintnumber to {\pgfmathresult}{\wl@currentwl}%
1147 \colorlet{\wl@currentwl}%
1148 \colorlet{\wl@temp!}{\wl@temp!@\wl@shade@opacity!
1149 \wl@shade@opacitycolor}%
1150 \edef\wl@colorname{\wlshcol@\Roman\n}\relax\colorlet{
1151 \wl@colorname}{\wl@currentwl}%
1152 \pgfmathparse{\wl@x@start@VIS@log+(\log 10(372+8*\n)-\log 10(380))*
1153 \wl@shade@logscale@factor}%
1154 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1155 \expandafter\edef\csname x@\Roman\n\endcsname{\wl@X@tmp bp}%
1156 }%
1157 \else% \wl@logshadefalse
1158 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1159 \edef\wl@shade@scale{\pgfmathresult}%
1160 \pgfmathparse{25+(379.9-\wl@shadebegin)/\wl@shade@scale}%
1161 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1162 \edef\wl@X@UV{\wl@X@tmp bp}%
1163 \pgfmathparse{25+(780.1-\wl@shadebegin)/\wl@shade@scale}%
1164 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1165 \edef\wl@X@IR{\wl@X@tmp bp}%
1166 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1167 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1168 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1169 \pgfmathparse{372+8*\n}% MANTER !!!
1170 \pgfmathprintnumber to {\pgfmathresult}{\wl@currentwl}%
1171 \colorlet{\wl@currentwl}%
1172 \colorlet{\wl@temp!}{\wl@temp!@\wl@shade@opacity!
1173 \wl@shade@opacitycolor}%
1174 \edef\wl@colorname{\wlshcol@\Roman\n}\relax\colorlet{
1175 \wl@colorname}{\wl@currentwl}%
1176 \pgfmathparse{25+(372-\wl@shadebegin+8*\n)/\wl@shade@scale}%
1177 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1178 \expandafter\edef\csname x@\Roman\n\endcsname{\wl@X@tmp bp}%
1179 }%
1180 \fi% \ifwl@logshade
1181 % the horizontal shading
1182 \pgfdeclarehorizontalshading[#2]{100bp}{color(0bp)=(\wl@UVcolor!
1183 \wl@shade@opacity!\wl@shade@opacitycolor);color(\wl@X@UV)=(\wl@UVcolor!

```

```

1175 \w@shade@opacity!\w@shade@opacitycolor);%
color(\xI)=(w\shcolI);color(\xII)=(w\shcolII);color(\xIII)=(w\shcolIII);color(
\xIV)=(w\shcolIV);color(\xV)=(w\shcolV);color(\xVI)=(w\shcolVI);color(\xVII)=(
w\shcolVII);color(\xVIII)=(w\shcolVIII);color(\xIX)=(w\shcolIX);color(\xX)=(
w\shcolX);color(\xXI)=(w\shcolXI);color(\xXII)=(w\shcolXII);color(\xXIII)=(
w\shcolXIII);color(\xXIV)=(w\shcolXIV);color(\xXV)=(w\shcolXV);color(\xXVI)=(
w\shcolXVI);color(\xXVII)=(w\shcolXVII);color(\xXVIII)=(w\shcolXVIII);color(
\xXIX)=(w\shcolXIX);color(\xXX)=(w\shcolXX);color(\xXXI)=(w\shcolXXI);color(
\xXXII)=(w\shcolXXII);color(\xXXIII)=(w\shcolXXIII);color(\xXXIV)=(w\shcolXXIV
);color(\xXXV)=(w\shcolXXV);color(\xXXVI)=(w\shcolXXVI);color(\xXXVII)=(
w\shcolXXVII);color(\xXXVIII)=(w\shcolXXVIII);color(\xXXIX)=(w\shcolXXIX);
color(\xXXX)=(w\shcolXXX);color(\xXXXI)=(w\shcolXXXI);color(\xXXXII)=(
w\shcolXXXII);color(\xXXXIII)=(w\shcolXXXIII);color(\xXXXIV)=(w\shcolXXXIV);
color(\xXXXV)=(w\shcolXXXV);color(\xXXXVI)=(w\shcolXXXVI);color(\xXXXVII)=(
w\shcolXXXVII);color(\xXXXVIII)=(w\shcolXXXVIII);color(\xXXXIX)=(w\shcolXXXIX
);color(\xXL)=(w\shcolXL);color(\xXLI)=(w\shcolXLI);color(\xXLII)=(w\shcolXLII
);color(\xXLIII)=(w\shcolXLIII);color(\xXLIV)=(w\shcolXLIV);color(\xXLV)=(
w\shcolXLV);color(\xXLVI)=(w\shcolXLVI);color(\xXLVII)=(w\shcolXLVII);color(
\xXLVIII)=(w\shcolXLVIII);color(\xXLIX)=(w\shcolXLIX);color(\xL)=(w\shcolL);
color(\xLI)=(w\shcolLI);%
1176 color(\w@X@IR)=(\w@IRcolor!\w@shade@opacity!\w@shade@opacitycolor);color
(100bp)=(\w@IRcolor!\w@shade@opacity!\w@shade@opacitycolor);%
1177 \else% \w@shadeend@pt>780pt
1178 %


---


1179 % \w@shadebegin<380nm & \w@shadeend<=780nm


---


1180 % Shade structure:
1181 % 25bp — UV — \w@X@UV — \w@X@VIS@i — xVIS_to_\w@shadeend[bp]@75bp
1182 % VIS region in shade (51 points) → \n=1,2,...,51
1183 %


---


1184 \pgfmathparse{int(\w@shadeend-380)}\pgfmathprintnumberto{\pgfmathresult}{
\w@shade@VIS@amp@int}%
1185 \ifw@logshade% \w@logshadetrue
1186 \ifnum\w@shadebegin=0\relax\edef\w@shadebegin{0.00001}\fi%
1187 \ifnum\w@shade@VIS@amp@int>20\relax%
1188 \pgfmathparse{50/(\log10(\w@shadeend)-\log10(\w@shadebegin))}%
1189 \pgfmathprintnumberto{\pgfmathresult}{\w@shade@logscale@factor}%
1190 \pgfmathparse{25+(\log10(379.9)-\log10(\w@shadebegin))*
\w@shade@logscale@factor}%
1191 \pgfmathprintnumberto{\pgfmathresult}{\w@X@tmp}%
1192 \edef\w@X@UV{\w@X@tmp bp}%
1193 \pgfmathparse{25+(\log10(380)-\log10(\w@shadebegin))*
\w@shade@logscale@factor}%
1194 \pgfmathprintnumberto{\pgfmathresult}{\w@X@tmp}%
1195 \edef\w@X@VIS{\w@X@tmp bp}%
1196 \pgfmathparse{((\w@shadeend-380)/50)%}
1197 \pgfmathprintnumberto{\pgfmathresult}{\w@X@tmp}%
1198 \edef\w@step@VIS{\w@X@tmp}%
1199 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1200 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1201 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1202 \pgfmathparse{380+\w@step@VIS*(\n-1)}%
1203 \pgfmathprintnumberto{\pgfmathresult}{\w@currentwl}%
1204 \w\color{\w@currentwl}%
1205 \colorlet{\w\color}{\w@temp!\w@shade@opacity!
\w@shade@opacitycolor}%
1206 \edef\w@colorname{\w\shcol@\Roman\n}\relax\colorlet{
\w@colorname}{\w\color}%
1207 \pgfmathparse{\w@X@VIS+\log10(1+(\n-1)*(\w@step@VIS/380))*
\w@shade@logscale@factor}% correction factor for log10 calc
1208 \pgfmathprintnumberto{\pgfmathresult}{\w@X@tmp}%
1209 \pgfmathparse{\w@X@tmp*0.99638}% correction factor for log10
calc
1210 \pgfmathprintnumberto{\pgfmathresult}{\w@X@tmp}%
1211 \expandafter\edef\csname x@\Roman\n\endcsname{\w@X@tmp bp}%
1212 }%

```

```

1213 \edef\xLI{75bp}% not always at 75bp... so forcing it!
1214 % the horizontal shading
1215 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
1216 \wl@shade@opacity!\wl@shade@opacitycolor);color(\wl@X@UV)=(
1217 \wl@UVcolor!\wl@shade@opacity!\wl@shade@opacitycolor);%
1218 color(\xI)=(wshcolI);color(\xII)=(wshcolII);color(\xIII)=(wshcolIII);
1219 color(\xIV)=(wshcolIV);color(\xV)=(wshcolV);color(\xVI)=(wshcolVI);
1220 color(\xVII)=(wshcolVII);color(\xVIII)=(wshcolVIII);color(\xIX)=(
1221 wshcolIX);color(\xX)=(wshcolX);color(\xXI)=(wshcolXI);color(\xXII)
1222 =(wshcolXII);color(\xXIII)=(wshcolXIII);color(\xXIV)=(wshcolXIV);
1223 color(\xXV)=(wshcolXV);color(\xXVI)=(wshcolXVI);color(\xXVII)=(
1224 wshcolXVII);color(\xXVIII)=(wshcolXVIII);color(\xXIX)=(wshcolXIX);
1225 color(\xXX)=(wshcolXX);color(\xXXI)=(wshcolXXI);color(\xXXII)=(
1226 wshcolXXII);color(\xXXIII)=(wshcolXXIII);color(\xXXIV)=(wshcolXXIV)
1227 ;color(\xXXV)=(wshcolXXV);color(\xXXVI)=(wshcolXXVI);color(\xXXVII)
1228 =(wshcolXXVII);color(\xXXVIII)=(wshcolXXVIII);color(\xXXIX)=(
1229 wshcolXXIX);color(\xXXX)=(wshcolXXX);color(\xXXXI)=(wshcolXXXI);
1230 color(\xXXXII)=(wshcolXXXII);color(\xXXXIII)=(wshcolXXXIII);color(
1231 \xXXXIV)=(wshcolXXXIV);color(\xXXXV)=(wshcolXXXV);color(\xXXXVI)=(
1232 wshcolXXXVI);color(\xXXXVII)=(wshcolXXXVII);color(\xXXXVIII)=(
1233 wshcolXXXVIII);color(\xXXXIX)=(wshcolXXXIX);color(\xXL)=(wshcolXL);
1234 color(\xXLI)=(wshcolXLI);color(\xXLII)=(wshcolXLII);color(\xXLIII)=(
1235 wshcolXLIII);color(\xXLIV)=(wshcolXLIV);color(\xXLV)=(wshcolXLV);
1236 color(\xXLVI)=(wshcolXLVI);color(\xXLVII)=(wshcolXLVII);color(
1237 \xXLVIII)=(wshcolXLVIII);color(\xXLIX)=(wshcolXLIX);color(\xL)=(
1238 wshcolL);color(\xLI)=(wshcolLI);%
1239 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
1240 color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);%
1241 \else% VIS@amp@int<=20%
1242 \pgfmathparse{1/(log10(\wl@shadeend)-log10(\wl@shadebegin))}%
1243 \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1244 \pgfmathparse{25+(log10(379.999)-log10(\wl@shadebegin))*50*
1245 \wl@shade@logscale@factor}%
1246 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1247 \ifnum\wl@shade@VIS@amp@int=0\relax%
1248 \pgfmathparse{\wl@X@tmp*0.99621-.1898}% correction factor for log10
1249 calc
1250 \else%
1251 \pgfmathparse{\wl@X@tmp*0.99621}% correction factor for log10 calc
1252 \fi%
1253 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1254 \edef\wl@X@UV{\wl@X@tmp bp}%
1255 \pgfmathparse{25+(log10(380)-log10(\wl@shadebegin))*50*
1256 \wl@shade@logscale@factor}%
1257 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1258 \ifnum\wl@shade@VIS@amp@int=0\relax%
1259 \pgfmathparse{\wl@X@tmp*0.99621-.1898}% correction factor for log10
1260 calc
1261 \else%
1262 \pgfmathparse{\wl@X@tmp*0.99621}% correction factor for log10 calc
1263 \fi%
1264 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1265 \edef\wl@X@VIS{\wl@X@tmp bp}%
1266 \pgfmathparse{(\wl@shadeend-380)/50}%
1267 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1268 \edef\wl@step@VIS{\wl@X@tmp}%
1269 \@for\n:=1,6,11,16,21,26,31,36,41,46,51\do{%
1270 \pgfmathparse{380+\wl@step@VIS*(\n-1)}%
1271 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1272 \wcolor{\wl@currentwl}%
1273 \colorlet{wcolor}{\wl@temp!\wl@shade@opacity!
1274 \wl@shade@opacitycolor}%
1275 \edef\wl@colorname{wshcol\@Roman\n}\relax\colorlet{
1276 \wl@colorname}{wcolor}%
1277 \pgfmathparse{\wl@X@VIS+log10(1+(\n-1)*(\wl@step@VIS/380))*50*
1278 \wl@shade@logscale@factor}%
1279 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1280 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1281 }%
1282 \edef\xLI{75bp}%

```

