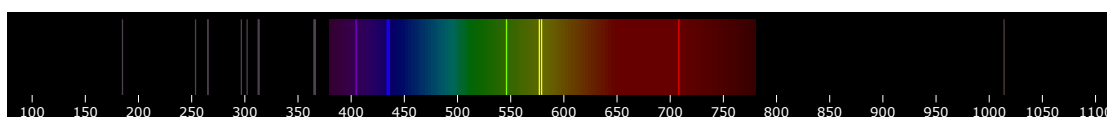


# Manual for pgf-spectra 2.1.1

Hugo Gomes  
hugo.parelho@gmail.com

28th June 2021



```
\pgfspectra[element=Hg,axis,axis step=50,begin=100,end=1100,back=visible40,gamma=.6,line width=.5pt]
```

## 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 this release – 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).

# Contents

<b>Installation and usage</b>	<b>1</b>
<b>What's new</b>	<b>2</b>
In version 2.1.1	2
In version 2.1.0	2
In version 2.0.0	3
<b>The lines data</b>	<b>4</b>
1 Data based on pst-spectra	4
2 Data based on NIST	4
<b>The commands</b>	<b>5</b>
\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[<I h>]{name}	9
\pgfspectrarainbow[<[tikz options]><(rainbow options)>]{radius}	10
<b>The options for \pgfspectra</b>	<b>11</b>
↪ 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
<b>The options for \pgfspectraplotshade</b>	<b>23</b>
~~ shade begin . . . . .	23
~~ shade end . . . . .	23
~~ shade opacity . . . . .	24
~~ shade opacity color . . . . .	24
~~ logarithmic . . . . .	25
<b>The options for \pgfspectrarainbow</b>	<b>26</b>
~~ rainbow start . . . . .	26
~~ rainbow knock out . . . . .	26
~~ rainbow fade . . . . .	27
~~ rainbow transparency . . . . .	27
~~ rainbow background . . . . .	27
<b>Examples</b>	<b>29</b>
Using \pgfspectra . . . . .	29
Using \pgfspectrashade in TikZ . . . . .	32
Using \pgfspectraplotshade and \pgfspectraplotmap with PGFPLOTS . . . . .	33
Using \pgfspectrarainbow . . . . .	50
<b>Alphabetical list of available options</b>	<b>52</b>
<b>Recommendations and known issues</b>	<b>54</b>
<b>The code</b>	<b>54</b>

## Installation and usage

pgf-spectra is placed under the terms of the L<sup>A</sup>T<sub>E</sub>X 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<sup>L</sup>A<sub>T</sub>E<sub>X</sub> 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<sup>L</sup>A<sub>T</sub>E<sub>X</sub>, 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  $10\text{ nm}$  to  $4000\text{ nm}$  (previous version from  $380\text{ nm}$  to  $780\text{ nm}$ ).
- Added the lines data outside the visible range for the 98 elements.
- No more dependency on the package `ifthen` (code rewritten with the `\ifx`  $\text{\TeX}$  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!):

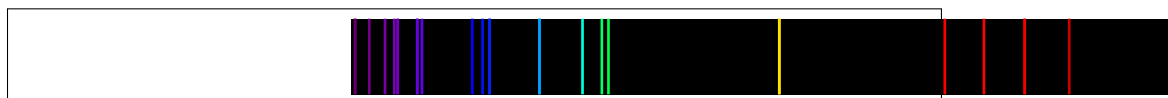


Many thanks to *Daniel García's* suggestion to solve this problem!

- Fixed the problem when putting the spectra inside any horizontal  $\text{\TeX}$  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  $10\text{ nm}$  and  $4000\text{ nm}$  (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]`
- `\wlcolor{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
```

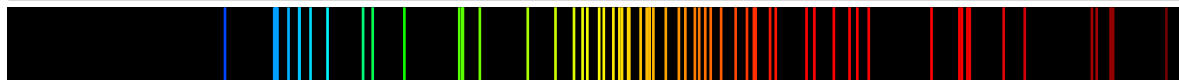


When using options, a continuous or discrete spectra in the visible region can be drawn, for instance:

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



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

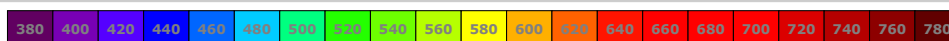


The options available for `\pgfspectra` are described in section [The options for \pgfspectra](#).

### ► Utilization of `\wlcolor{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 `\wlcolor`:

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





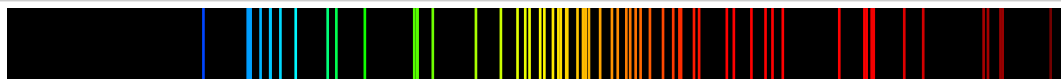
```
\tikz{\foreach \x/\y in {10/0,100/1,500/2,1000/3,2000/4,3000/5,4000/6}{\wlcolor{\x}
\draw[fill=wlcolor] (\y,0) rectangle ++(1,.5)
node[midway,font=\tiny\bfseries,text=black!50] {\x};}}
```



### ► Utilization of `\pgfspectraStyle[options]`

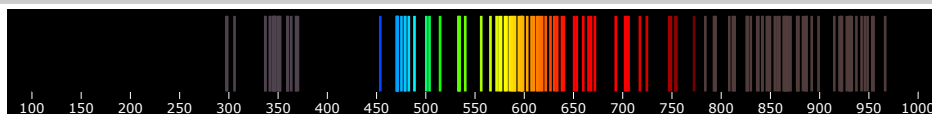
Use this command to set the global style of all the subsequent drawn spectra:

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

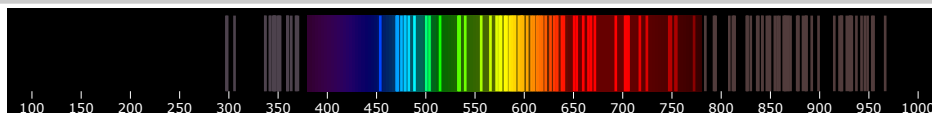