```

1253 % the horizontal shading
1254 \pgfdeclarehorizontalshading{#2}{100bp}{ color(0bp)=(\wl@UVcolor!
\wl@shade@opacity!\wl@shade@opacitycolor) ; color(\wl@X@UV)=(
\wl@UVcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;%
1255 color(\xl)=(w!shcolI) ; color(\xVI)=(w!shcolVI) ; color(\xXI)=(w!shcolXI) ;
color(\xXVI)=(w!shcolXVI) ; color(\xXXI)=(w!shcolXXI) ; color(\xXXVI)=(
w!shcolXXVI) ; color(\xXXI)=(w!shcolXXI) ; color(\xXXVI)=(w!shcolXXVI)
; color(\xXLI)=(w!shcolXLI) ; color(\xXLVI)=(w!shcolXLVI) ; color(\xLI)=(
w!shcolLI) ;%
1256 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;
color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;%
1257 \fi%
1258 \else% \wl@logshadefalse
1259 \ifnum\wl@shade@VIS@amp@int=0\relax%
1260 \w!color{380}%
1261 \colorlet{w!color}{\wl@temp!\wl@shade@opacity!\wl@shade@opacitycolor}%
1262 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1263 \edef\wl@shade@scale{\pgfmathresult}%
1264 \pgfmathparse{25+(379.9-\wl@shadebegin)/\wl@shade@scale-.1898}%
1265 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@UV}%
1266 \pgfmathparse{25+(379.99-\wl@shadebegin)/\wl@shade@scale-.1898}%
1267 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@VIS}%
1268 % the horizontal shading
1269 \pgfdeclarehorizontalshading{#2}{100bp}{ color(0bp)=(\wl@UVcolor!
\wl@shade@opacity!\wl@shade@opacitycolor) ; color(\wl@X@UV)=(
\wl@UVcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;%
1270 color(\wl@X@VIS)=(w!color) ; color(75bp)=(w!color) ;%
1271 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;
color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;%
1272 \else%
1273 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1274 \edef\wl@shade@scale{\pgfmathresult}%
1275 \pgfmathparse{25+(379.99-\wl@shadebegin)/\wl@shade@scale}%
1276 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1277 \edef\wl@X@UV{\wl@X@tmp bp}%
1278 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1279 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1280 38,39,40,41,42,43,44,45,46,47,48,49,50,51\do{%
1281 \pgfmathparse{380+(\wl@shadeend-380)/50*(\n-1)}%
1282 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1283 \w!color{\wl@currentwl}%
1284 \colorlet{w!color}{\wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1285 \edef\wl@colorname{w!shcol\Roman\n}\relax\colorlet{
\wl@colorname}{w!color}%
1286 \pgfmathparse{25+(380-\wl@shadebegin+(\wl@shadeend-380)/50*(\n
-1))/\wl@shade@scale}%
1287 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1288 \expandafter\edef\csname x\Roman\n\endcsname{\wl@X@tmp bp}%
1289 }%
1290 % the horizontal shading
1291 \pgfdeclarehorizontalshading{#2}{100bp}{ color(0bp)=(\wl@UVcolor!
\wl@shade@opacity!\wl@shade@opacitycolor) ; color(\wl@X@UV)=(
\wl@UVcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;%
1292 color(\xl)=(w!shcolI) ; color(\xII)=(w!shcolII) ; color(\xIII)=(w!shcolIII) ;
color(\xIV)=(w!shcolIV) ; color(\xV)=(w!shcolV) ; color(\xVI)=(w!shcolVI) ;
color(\xVII)=(w!shcolVII) ; color(\xVIII)=(w!shcolVIII) ; color(\xIX)=(
w!shcolIX) ; color(\xX)=(w!shcolX) ; color(\xXI)=(w!shcolXI) ; color(\xXII)=(
w!shcolXII) ; color(\xXIII)=(w!shcolXIII) ; color(\xXIV)=(w!shcolXIV) ;
color(\xXV)=(w!shcolXV) ; color(\xXVI)=(w!shcolXVI) ; color(\xXVII)=(
w!shcolXVII) ; color(\xXVIII)=(w!shcolXVIII) ; color(\xXIX)=(w!shcolXIX) ;
color(\xXX)=(w!shcolXX) ; color(\xXXI)=(w!shcolXXI) ; color(\xXXII)=(
w!shcolXXII) ; color(\xXXIII)=(w!shcolXXIII) ; color(\xXXIV)=(w!shcolXXIV) ;
color(\xXXV)=(w!shcolXXV) ; color(\xXXVI)=(w!shcolXXVI) ; color(\xXXVII)=(
w!shcolXXVII) ; color(\xXXVIII)=(w!shcolXXVIII) ; color(\xXXIX)=(
w!shcolXXIX) ; color(\xXXX)=(w!shcolXXX) ; color(\xXXXI)=(w!shcolXXXI) ;
color(\xXXXII)=(w!shcolXXXII) ; color(\xXXXIII)=(w!shcolXXXIII) ; color(
\xXXXIV)=(w!shcolXXXIV) ; color(\xXXXV)=(w!shcolXXXV) ; color(\xXXXVI)=(
w!shcolXXXVI) ; color(\xXXXVII)=(w!shcolXXXVII) ; color(\xXXXVIII)=(
w!shcolXXXVIII) ; color(\xXXXIX)=(w!shcolXXXIX) ; color(\xXL)=(w!shcolXL) ;

```

```

1293     color(\xXLI)=(wlishcolXLI); color(\xXLII)=(wlishcolXLII); color(\xXLIII)=(
        wlishcolXLIII); color(\xXLIV)=(wlishcolXLIV); color(\xXLV)=(wlishcolXLV);
        color(\xXLVI)=(wlishcolXLVI); color(\xXLVII)=(wlishcolXLVII); color(
        \xXLVIII)=(wlishcolXLVIII); color(\xXLIX)=(wlishcolXLIX); color(\xL)=(
        wlishcolL); color(\xLI)=(wlishcolLI);%
1294     color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
        color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);%
1295     \fi%
1296     \fi% \ifwl@logshade
1297     \else% \wl@shadeend@pt>780pt
1298     \ifdim \wl@shadeend@pt>780pt \relax%
1299     %

```

```

1300     % \wl@shadebegin>=380nm & \wl@shadeend>780nm

```

```

1301     % Shade structure :
1302     %     xShade(bp) = f(wl)
1303     %     xVIS_from_\wl@shadebegin[bp]@25bp — xVIS_to_\wl@shadeend[bp]@75bp
1304     %     VIS region in shade (51 points) → \n=1,2,...,51
1305     %

```

```

1306     \ifwl@logshade% \wl@logshadetrue
1307     \pgfmathparse{int(780-\wl@shadebegin)}\pgfmathprintnumber to{\pgfmathresult}{
        \wl@shade@VIS@amp@int}%
1308     \ifnum \wl@shade@VIS@amp@int>20 \relax%
1309     \pgfmathparse{50/(log10(\wl@shadeend)-log10(\wl@shadebegin))}%
1310     \pgfmathprintnumber to{\pgfmathresult}{\wl@shade@logscale@factor}%
1311     \pgfmathparse{25+(log10(780.1)-log10(\wl@shadebegin))*
        \wl@shade@logscale@factor}%
1312     \pgfmathprintnumber to{\pgfmathresult}{\wl@X@tmp}%
1313     \edef \wl@X@IR{\wl@X@tmp bp}%
1314     \@for \n:= {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1315     21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1316     38,39,40,41,42,43,44,45,46,47,48,49,50,51} \do {%
1317     \pgfmathparse{\wl@shadebegin+(780-\wl@shadebegin)/50*(\n-1)}%
1318     \pgfmathprintnumber to{\pgfmathresult}{\wl@currentwl}%
1319     \wcolor{\wl@currentwl}%
1320     \colorlet{\wcolor}{\wl@temp!\wl@shade@opacity!
        \wl@shade@opacitycolor}%
1321     \edef \wl@colorname{wlishcol@\Roman\n} \relax \colorlet{
        \wl@colorname}{\wcolor}%
1322     \pgfmathparse{25+(log10(\wl@shadebegin+(780-\wl@shadebegin)/50*(
        \n-1))-log10(\wl@shadebegin))*\wl@shade@logscale@factor}%
1323     \pgfmathprintnumber to{\pgfmathresult}{\wl@X@tmp}%
1324     \expandafter \edef \csname x@\Roman\n\endcsname {\wl@X@tmp bp}%
1325     }%
1326     % the horizontal shading
1327     \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
        \@wl@shade@opacity!\wl@shade@opacitycolor); color(24.99bp)=(\wl@UVcolor
        !\wl@shade@opacity!\wl@shade@opacitycolor);%
1328     color(\xI)=(wlishcolI); color(\xII)=(wlishcolII); color(\xIII)=(wlishcolIII);
        color(\xIV)=(wlishcolIV); color(\xV)=(wlishcolV); color(\xVI)=(wlishcolVI);
        color(\xVII)=(wlishcolVII); color(\xVIII)=(wlishcolVIII); color(\xIX)=(
        wlishcolIX); color(\xX)=(wlishcolX); color(\xXI)=(wlishcolXI); color(\xXII)=
        (wlishcolXII); color(\xXIII)=(wlishcolXIII); color(\xXIV)=(wlishcolXIV);
        color(\xXV)=(wlishcolXV); color(\xXVI)=(wlishcolXVI); color(\xXVII)=(
        wlishcolXVII); color(\xXVIII)=(wlishcolXVIII); color(\xXIX)=(wlishcolXIX);
        color(\xXX)=(wlishcolXX); color(\xXXI)=(wlishcolXXI); color(\xXXII)=(
        wlishcolXXII); color(\xXXIII)=(wlishcolXXIII); color(\xXXIV)=(wlishcolXXIV);
        color(\xXXV)=(wlishcolXXV); color(\xXXVI)=(wlishcolXXVI); color(\xXXVII)=
        (wlishcolXXVII); color(\xXXVIII)=(wlishcolXXVIII); color(\xXXIX)=(
        wlishcolXXIX); color(\xXXX)=(wlishcolXXX); color(\xXXXI)=(wlishcolXXXI);
        color(\xXXXII)=(wlishcolXXXII); color(\xXXXIII)=(wlishcolXXXIII); color(
        \xXXXIV)=(wlishcolXXXIV); color(\xXXXV)=(wlishcolXXXV); color(\xXXXVI)=(
        wlishcolXXXVI); color(\xXXXVII)=(wlishcolXXXVII); color(\xXXXVIII)=(
        wlishcolXXXVIII); color(\xXXXIX)=(wlishcolXXXIX); color(\xXL)=(wlishcolXL);
        color(\xXLI)=(wlishcolXLI); color(\xXLII)=(wlishcolXLII); color(\xXLIII)=(

```

```

w\shcolXLIII); color(\xXLIV)=(w\shcolXLIV); color(\xXLV)=(w\shcolXLV);
color(\xXLVI)=(w\shcolXLVI); color(\xXLVII)=(w\shcolXLVII); color(
\xXLVIII)=(w\shcolXLVIII); color(\xXLIX)=(w\shcolXLIX); color(\xL)=(
w\shcolL); color(\xLI)=(w\shcolLI);%
1329 color(\wl@X@IR)=(\wl@IRcolor!\@wl@shade@opacity!\wl@shade@opacitycolor);
color(75.01bp)=(\wl@IRcolor!\@wl@shade@opacity!\wl@shade@opacitycolor);
color(100bp)=(\wl@IRcolor!\@wl@shade@opacity!\wl@shade@opacitycolor)}
%
1330 \else% amp<=20
1331 \pgfmathparse{1/(log10(\wl@shadeend)-log10(\wl@shadebegin))}%
1332 \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1333 \ifnum\wl@shade@VIS@amp=int=0\relax%
1334 \pgfmathparse{25+(log10(780.1)-log10(\wl@shadebegin))*50*
\wl@shade@logscale@factor}%
1335 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1336 \edef\wl@X@IR{\wl@X@tmp bp}%
1337 \else%
1338 \pgfmathparse{25+(log10(781)-log10(\wl@shadebegin))*50*
\wl@shade@logscale@factor}%
1339 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1340 \edef\wl@X@IR{\wl@X@tmp bp}%
1341 \fi%
1342 \@for\n:={1,6,11,16,21,26,31,36,41,46,51}\do{%
1343 \pgfmathparse{\wl@shadebegin+(780-\wl@shadebegin)/50*(\n-1)}%
1344 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1345 \wcolor{\wl@currentwl}%
1346 \colorlet{\wcolor}{\wl@temp!\@wl@shade@opacity!
\wl@shade@opacitycolor}%
1347 \edef\wl@colorname{w\shcol@\Roman\n}\relax\colorlet{
\wl@colorname}{\wcolor}%
1348 \pgfmathparse{25+(log10(\wl@shadebegin+(780-\wl@shadebegin)/50*(
\n-1))-log10(\wl@shadebegin))*50*\wl@shade@logscale@factor}%
1349 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1350 \expandafter\edef\csname x@\Roman\n\endcsname{\wl@X@tmp bp}%
1351 }%
1352 % the horizontal shading
1353 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
\@wl@shade@opacity!\wl@shade@opacitycolor); color(24.99bp)=(\wl@UVcolor
!\@wl@shade@opacity!\wl@shade@opacitycolor);%
1354 color(\xI)=(w\shcolI); color(\xVI)=(w\shcolVI); color(\xXI)=(w\shcolXI);
color(\xXVI)=(w\shcolXVI); color(\xXXI)=(w\shcolXXI); color(\xXXVI)=(
w\shcolXXVI); color(\xXXXI)=(w\shcolXXXI); color(\xXXXVI)=(w\shcolXXXVI);
color(\xXLI)=(w\shcolXLI); color(\xXLVI)=(w\shcolXLVI); color(\xLI)=(
w\shcolLI);%
1355 color(\wl@X@IR)=(\wl@IRcolor!\@wl@shade@opacity!\wl@shade@opacitycolor);
color(75.01bp)=(\wl@IRcolor!\@wl@shade@opacity!\wl@shade@opacitycolor);
color(100bp)=(\wl@IRcolor!\@wl@shade@opacity!\wl@shade@opacitycolor)}
%
1356 \fi%
1357 \else% \wl@logshadefalse
1358 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1359 \edef\wl@shade@scale{\pgfmathresult}%
1360 \pgfmathparse{25+(780.01-\wl@shadebegin)/\wl@shade@scale}%
1361 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1362 \edef\wl@X@IR{\wl@X@tmp bp}%
1363 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1364 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1365 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1366 \pgfmathparse{\wl@shadebegin+(780-\wl@shadebegin)/50*(\n-1)}%
1367 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1368 \wcolor{\wl@currentwl}%
1369 \colorlet{\wcolor}{\wl@temp!\@wl@shade@opacity!
\wl@shade@opacitycolor}%
1370 \edef\wl@colorname{w\shcol@\Roman\n}\relax\colorlet{
\wl@colorname}{\wcolor}%
1371 \pgfmathparse{25+(780-\wl@shadebegin)/50*(\n-1)/\wl@shade@scale}
%
1372 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1373 \expandafter\edef\csname x@\Roman\n\endcsname{\wl@X@tmp bp}%
1374 }%

```

```

1375 % the horizontal shading
1376 \pgfdeclarehorizontalshading{#2}{100bp}{ color(0bp)=(\wl@UVcolor!
    \wl@shade@opacity!\wl@shade@opacitycolor) ; color(24.99bp)=(\wl@UVcolor!
    \wl@shade@opacity!\wl@shade@opacitycolor) ;%
1377 color(\xI)=(wlscolI) ; color(\xII)=(wlscolII) ; color(\xIII)=(wlscolIII) ;
    color(\xIV)=(wlscolIV) ; color(\xV)=(wlscolV) ; color(\xVI)=(wlscolVI) ;
    color(\xVII)=(wlscolVII) ; color(\xVIII)=(wlscolVIII) ; color(\xIX)=(
    wlscolIX) ; color(\xX)=(wlscolX) ; color(\xXI)=(wlscolXI) ; color(\xXII)=(
    wlscolXII) ; color(\xXIII)=(wlscolXIII) ; color(\xXIV)=(wlscolXIV) ; color(
    \xXV)=(wlscolXV) ; color(\xXVI)=(wlscolXVI) ; color(\xXVII)=(wlscolXVII) ;
    color(\xXVIII)=(wlscolXVIII) ; color(\xXIX)=(wlscolXIX) ; color(\xXX)=(
    wlscolXX) ; color(\xXXI)=(wlscolXXI) ; color(\xXXII)=(wlscolXXII) ; color(
    \xXXIII)=(wlscolXXIII) ; color(\xXXIV)=(wlscolXXIV) ; color(\xXXV)=(
    wlscolXXV) ; color(\xXXVI)=(wlscolXXVI) ; color(\xXXVII)=(wlscolXXVII) ;
    color(\xXXVIII)=(wlscolXXVIII) ; color(\xXXIX)=(wlscolXXIX) ; color(\xXXX)=(
    wlscolXXX) ; color(\xXXXI)=(wlscolXXXI) ; color(\xXXXII)=(wlscolXXXII) ;
    color(\xXXXIII)=(wlscolXXXIII) ; color(\xXXXIV)=(wlscolXXXIV) ; color(\xXXXV
    )=(wlscolXXXV) ; color(\xXXXVI)=(wlscolXXXVI) ; color(\xXXXVII)=(
    wlscolXXXVII) ; color(\xXXXVIII)=(wlscolXXXVIII) ; color(\xXXXIX)=(
    wlscolXXXIX) ; color(\xXL)=(wlscolXL) ; color(\xXLI)=(wlscolXLI) ; color(
    \xXLII)=(wlscolXLII) ; color(\xXLIII)=(wlscolXLIII) ; color(\xXLIV)=(
    wlscolXLIV) ; color(\xXLV)=(wlscolXLV) ; color(\xXLVI)=(wlscolXLVI) ; color(
    \xXLVII)=(wlscolXLVII) ; color(\xXLVIII)=(wlscolXLVIII) ; color(\xXLIX)=(
    wlscolXLIX) ; color(\xL)=(wlscolL) ; color(\xLI)=(wlscolLI) ;%
1378 color(\wl@X@IR)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;
    color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor) ;
    color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor)}%
1379 \fi% \ifwl@logshade
1380 \else% \wl@shadeend@pt>780pt
1381 %


---


1382 % \wl@shadebegin>=380nm & \wl@shadeend<=780nm
1383 % Shade structure :
1384 % xShade(bp) = f(wl)
1385 % xVIS_from_\wl@shadebegin[bp]@25bp — xVIS_to_\wl@shadeend[bp]@75bp
1386 % VIS region in shade (51 points) -> \n=1,2,...,51
1387 %


---


1388 \pgfmathparse{((\wl@shadeend-\wl@shadebegin)/50)\pgfmathprintnumberto{
    \pgfmathresult}{\wl@shade@step}%
1389 \ifwl@logshade% \wl@logshadetrue
1390 % scale (25 to 75 bp) : \wl@shade@amp=\wl@shadeend-\wl@shadebegin ;
    \wl@shade@scale=\wl@shade@amp/50
1391 \pgfmathparse{\wl@shadeend-\wl@shadebegin} \edef\wl@shade@amp{\pgfmathresult}
    %
1392 \pgfmathprintnumberto{\wl@shade@amp}{\wl@shade@amp@int}%
1393 \ifnum\wl@shade@amp@int>5\relax%
1394 \pgfmathparse{50/(log10(\wl@shadeend)-log10(\wl@shadebegin))}
    \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1395 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1396 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1397 38,39,40,41,42,43,44,45,46,47,48,49,50,51\do{%
1398 \pgfmathparse{\wl@shadebegin+(\n-1)*\wl@shade@step}%
1399 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1400 \wcolor{\wl@currentwl}%
1401 \colorlet{\wcolor}{\wl@temp!\wl@shade@opacity!
    \wl@shade@opacitycolor}%
1402 \edef\wl@colorname{wlscol\@Roman\n} \relax \colorlet{
    \wl@colorname}{\wcolor}%
1403 \pgfmathparse{25+((log10(\wl@currentwl)-log10(\wl@shadebegin))*
    \wl@shade@logscale@factor)%
1404 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1405 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1406 }%
1407 \edef\xLI{75bp}%
1408 % the horizontal shading
1409 \pgfdeclarehorizontalshading{#2}{100bp}{ color(0bp)=(\wl@UVcolor!
    \wl@shade@opacity!\wl@shade@opacitycolor) ; color(24.94bp)=(\wl@UVcolor

```

```

! \@wl@shade@opacity! \@wl@shade@opacitycolor) ; color (24.95bp) = (wshcoll) ;
%
1410 color (\xI) = (wshcoll) ; color (\xII) = (wshcolII) ; color (\xIII) = (
wshcolIII) ; color (\xIV) = (wshcolIV) ; color (\xV) = (wshcolV) ; color (
\xVI) = (wshcolVI) ; color (\xVII) = (wshcolVII) ; color (\xVIII) = (
wshcolVIII) ; color (\xIX) = (wshcolIX) ; color (\xX) = (wshcolX) ; color (
\xXI) = (wshcolXI) ; color (\xXII) = (wshcolXII) ; color (\xXIII) = (
wshcolXIII) ; color (\xXIV) = (wshcolXIV) ; color (\xXV) = (wshcolXV) ;
color (\xXVI) = (wshcolXVI) ; color (\xXVII) = (wshcolXVII) ; color (
\xXVIII) = (wshcolXVIII) ; color (\xXIX) = (wshcolXIX) ; color (\xXX) = (
wshcolXX) ; color (\xXXI) = (wshcolXXI) ; color (\xXXII) = (wshcolXXII) ;
color (\xXXIII) = (wshcolXXIII) ; color (\xXXIV) = (wshcolXXIV) ; color (
\xXXV) = (wshcolXXV) ; color (\xXXVI) = (wshcolXXVI) ; color (\xXXVII) = (
wshcolXXVII) ; color (\xXXVIII) = (wshcolXXVIII) ; color (\xXXIX) = (
wshcolXXIX) ; color (\xXXX) = (wshcolXXX) ; color (\xXXXI) = (wshcolXXXI)
; color (\xXXXII) = (wshcolXXXII) ; color (\xXXXIII) = (wshcolXXXIII) ;
color (\xXXXIV) = (wshcolXXXIV) ; color (\xXXXV) = (wshcolXXXV) ; color (
\xXXXVI) = (wshcolXXXVI) ; color (\xXXXVII) = (wshcolXXXVII) ; color (
\xXXXVIII) = (wshcolXXXVIII) ; color (\xXXXIX) = (wshcolXXXIX) ; color (
\xXL) = (wshcolXL) ; color (\xXLI) = (wshcolXLI) ; color (\xXLII) = (
wshcolXLII) ; color (\xXLIII) = (wshcolXLIII) ; color (\xXLIV) = (
wshcolXLIV) ; color (\xXLV) = (wshcolXLV) ; color (\xXLVI) = (wshcolXLVI)
; color (\xXLVII) = (wshcolXLVII) ; color (\xXLVIII) = (wshcolXLVIII) ;
color (\xXLIX) = (wshcolXLIX) ; color (\xL) = (wshcolL) ; color (\xLI) = (
wshcolLI) ; %
1411 color (75.01bp) = (\wl@IRcolor! \@wl@shade@opacity!
\wl@shade@opacitycolor) ; color (100bp) = (\wl@IRcolor!
\wl@shade@opacity! \@wl@shade@opacitycolor) %
1412 \else% \wl@shade@amp<5
1413 \pgfmathparse{(log 10(\wl@shadeend)-log 10(\wl@shadebegin))}
\pgfmathprintnumero{\pgfmathresult}{\wl@shade@logscale@factor}%
1414 \wlcOLOR{\wl@shadebegin}%
1415 \colorlet{\wlcOLOR}{\wl@temp! \@wl@shade@opacity! \@wl@shade@opacitycolor}%
1416 \colorlet{wshcoll}{\wlcOLOR}%
1417 \edef\xI{25bp}%
1418 \@for\n:={11,21,31,41}\do{%
1419 \pgfmathparse{\wl@shadebegin+(\n-1)*\wl@shade@step}%
1420 \pgfmathprintnumero{\pgfmathresult}{\wl@currentwl}%
1421 \wlcOLOR{\wl@currentwl}%
1422 \colorlet{\wlcOLOR}{\wl@temp! \@wl@shade@opacity!
\wl@shade@opacitycolor}%
1423 \edef\wl@colorname{wshcol\@Roman\n} \relax \colorlet{
\wl@colorname}{\wlcOLOR}%
1424 \pgfmathparse{25+(log 10(\wl@currentwl)*50-log 10(\wl@shadebegin)
*50)/\wl@shade@logscale@factor}%
1425 \pgfmathprintnumero{\pgfmathresult}{\wl@X@tmp}%
1426 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1427 }%
1428 \wlcOLOR{\wl@shadeend}%
1429 \colorlet{\wlcOLOR}{\wl@temp! \@wl@shade@opacity! \@wl@shade@opacitycolor}%
1430 \colorlet{wshcolLI}{\wlcOLOR}%
1431 \edef\xLI{75bp}%
1432 % the horizontal shading
1433 \pgfdeclarehorizontalshading{#2}{100bp}{ color (0bp) = (\wl@UVcolor!
\wl@shade@opacity! \@wl@shade@opacitycolor) ; color (24.94bp) = (\wl@UVcolor
! \@wl@shade@opacity! \@wl@shade@opacitycolor) ; color (24.95bp) = (wshcoll) ;
%
1434 color (\xI) = (wshcoll) ; color (\xXI) = (wshcolXI) ; color (\xXXI) = (
wshcolXXI) ; color (\xXXXI) = (wshcolXXXI) ; color (\xXLI) = (wshcolXLI) ;
color (\xLI) = (wshcolLI) ; %
1435 color (75.01bp) = (\wl@IRcolor! \@wl@shade@opacity!
\wl@shade@opacitycolor) ; color (100bp) = (\wl@IRcolor!
\wl@shade@opacity! \@wl@shade@opacitycolor) %
1436 \fi% amp>5
1437 \else% \wl@logshadefalse
1438 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1439 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1440 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1441 \pgfmathparse{\wl@shadebegin+(\n-1)*\wl@shade@step}%
1442 \pgfmathprintnumero{\pgfmathresult}{\wl@currentwl}%

```

```

1443         \wcolor{\wl@currentwl}%
1444         \colorlet{wcolor}{wl@temp!\wl@shade@opacity!
           \wl@shade@opacitycolor}%
1445         \edef\wl@colorname{\wshcol\@Roman\n}\relax\colorlet{
           \wl@colorname}{wcolor}%
1446         \pgfmathparse{24+\n}%
1447         \pgfmathprintnumber{\pgfmathresult}{\wl@X@tmp}%
1448         \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1449     }%
1450 % the horizontal shading
1451 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
           \wl@shade@opacity!\wl@shade@opacitycolor);color(24.94bp)=(\wl@UVcolor!
           \wl@shade@opacity!\wl@shade@opacitycolor);color(24.95bp)=(wshcolI);%
1452     color(\xI)=(wshcolI);color(\xII)=(wshcolII);color(\xIII)=(wshcolIII);
           color(\xIV)=(wshcolIV);color(\xV)=(wshcolV);color(\xVI)=(wshcolVI);
           color(\xVII)=(wshcolVII);color(\xVIII)=(wshcolVIII);color(\xIX)=(
           wshcolIX);color(\xX)=(wshcolX);color(\xXI)=(wshcolXI);color(\xXII)
           =(wshcolXII);color(\xXIII)=(wshcolXIII);color(\xXIV)=(wshcolXIV);
           color(\xXV)=(wshcolXV);color(\xXVI)=(wshcolXVI);color(\xXVII)=(
           wshcolXVII);color(\xXVIII)=(wshcolXVIII);color(\xXIX)=(wshcolXIX);
           color(\xXX)=(wshcolXX);color(\xXXI)=(wshcolXXI);color(\xXXII)=(
           wshcolXXII);color(\xXXIII)=(wshcolXXIII);color(\xXXIV)=(wshcolXXIV)
           ;color(\xXXV)=(wshcolXXV);color(\xXXVI)=(wshcolXXVI);color(\xXXVII)
           =(wshcolXXVII);color(\xXXVIII)=(wshcolXXVIII);color(\xXXIX)=(
           wshcolXXIX);color(\xXXX)=(wshcolXXX);color(\xXXXI)=(wshcolXXXI);
           color(\xXXXII)=(wshcolXXXII);color(\xXXXIII)=(wshcolXXXIII);color(
           \xXXXIV)=(wshcolXXXIV);color(\xXXXV)=(wshcolXXXV);color(\xXXXVI)=(
           wshcolXXXVI);color(\xXXXVII)=(wshcolXXXVII);color(\xXXXVIII)=(
           wshcolXXXVIII);color(\xXXXIX)=(wshcolXXXIX);color(\xXL)=(wshcolXL);
           color(\xXLI)=(wshcolXLI);color(\xXLII)=(wshcolXLII);color(\xXLIII)=(
           wshcolXLIII);color(\xXLIV)=(wshcolXLIV);color(\xXLV)=(wshcolXLV);
           color(\xXLVI)=(wshcolXLVI);color(\xXLVII)=(wshcolXLVII);color(
           \xXLVIII)=(wshcolXLVIII);color(\xXLIX)=(wshcolXLIX);color(\xL)=(
           wshcolL);color(\xLI)=(wshcolLI);%
1453     color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
           color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor)}%
1454     \fi% \ifwl@logshade
1455     \fi% \wl@shadeend@pt>780pt
1456     \fi% \wl@shadebegin@pt<380pt\relax
1457 \fi\fi% \wl@shadeend@pt<380pt || \wl@shadebegin@pt>780pt
1458 \fi% \wl@shadeend@pt<\wl@shadebegin@pt
1459 \fi\fi% \wl<0nm && \wl > 16000 nm
1460 }%
1461 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1462 % \pgfspectraplotmap<|h>{name}
1463 % l -> low resolution (51 colors -> 380 to 780 nm ; step 8 nm) ! Default
1464 % h -> high resolution (401 colors -> 380 to 780 nm ; step 1 nm)
1465 \def\pgfspectraplotmap{\ignorespaces\@ifnextchar[\wl@pgfspectraplotmap{\wl@pgfspectraplotmap
           [l]}}%
1466 \def\wl@pgfspectraplotmap[#1]#2{\ignorespaces%
1467 \def\wl@test{#1}\def\wl@l{l}\def\wl@h{h}%
1468 \ifx\wl@test\wl@l\relax%
1469 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1470 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1471 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1472 %\pgfmathparse{380+(\n-1)*8}\edef\wl@currentwl{\pgfmathresult}%
1473 \pgfmathparse{372+8*\n}\edef\wl@currentwl{\pgfmathresult}%
1474 \wcolor{\wl@currentwl}%
1475 \edef\wl@colorname{\wshcol\@Roman\n}\relax\colorlet{\wl@colorname}{wcolor}%
1476 }%
1477 % the pgfplots colormap
1478 \pgfplotsset{/pgfplots/colormap={#2}{color(0bp)=(wshcolI);color(1bp)=(wshcolII);color(2
           bp)=(wshcolIII);color(3bp)=(wshcolIV);color(4bp)=(wshcolV);color(5bp)=(wshcolVI);
           color(6bp)=(wshcolVII);color(7bp)=(wshcolVIII);color(8bp)=(wshcolIX);color(9bp)=(
           wshcolX);color(10bp)=(wshcolXI);color(11bp)=(wshcolXII);color(12bp)=(wshcolXIII);
           color(13bp)=(wshcolXIV);color(14bp)=(wshcolXV);color(15bp)=(wshcolXVI);color(16bp)=(
           wshcolXVII);color(17bp)=(wshcolXVIII);color(18bp)=(wshcolXIX);color(19bp)=(wshcolXX
           );color(20bp)=(wshcolXXI);color(21bp)=(wshcolXXII);color(22bp)=(wshcolXXIII);color
           (23bp)=(wshcolXXIV);color(24bp)=(wshcolXXV);color(25bp)=(wshcolXXVI);color(26bp)=(
           wshcolXXVII);color(27bp)=(wshcolXXVIII);color(28bp)=(wshcolXXIX);color(29bp)=(

```

```

wshcolXXX); color(30bp)=(wshcolXXXI); color(31bp)=(wshcolXXXII); color(32bp)=(
wshcolXXXIII); color(33bp)=(wshcolXXXIV); color(34bp)=(wshcolXXXV); color(35bp)=(
wshcolXXXVI); color(36bp)=(wshcolXXXVII); color(37bp)=(wshcolXXXVIII); color(38bp)=(
wshcolXXXIX); color(39bp)=(wshcolXL); color(40bp)=(wshcolXLI); color(41bp)=(wshcolXLII
); color(42bp)=(wshcolXLIII); color(43bp)=(wshcolXLIV); color(44bp)=(wshcolXLV); color
(45bp)=(wshcolXLVI); color(46bp)=(wshcolXLVII); color(47bp)=(wshcolXLVIII); color(48bp)
=(wshcolXLIX); color(49bp)=(wshcolL); color(50bp)=(wshcolLI)}}%
1479 \else \ifx \wl@test\wl@h\relax %
1480 \pgfplotsset{/pgfplots/colormap=#2}{rgb(0bp)=(.3,0,.3); rgb(1bp)=(.3122,0,.3175); rgb(2bp)
=(.3238,0,.335); rgb(3bp)=(.3349,0,.3525); rgb(4bp)=(.3453,0,.37); rgb(5bp)
=(.3552,0,.3875); rgb(6bp)=(.3645,0,.405); rgb(7bp)=(.3732,0,.4225); rgb(8bp)
=(.3813,0,.44); rgb(9bp)=(.3889,0,.4575); rgb(10bp)=(.3958,0,.475); rgb(11bp)
=(.4022,0,.4925); rgb(12bp)=(.408,0,.51); rgb(13bp)=(.4132,0,.5275); rgb(14bp)
=(.4178,0,.545); rgb(15bp)=(.4219,0,.5625); rgb(16bp)=(.4253,0,.58); rgb(17bp)
=(.4282,0,.5975); rgb(18bp)=(.4305,0,.615); rgb(19bp)=(.4322,0,.6325); rgb(20bp)
=(.4333,0,.65); rgb(21bp)=(.4339,0,.6675); rgb(22bp)=(.4338,0,.685); rgb(23bp)
=(.4332,0,.7025); rgb(24bp)=(.432,0,.72); rgb(25bp)=(.4302,0,.7375); rgb(26bp)
=(.4278,0,.755); rgb(27bp)=(.4249,0,.7725); rgb(28bp)=(.4213,0,.79); rgb(29bp)
=(.4172,0,.8075); rgb(30bp)=(.4125,0,.825); rgb(31bp)=(.4072,0,.8425); rgb(32bp)
=(.4013,0,.86); rgb(33bp)=(.3949,0,.8775); rgb(34bp)=(.3878,0,.895); rgb(35bp)
=(.3802,0,.9125); rgb(36bp)=(.372,0,.93); rgb(37bp)=(.3632,0,.9475); rgb(38bp)
=(.3538,0,.965); rgb(39bp)=(.3439,0,.9825); rgb(40bp)=(.3333,0,1); rgb(41bp)=(.3167,0,1);
rgb(42bp)=(.3,0,1); rgb(43bp)=(.2833,0,1); rgb(44bp)=(.2667,0,1); rgb(45bp)=(.25,0,1); rgb
(46bp)=(.2333,0,1); rgb(47bp)=(.2167,0,1); rgb(48bp)=(.2,0,1); rgb(49bp)=(.1833,0,1); rgb
(50bp)=(.1667,0,1); %
1481 rgb(51bp)=(.15,0,1); rgb(52bp)=(.1333,0,1); rgb(53bp)=(.1167,0,1); rgb(54bp)=(.1,0,1); rgb
(55bp)=(.0833,0,1); rgb(56bp)=(.0667,0,1); rgb(57bp)=(.05,0,1); rgb(58bp)=(.0333,0,1); rgb
(59bp)=(.0167,0,1); rgb(60bp)=(0,0,1); rgb(61bp)=(0,.02,1); rgb(62bp)=(0,.04,1); rgb(63bp)
=(0,.06,1); rgb(64bp)=(0,.08,1); rgb(65bp)=(0,.1,1); rgb(66bp)=(0,.12,1); rgb(67bp)
=(0,.14,1); rgb(68bp)=(0,.16,1); rgb(69bp)=(0,.18,1); rgb(70bp)=(0,.2,1); rgb(71bp)
=(0,.22,1); rgb(72bp)=(0,.24,1); rgb(73bp)=(0,.26,1); rgb(74bp)=(0,.28,1); rgb(75bp)
=(0,.3,1); rgb(76bp)=(0,.32,1); rgb(77bp)=(0,.34,1); rgb(78bp)=(0,.36,1); rgb(79bp)
=(0,.38,1); rgb(80bp)=(0,.4,1); rgb(81bp)=(0,.42,1); rgb(82bp)=(0,.44,1); rgb(83bp)
=(0,.46,1); rgb(84bp)=(0,.48,1); rgb(85bp)=(0,.5,1); rgb(86bp)=(0,.52,1); rgb(87bp)
=(0,.54,1); rgb(88bp)=(0,.56,1); rgb(89bp)=(0,.58,1); rgb(90bp)=(0,.6,1); rgb(91bp)
=(0,.62,1); rgb(92bp)=(0,.64,1); rgb(93bp)=(0,.66,1); rgb(94bp)=(0,.68,1); rgb(95bp)
=(0,.7,1); rgb(96bp)=(0,.72,1); rgb(97bp)=(0,.74,1); rgb(98bp)=(0,.76,1); rgb(99bp)
=(0,.78,1); rgb(100bp)=(0,.8,1); %
1482 rgb(101bp)=(0,.82,1); rgb(102bp)=(0,.84,1); rgb(103bp)=(0,.86,1); rgb(104bp)=(0,.88,1); rgb
(105bp)=(0,.9,1); rgb(106bp)=(0,.92,1); rgb(107bp)=(0,.94,1); rgb(108bp)=(0,.96,1); rgb
(109bp)=(0,.98,1); rgb(110bp)=(0,1,1); rgb(111bp)=(0,1,.95); rgb(112bp)=(0,1,.9); rgb(113
bp)=(0,1,.85); rgb(114bp)=(0,1,.8); rgb(115bp)=(0,1,.75); rgb(116bp)=(0,1,.7); rgb(117bp)
=(0,1,.65); rgb(118bp)=(0,1,.6); rgb(119bp)=(0,1,.55); rgb(120bp)=(0,1,.5); rgb(121bp)
=(0,1,.45); rgb(122bp)=(0,1,.4); rgb(123bp)=(0,1,.35); rgb(124bp)=(0,1,.3); rgb(125bp)
=(0,1,.25); rgb(126bp)=(0,1,.2); rgb(127bp)=(0,1,.15); rgb(128bp)=(0,1,.1); rgb(129bp)
=(0,1,.05); rgb(130bp)=(0,1,0); rgb(131bp)=(.0143,1,0); rgb(132bp)=(.0286,1,0); rgb(133bp)
=(.0429,1,0); rgb(134bp)=(.0571,1,0); rgb(135bp)=(.0714,1,0); rgb(136bp)=(.0857,1,0); rgb
(137bp)=(.1,1,0); rgb(138bp)=(.1143,1,0); rgb(139bp)=(.1286,1,0); rgb(140bp)=(.1429,1,0);
rgb(141bp)=(.1571,1,0); rgb(142bp)=(.1714,1,0); rgb(143bp)=(.1857,1,0); rgb(144bp)
=(.2,1,0); rgb(145bp)=(.2143,1,0); rgb(146bp)=(.2286,1,0); rgb(147bp)=(.2429,1,0); rgb(148
bp)=(.2571,1,0); rgb(149bp)=(.2714,1,0); rgb(150bp)=(.2857,1,0); %
1483 rgb(151bp)=(.3,1,0); rgb(152bp)=(.3143,1,0); rgb(153bp)=(.3286,1,0); rgb(154bp)=(.3429,1,0);
rgb(155bp)=(.3571,1,0); rgb(156bp)=(.3714,1,0); rgb(157bp)=(.3857,1,0); rgb(158bp)
=(.4,1,0); rgb(159bp)=(.4143,1,0); rgb(160bp)=(.4286,1,0); rgb(161bp)=(.4429,1,0); rgb(162
bp)=(.4571,1,0); rgb(163bp)=(.4714,1,0); rgb(164bp)=(.4857,1,0); rgb(165bp)=(.5,1,0); rgb
(166bp)=(.5143,1,0); rgb(167bp)=(.5286,1,0); rgb(168bp)=(.5429,1,0); rgb(169bp)
=(.5571,1,0); rgb(170bp)=(.5714,1,0); rgb(171bp)=(.5857,1,0); rgb(172bp)=(.6,1,0); rgb(173
bp)=(.6143,1,0); rgb(174bp)=(.6286,1,0); rgb(175bp)=(.6429,1,0); rgb(176bp)=(.6571,1,0);
rgb(177bp)=(.6714,1,0); rgb(178bp)=(.6857,1,0); rgb(179bp)=(.7,1,0); rgb(180bp)
=(.7143,1,0); rgb(181bp)=(.7286,1,0); rgb(182bp)=(.7429,1,0); rgb(183bp)=(.7571,1,0); rgb
(184bp)=(.7714,1,0); rgb(185bp)=(.7857,1,0); rgb(186bp)=(.8,1,0); rgb(187bp)=(.8143,1,0);
rgb(188bp)=(.8286,1,0); rgb(189bp)=(.8429,1,0); rgb(190bp)=(.8571,1,0); rgb(191bp)
=(.8714,1,0); rgb(192bp)=(.8857,1,0); rgb(193bp)=(.9,1,0); rgb(194bp)=(.9143,1,0); rgb(195
bp)=(.9286,1,0); rgb(196bp)=(.9429,1,0); rgb(197bp)=(.9571,1,0); rgb(198bp)=(.9714,1,0);
rgb(199bp)=(.9857,1,0); rgb(200bp)=(1,1,0); %
1484 rgb(201bp)=(1,.9846,0); rgb(202bp)=(1,.9692,0); rgb(203bp)=(1,.9538,0); rgb(204bp)
=(1,.9385,0); rgb(205bp)=(1,.9231,0); rgb(206bp)=(1,.9077,0); rgb(207bp)=(1,.8923,0); rgb
(208bp)=(1,.8769,0); rgb(209bp)=(1,.8615,0); rgb(210bp)=(1,.8462,0); rgb(211bp)
=(1,.8308,0); rgb(212bp)=(1,.8154,0); rgb(213bp)=(1,.8,0); rgb(214bp)=(1,.7846,0); rgb(215
bp)=(1,.7692,0); rgb(216bp)=(1,.7538,0); rgb(217bp)=(1,.7385,0); rgb(218bp)=(1,.7231,0);

```

```

    rgb(219bp)=(1,.7077,0); rgb(220bp)=(1,.6923,0); rgb(221bp)=(1,.6769,0); rgb(222bp)
    =(1,.6615,0); rgb(223bp)=(1,.6462,0); rgb(224bp)=(1,.6308,0); rgb(225bp)=(1,.6154,0); rgb
    (226bp)=(1,.6,0); rgb(227bp)=(1,.5846,0); rgb(228bp)=(1,.5692,0); rgb(229bp)=(1,.5538,0);
    rgb(230bp)=(1,.5385,0); rgb(231bp)=(1,.5231,0); rgb(232bp)=(1,.5077,0); rgb(233bp)
    =(1,.4923,0); rgb(234bp)=(1,.4769,0); rgb(235bp)=(1,.4615,0); rgb(236bp)=(1,.4462,0); rgb
    (237bp)=(1,.4308,0); rgb(238bp)=(1,.4154,0); rgb(239bp)=(1,.4,0); rgb(240bp)=(1,.3846,0);
    rgb(241bp)=(1,.3692,0); rgb(242bp)=(1,.3538,0); rgb(243bp)=(1,.3385,0); rgb(244bp)
    =(1,.3231,0); rgb(245bp)=(1,.3077,0); rgb(246bp)=(1,.2923,0); rgb(247bp)=(1,.2769,0); rgb
    (248bp)=(1,.2615,0); rgb(249bp)=(1,.2462,0); rgb(250bp)=(1,.2308,0); %
1485 rgb(251bp)=(1,.2154,0); rgb(252bp)=(1,.2,0); rgb(253bp)=(1,.1846,0); rgb(254bp)=(1,.1692,0);
    rgb(255bp)=(1,.1538,0); rgb(256bp)=(1,.1385,0); rgb(257bp)=(1,.1231,0); rgb(258bp)
    =(1,.1077,0); rgb(259bp)=(1,.0923,0); rgb(260bp)=(1,.0769,0); rgb(261bp)=(1,.0615,0); rgb
    (262bp)=(1,.0462,0); rgb(263bp)=(1,.0308,0); rgb(264bp)=(1,.0154,0); rgb(265bp)=(1,0,0);
    rgb(266bp)=(1,0,0); rgb(267bp)=(1,0,0); rgb(268bp)=(1,0,0); rgb(269bp)=(1,0,0); rgb(270bp)
    =(1,0,0); rgb(271bp)=(1,0,0); rgb(272bp)=(1,0,0); rgb(273bp)=(1,0,0); rgb(274bp)=(1,0,0);
    rgb(275bp)=(1,0,0); rgb(276bp)=(1,0,0); rgb(277bp)=(1,0,0); rgb(278bp)=(1,0,0); rgb(279bp)
    =(1,0,0); rgb(280bp)=(1,0,0); rgb(281bp)=(1,0,0); rgb(282bp)=(1,0,0); rgb(283bp)=(1,0,0);
    rgb(284bp)=(1,0,0); rgb(285bp)=(1,0,0); rgb(286bp)=(1,0,0); rgb(287bp)=(1,0,0); rgb(288bp)
    =(1,0,0); rgb(289bp)=(1,0,0); rgb(290bp)=(1,0,0); rgb(291bp)=(1,0,0); rgb(292bp)=(1,0,0);
    rgb(293bp)=(1,0,0); rgb(294bp)=(1,0,0); rgb(295bp)=(1,0,0); rgb(296bp)=(1,0,0); rgb(297bp)
    =(1,0,0); rgb(298bp)=(1,0,0); rgb(299bp)=(1,0,0); rgb(300bp)=(1,0,0); %
1486 rgb(301bp)=(1,0,0); rgb(302bp)=(1,0,0); rgb(303bp)=(1,0,0); rgb(304bp)=(1,0,0); rgb(305bp)
    =(1,0,0); rgb(306bp)=(1,0,0); rgb(307bp)=(1,0,0); rgb(308bp)=(1,0,0); rgb(309bp)=(1,0,0);
    rgb(310bp)=(1,0,0); rgb(311bp)=(1,0,0); rgb(312bp)=(1,0,0); rgb(313bp)=(1,0,0); rgb(314bp)
    =(1,0,0); rgb(315bp)=(1,0,0); rgb(316bp)=(1,0,0); rgb(317bp)=(1,0,0); rgb(318bp)=(1,0,0);
    rgb(319bp)=(1,0,0); rgb(320bp)=(1,0,0); rgb(321bp)=(.9913,0,0); rgb(322bp)=(.9825,0,0);
    rgb(323bp)=(.9738,0,0); rgb(324bp)=(.965,0,0); rgb(325bp)=(.9563,0,0); rgb(326bp)
    =(.9475,0,0); rgb(327bp)=(.9388,0,0); rgb(328bp)=(.93,0,0); rgb(329bp)=(.9213,0,0); rgb
    (330bp)=(.9125,0,0); rgb(331bp)=(.9038,0,0); rgb(332bp)=(.895,0,0); rgb(333bp)=(.8863,0,0)
    ; rgb(334bp)=(.8775,0,0); rgb(335bp)=(.8688,0,0); rgb(336bp)=(.86,0,0); rgb(337bp)
    =(.8513,0,0); rgb(338bp)=(.8425,0,0); rgb(339bp)=(.8338,0,0); rgb(340bp)=(.825,0,0); rgb
    (341bp)=(.8163,0,0); rgb(342bp)=(.8075,0,0); rgb(343bp)=(.7988,0,0); rgb(344bp)=(.79,0,0);
    rgb(345bp)=(.7813,0,0); rgb(346bp)=(.7725,0,0); rgb(347bp)=(.7638,0,0); rgb(348bp)
    =(.755,0,0); rgb(349bp)=(.7463,0,0); rgb(350bp)=(.7375,0,0); %
1487 rgb(351bp)=(.7288,0,0); rgb(352bp)=(.72,0,0); rgb(353bp)=(.7113,0,0); rgb(354bp)=(.7025,0,0)
    ; rgb(355bp)=(.6938,0,0); rgb(356bp)=(.685,0,0); rgb(357bp)=(.6763,0,0); rgb(358bp)
    =(.6675,0,0); rgb(359bp)=(.6588,0,0); rgb(360bp)=(.65,0,0); rgb(361bp)=(.6413,0,0); rgb
    (362bp)=(.6325,0,0); rgb(363bp)=(.6238,0,0); rgb(364bp)=(.615,0,0); rgb(365bp)=(.6063,0,0)
    ; rgb(366bp)=(.5975,0,0); rgb(367bp)=(.5888,0,0); rgb(368bp)=(.58,0,0); rgb(369bp)
    =(.5713,0,0); rgb(370bp)=(.5625,0,0); rgb(371bp)=(.5538,0,0); rgb(372bp)=(.545,0,0); rgb
    (373bp)=(.5363,0,0); rgb(374bp)=(.5275,0,0); rgb(375bp)=(.5188,0,0); rgb(376bp)=(.51,0,0);
    rgb(377bp)=(.5013,0,0); rgb(378bp)=(.4925,0,0); rgb(379bp)=(.4838,0,0); rgb(380bp)
    =(.475,0,0); rgb(381bp)=(.4663,0,0); rgb(382bp)=(.4575,0,0); rgb(383bp)=(.4488,0,0); rgb
    (384bp)=(.44,0,0); rgb(385bp)=(.4313,0,0); rgb(386bp)=(.4225,0,0); rgb(387bp)=(.4138,0,0);
    rgb(388bp)=(.405,0,0); rgb(389bp)=(.3963,0,0); rgb(390bp)=(.3875,0,0); rgb(391bp)
    =(.3788,0,0); rgb(392bp)=(.37,0,0); rgb(393bp)=(.3613,0,0); rgb(394bp)=(.3525,0,0); rgb
    (395bp)=(.3438,0,0); rgb(396bp)=(.335,0,0); rgb(397bp)=(.3263,0,0); rgb(398bp)=(.3175,0,0)
    ; rgb(399bp)=(.3088,0,0); rgb(400bp)=(.3,0,0); %
1488 }}%
1489 \fi \fi%
1490 }%
1491 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1492 % \pgfspectrainbow[<tikz options>](<rainbow fade>,<rainbow start>,<rainbow knock out>,<
    rainbow background>,<rainbow transparency>){radius}
1493 % tikz options -> color, opacity, scope fading
1494 % rainbow clip -> applies a scope fading in clipped region
1495 % ...
1496 \pgfkeys{/wl/.cd,%
1497 rainbow fade/.get=\wl@rainbowfade,%
1498 rainbow fade/.store in=\wl@rainbowfade,%
1499 rainbow fade/.default={},%
1500 rainbow start/.get=\wl@rainbow@start,%
1501 rainbow start/.store in=\wl@rainbow@start,%
1502 rainbow start/.default=.6,% -> 60%
1503 rainbow knock out/.get=\wl@rainbow@KO,%
1504 rainbow knock out/.store in=\wl@rainbow@KO,%
1505 rainbow knock out/.default=.4,% -> 40%
1506 rainbow background/.get=\wl@rainbowback,%
1507 rainbow background/.store in=\wl@rainbowback,%
1508 rainbow background/.default=white,%

```

```

1509 rainbow transparency/.get=\wl@rainbowtransp,%
1510 rainbow transparency/.store in=\wl@rainbowtransp,%
1511 rainbow transparency/.default=0}% -> 0%
1512 %
1513 \def\pgfspectrarainbow{\ignorespaces \@ifnextchar[\pgf@spectrarainbow{\pgf@spectrarainbow []}]
%
1514 \def\pgf@spectrarainbow[#1]{\ignorespaces \@ifnextchar({\pgf@spectra@rainbow{#1}){\pgf@spectra@rainbow{#1}}}%
%
1515 %
1516 \def\pgf@spectra@rainbow#1(#2)#3{\ignorespaces%
1517 \pgfkeys{/wl/.cd,rainbow fade,rainbow start,rainbow knock out,rainbow background,rainbow
transparency}%
1518 \pgfkeys{/wl/.cd,#2}%
1519 \pgfmathparse{100-\wl@rainbowtransp*100}\edef\wl@rainbow@transp{\pgfmathresult}%
1520 \pgfmathparse{\wl@rainbowtransp*100}\edef\wl@rainbow@transp@w{\pgfmathresult}%
1521 \edef\wl@rainbowend{.8875cm}\pgfmathparse{\wl@rainbow@start*\wl@rainbowend/1cm}\edef
\wl@rainbowstart{\pgfmathresult cm}%
1522 \pgfmathparse{\wl@rainbow@KO*#3/1cm}\edef\wl@rainbowKO{\pgfmathresult cm}%\edef
\wl@rainbowKO{\wl@rainbow@KO cm}%
1523 \ifdim\wl@rainbowstart<\wl@rainbowend\relax%
1524 \ifdim\wl@rainbowstart<.0175cm\relax\edef\wl@rainbowstart{.0175cm}\fi% ensuring there is no
error in radial shading
1525 \pgfkeys{/pgf/number format/.cd,fixed,precision=3,set thousands separator={},assume math
mode=true}%
1526 \pgfmathparse{\wl@rainbowstart-1/50*(\wl@rainbowend-\wl@rainbowstart)}%
1527 \pgfmathprintnumber to{\pgfmathresult}{\wl@rainbowresult}%
1528 \edef\rO{\wl@rainbowresult pt}%
1529 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1530 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1531 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1532 %\pgfmathparse{380+(\n-1)*8}\edef\wl@currentwl{\pgfmathresult}%
1533 \pgfmathparse{372+8*\n}\edef\wl@currentwl{\pgfmathresult}%
1534 \wcolor{\wl@currentwl}%
1535 \edef\wl@colorname{wlshcol@Roman\n}\relax\colorlet{\wl@colorname}{\wcolor!100!transparent!
\wl@rainbow@transp}%
1536 \pgfmathparse{\wl@rainbowstart+1/50*(\n-1)*(\wl@rainbowend-\wl@rainbowstart)}%
1537 \pgfmathprintnumber to{\pgfmathresult}{\wl@rainbowresult}%
1538 \expandafter\edef\csname r@Roman\n\endcsname{\wl@rainbowresult pt}%
1539 }%
1540 \pgfdeclareradialshading{pgfspectrarainbow}{\pgfpoint{0pt}{0pt}}{%
1541 color(0cm)=(\wl@rainbowback!\wl@rainbow@transp@w!white);color(4/5*\rO)=(\wl@rainbowback!
\wl@rainbow@transp@w!white!50);color(\rO)=(white);%
1542 color(\rI)=(wlshcolI);color(\rII)=(wlshcolII);color(\rIII)=(wlshcolIII);color(\rIV)=(
wlshcolIV);color(\rV)=(wlshcolV);color(\rVI)=(wlshcolVI);color(\rVII)=(wlshcolVII);color(
\rVIII)=(wlshcolVIII);color(\rIX)=(wlshcolIX);color(\rX)=(wlshcolX);%
1543 color(\rXI)=(wlshcolXI);color(\rXII)=(wlshcolXII);color(\rXIII)=(wlshcolXIII);color(\rXIV)=(
wlshcolXIV);color(\rXV)=(wlshcolXV);color(\rXVI)=(wlshcolXVI);color(\rXVII)=(wlshcolXVII);
color(\rXVIII)=(wlshcolXVIII);color(\rXIX)=(wlshcolXIX);color(\rXX)=(wlshcolXX);%
1544 color(\rXXI)=(wlshcolXXI);color(\rXXII)=(wlshcolXXII);color(\rXXIII)=(wlshcolXXIII);color(
\rXXIV)=(wlshcolXXIV);color(\rXXV)=(wlshcolXXV);color(\rXXVI)=(wlshcolXXVI);color(\rXXVII)
=(wlshcolXXVII);color(\rXXVIII)=(wlshcolXXVIII);color(\rXXIX)=(wlshcolXXIX);color(\rXXX)=(
wlshcolXXX);%
1545 color(\rXXXI)=(wlshcolXXXI);color(\rXXXII)=(wlshcolXXXII);color(\rXXXIII)=(wlshcolXXXIII);
color(\rXXXIV)=(wlshcolXXXIV);color(\rXXXV)=(wlshcolXXXV);color(\rXXXVI)=(wlshcolXXXVI);
color(\rXXXVII)=(wlshcolXXXVII);color(\rXXXVIII)=(wlshcolXXXVIII);color(\rXXXIX)=(
wlshcolXXXIX);color(\rXL)=(wlshcolXL);%
1546 color(\rXLI)=(wlshcolXLI);color(\rXLII)=(wlshcolXLII);color(\rXLIII)=(wlshcolXLIII);color(
\rXLIV)=(wlshcolXLIV);color(\rXLV)=(wlshcolXLV);color(\rXLVI)=(wlshcolXLVI);color(\rXLVII)
=(wlshcolXLVII);color(\rXLVIII)=(wlshcolXLVIII);color(\rXLIX)=(wlshcolXLIX);color(\rL)=(
wlshcolL);color(\rLI)=(wlshcolLI);%
1547 color(.95cm)=(wlshcolLI)%
1548 }%
1549 \ifx\wl@rainbowfade\@empty\relax%
1550 \tikz{\clip(-#3,\wl@rainbowKO) rectangle ++(2*#3,#3-\wl@rainbowKO);%
1551 \fill[#1,shading=pgfspectrarainbow](0,0) circle(#3);}%
1552 \else%
1553 \tikz{\clip[scope fading=\wl@rainbowfade](-#3,\wl@rainbowKO) rectangle ++(2*#3,#3-
\wl@rainbowKO);%
1554 \fill[shading=pgfspectrarainbow,\wl@rainbowback,#1](0,0) circle(#3);}%
1555 \fi%

```

```

1556 \else\PackageError{pgf-spectra}{invalid 'rainbow start' value (rainbow start=
      \wl@rainbow@start). The rainbow start should be greater or equal then 0 and lower then
      1.}{Don't forget that 'rainbow start' value is the fraction from witch the colors begin,
      relative to the center of a circle with radius 1...}%
1557 \fi
1558 }%
1559 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1560 % \tempercolor -> Convert a temperature in Kelvin to rgb color
1561 %
1562 % The original algorithm used in this macro can be found at
1563 %
1564 % https://github.com/neilbartlett/color-temperature
1565 %
1566 % More information at...
1567 %
1568 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1569 %

1570 % https://tannerhelland.com/2012/09/18/convert-temperature-rgb-algorithm-code.html
1571 %
1572 % «Start with a temperature, in Kelvin, somewhere between 1000 and 40000. (Other values may
      work,
1573 % but I can't make any promises about the quality of the algorithm's estimates above 40000 K
      .)»
1574 %

      Tanner Helland
1575 % https://www.zombieprototypes.com/?p=210
1576 %

1577 % and the improved algorithm at...
1578 %

1579 % https://github.com/neilbartlett/color-temperature
1580 %
1581 % neilbartlett/color-temperature is licensed under the
1582 %
1583 % MIT License
1584 % A short and simple permissive license with conditions only requiring preservation of
      copyright and license notices.
1585 % Licensed works, modifications, and larger works may be distributed under different terms
      and without source code.
1586 %
1587 % Permissions
1588 % Commercial use, Modification, Distribution, Private use
1589 % Limitations
1590 % Liability, Warranty
1591 %
1592 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1593 % The MIT License (MIT)
1594 %
1595 % Copyright (c) 2015 Neil Bartlett
1596 %
1597 % Permission is hereby granted, free of charge, to any person obtaining a copy
      of this software and associated documentation files (the "Software"), to deal
1598 % in the Software without restriction, including without limitation the rights
1599 % to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
1600 % copies of the Software, and to permit persons to whom the Software is
1601 % furnished to do so, subject to the following conditions:
1602 %
1603 %
1604 % The above copyright notice and this permission notice shall be included in
1605 % all copies or substantial portions of the Software.
1606 %
1607 % THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1608 % IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1609 % FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
1610 % AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

```

```

1611 % LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1612 % OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1613 % THE SOFTWARE.
1614 % © 2021 GitHub, Inc.
1615 %
1616 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1617 \definecolor{tempercolor}{rgb}{0,0,0}%
1618 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1619 % \tempercolor{Kelvin}
1620 \def\tempercolor#1{\ignorespaces%
1621 %Set Temperature = Temperature \ 100
1622 \edef\wl@temperatura{#1sp}%
1623 \ifdim\wl@temperatura>16380sp\relax% for supporting large values of T
1624 \pgfkeys{/pgf/number format/.cd, fixed, precision=2, fixed zerofill, set thousands separator={},
    assume math mode=true}%
1625 \pgfmathparse{#1sp/100sp*1.0026931+.002}% correction factor for the division made in sp
1626 \pgfmathprintnumberto{\pgfmathresult}{\wl@T@tmp}%
1627 \edef\wl@temperatura{\wl@T@tmp}%
1628 \edef\wl@temperatura@pt{\pgfmathresult pt}%
1629 \else%
1630 \pgfmathparse{#1/100}%
1631 \edef\wl@temperatura{\pgfmathresult}%
1632 \edef\wl@temperatura@pt{\pgfmathresult pt}%
1633 \fi%
1634 \ifdim\wl@temperatura@pt<10pt\relax%
1635 \PackageError{pgf-spectra}{tempercolor: the temperature must be at least 1000K}{Type a
    greater temperature ...}%
1636 \else\ifdim\wl@temperatura@pt>400pt\relax%
1637 \PackageError{pgf-spectra}{tempercolor: the temperature must be at most 40000K}{Type a
    smaller temperature ...}%
1638 \else%
1639 %Calculate Red:
1640 \ifdim\wl@temperatura@pt>66pt\relax%
1641 \pgfmathparse{\wl@temperatura-55}\edef\wl@tempercolor@r{\pgfmathresult}%
1642 \pgfmathparse{351.976906+ 0.114207*\wl@tempercolor@r-40.253663*\ln(\wl@tempercolor@r)}%
1643 \edef\wl@tempercolor@r{\pgfmathresult}%
1644 \edef\wl@tempercolor@r@pt{\pgfmathresult pt}%
1645 \ifdim\wl@tempercolor@r@pt<0pt\edef\wl@tempercolor@r{0}\relax\fi%
1646 \ifdim\wl@tempercolor@r@pt>255pt\edef\wl@tempercolor@r{255}\relax\fi%
1647 \pgfmathparse{\wl@tempercolor@r/255}\edef\wl@tempercolor@r{\pgfmathresult}%
1648 \else%
1649 \edef\wl@tempercolor@r{1}%
1650 \fi%
1651 %Calculate Green:
1652 \ifdim\wl@temperatura@pt>66pt\relax%
1653 \pgfmathparse{\wl@temperatura-50}\edef\wl@tempercolor@g{\pgfmathresult}%
1654 \pgfmathparse{325.449413+0.079435*\wl@tempercolor@g-28.085296*\ln(\wl@tempercolor@g)}%
1655 \edef\wl@tempercolor@g{\pgfmathresult}%
1656 \edef\wl@tempercolor@g@pt{\pgfmathresult pt}%
1657 \ifdim\wl@tempercolor@g@pt<0pt\edef\wl@tempercolor@g{0}\relax\fi%
1658 \ifdim\wl@tempercolor@g@pt>255pt\edef\wl@tempercolor@g{255}\relax\fi%
1659 \pgfmathparse{\wl@tempercolor@g/255}\edef\wl@tempercolor@g{\pgfmathresult}%
1660 \else%
1661 \pgfmathparse{\wl@temperatura-2}\edef\wl@tempercolor@g{\pgfmathresult}%
1662 \pgfmathparse{-155.254856-0.445970*\wl@tempercolor@g+104.492162*\ln(\wl@tempercolor@g)}%
1663 \edef\wl@tempercolor@g{\pgfmathresult}%
1664 \edef\wl@tempercolor@g@pt{\pgfmathresult pt}%
1665 \ifdim\wl@tempercolor@g@pt<0pt\edef\wl@tempercolor@g{0}\relax\fi%
1666 \ifdim\wl@tempercolor@g@pt>255pt\edef\wl@tempercolor@g{255}\relax\fi%
1667 \pgfmathparse{\wl@tempercolor@g/255}\edef\wl@tempercolor@g{\pgfmathresult}%
1668 \fi%
1669 %Calculate Blue:
1670 \ifdim\wl@temperatura@pt<66pt\relax%
1671 \ifdim\wl@temperatura@pt>20pt\relax%
1672 \pgfmathparse{\wl@temperatura-10}\edef\wl@tempercolor@b{\pgfmathresult}%
1673 \pgfmathparse{-254.769352+0.827410*\wl@tempercolor@b+115.679944*\ln(\wl@tempercolor@b)}%
1674 \edef\wl@tempercolor@b{\pgfmathresult}%
1675 \edef\wl@tempercolor@b@pt{\pgfmathresult pt}%
1676 \ifdim\wl@tempercolor@b@pt<0pt\edef\wl@tempercolor@b{0}\relax\fi%
1677 \ifdim\wl@tempercolor@b@pt>255pt\edef\wl@tempercolor@b{255}\relax\fi%

```

```

1678 \pgfmathparse{\wl@tempercolor@b/255}\edef\wl@tempercolor@b{\pgfmathresult}%
1679 \else%
1680 \edef\wl@tempercolor@b{0}%
1681 \fi%
1682 \else%
1683 \edef\wl@tempercolor@b{1}%
1684 \fi%
1685 \definecolor{wl@@tempercolor}{rgb}{\wl@tempercolor@r,\wl@tempercolor@g,\wl@tempercolor@b}%
1686 \colorlet{tempercolor}{wl@@tempercolor}%
1687 \fi\fi% check limits 1000K<T<4000K
1688 }%
1689 % <--- NEW v2.1.0 ---
1690 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1691 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1692 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1693 %
1694 % nm2rgb convert nanometre wavelength to rgb
1695 % (380 <= Lambda <= 780 ) -> r,g,b on stack
1696 %
1697 % BASED on FORTRAN Code
1698 % RGB VALUES FOR VISIBLE WAVELENGTHS by Dan Bruton (astro@tamu.edu)
1699 % This program can be found at
1700 % http://www.physics.sfasu.edu/astro/color.html
1701 % and was last updated on February 20, 1996.
1702 % The spectrum is generated using approximate RGB values for visible
1703 % wavelengths between 380 nm and 780 nm.
1704 % The red, green and blue values (RGB) are
1705 % assumed to vary linearly with wavelength (for GAMMA=1).
1706 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1707 \newdimen\wl%wavelength
1708 \newdimen\wl%i%intensity
1709 \newdimen\wl@gamma%gamma
1710 \newdimen\wlc@lorr%red (0. - 1)
1711 \newdimen\wlc@lorg%green (0. - 1)
1712 \newdimen\wlc@lorb%blue (0. - 1) % wavelength to rgb values
1713 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1714 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1715 % \wlc@color{wavelength}
1716 \def\wlc@color#1{\ignorespaces%
1717 \ifx\wl@bright\undefined\relax\def\wl@bright{100}\fi% NEW v2.1.0
1718 \wl=#1pt%
1719 \@wl@gamma=\wl@gamma pt%
1720 % compute the rgb components
1721 \ifdim\wl<10pt\relax\PackageWarning{pgf-spectra}{(#1nm) wavelength out of range ignored. The
wavelength must be greater or equal to 10nm (EUV)...}\else% NEW v2.0.0
1722 \ifdim\wl<379.99999pt\relax\else% NEW v2.0.0
1723 \ifdim\wl<440pt\wlc@lorr=440pt\advance\wlc@lorr by-\wl\divide\wlc@lorr by60\wlc@lorg=0pt
\wlc@lorb=1pt\else%
1724 \ifdim\wl<490pt\wlc@lorr=0pt\wlc@lorg=\wl\advance\wlc@lorg by-440pt\divide\wlc@lorg by50
\wlc@lorb=1pt\else%
1725 \ifdim\wl<510pt\wlc@lorr=0pt\wlc@lorg=1pt\wlc@lorb=510pt\advance\wlc@lorb by-\wl\divide
\wlc@lorb by20\else%
1726 \ifdim\wl<580pt\wlc@lorr=\wl\advance\wlc@lorr by-510pt\divide\wlc@lorr by70\wlc@lorg=1pt
\wlc@lorb=0pt\else%
1727 \ifdim\wl<645pt\wlc@lorr=1pt\wlc@lorg=645pt\advance\wlc@lorg by-\wl\divide\wlc@lorg by65
\wlc@lorb=0pt\else%
1728 \ifdim\wl<780.00001pt\wlc@lorr=1pt\wlc@lorg=0pt\wlc@lorb=0pt\else%
1729 \ifdim\wl>4000pt\relax\PackageWarning{pgf-spectra}{invalid wavelength (#1nm). The wavelength
must be lesser or equal to 4000nm (NIR)...}% NEW v2.0.0
1730 \relax%
1731 \fi\fi\fi\fi\fi\fi\fi\fi\fi\fi%
1732 % intensity correction at vision limits
1733 \ifdim\wl>700pt\ifdim\wl<780.00001pt\wl@i=780pt\advance\wl@i by-\wl\divide\wl@i by80
\multiply\wl@i by7\advance\wl@i by3pt\divide\wl@i by10\fi\else% NEW
1734 \ifdim\wl<420pt\ifdim\wl>379.99999pt\wl@i=\wl\advance\wl@i by-380pt\divide\wl@i by40
\multiply\wl@i by7\advance\wl@i by3pt\divide\wl@i by10\fi\else%
1735 \wl@i=1pt%
1736 \fi\fi%
1737 %apply intensity at vision limits correction and gamma

```

```

1738 \ifdim\wl<380pt\colorlet{wl@temp}{\wl@UVcolor}\else\ifdim\wl>780pt\colorlet{wl@temp}{
      \wl@IRcolor}\else% NEW v2.0.0
1739 \pgfmathparse{\wlc@lorr*\wl@i^\wl@gamma}\edef\wl@red{\pgfmathresult}%
1740 \pgfmathparse{\wlc@lorg*\wl@i^\wl@gamma}\edef\wl@green{\pgfmathresult}%
1741 \pgfmathparse{\wlc@lorb*\wl@i^\wl@gamma}\edef\wl@blue{\pgfmathresult}%
1742 \definecolor{wl@temp}{rgb}{\wl@red,\wl@green,\wl@blue}%
1743 \fi\fi% NEW v2.0.0
1744 \ifwl@usevisibleshade\relax% NEW v2.1.0
1745 \colorlet{wlcolor}{wl@temp!\wl@bright!\wl@backvisible}%
1746 \else%
1747 \colorlet{wlcolor}{wl@temp}%
1748 \fi%
1749 }%
1750 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1751 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1752 \def\wl@elt@search#1#2#3#4{\ignorespaces%
1753 % #1 Chemical Symbol, entered by USER
1754 % #2 Chemical Symbol to compare to, e.g. Na
1755 % #3 Emission Lines Data (or error message)
1756 % #4 lmax
1757 \edef\wl@CS@user{#1}\edef\wl@CS@comp{#2}\relax% New
1758 \ifx\wl@CS@user\wl@CS@comp\relax% New -> was \iftthenelse {...
1759 \def\wl@elt@chemsym{#2}% set chemical symbol
1760 \def\wl@elt@elemdata{#3}% set element lines data
1761 \def\wl@elt@lmax{#4}% set element lmax
1762 \fi%
1763 }%
1764 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1765 \endinput

```