```
\pgfspectraStyle[width=.75\textwidth,axis,begin=100,end=1000,axis step=50]
```

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

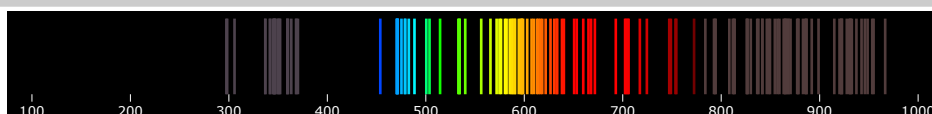


```
\pgfspectra[element=Ne,back=visible40,gamma=.6] (adding other options)
```



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

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

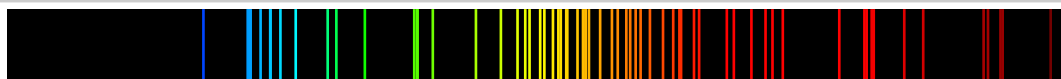


### ► Utilization of `\pgfspectraStyleReset`

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]%
}%
```

1000 K
1700 K
1850 K
2400 K
2550 K
2700 K
3000 K
3200 K
3350 K
5000 K
5500 K
6000 K
6200 K
6500 K
6600 K
6700 K
9500 K
15000 K
27000 K
40000 K

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

This commands 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 then the `'end'` wavelength and is in the visible region:  $\lambda_{\text{start}} < \lambda_{\text{end}}$  and  $380 \leq \lambda \leq 780$ .*

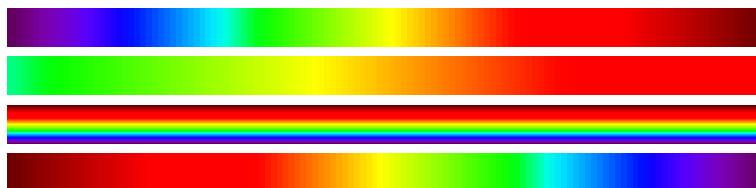
The optional parameter takes the value **h** or **v** and has the default value of **h**.

```

\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}`

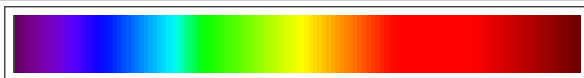
This command, without any options, builds and makes available a shading in the wavelength range from 380 nm to 780 nm to use in PGFPLOTS with the provided `name`.

```

\pgfspectraplotshade{myPlotShadeA}

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

```



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

```

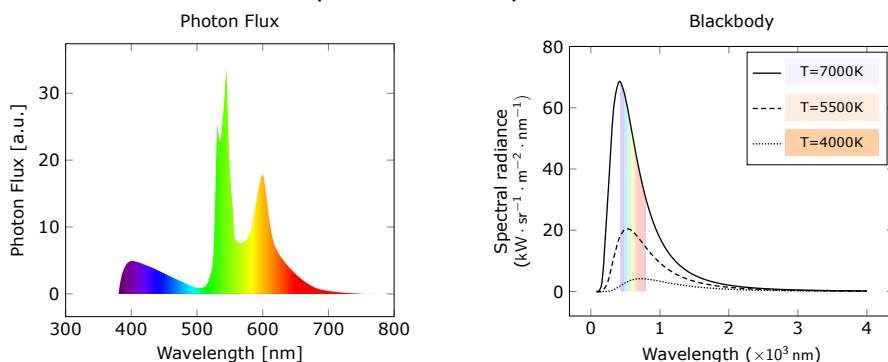
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=white,UVcolor=white,
shade opacity=.2,gamma=.6]{myPlotShadeB}

\fbbox{\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:



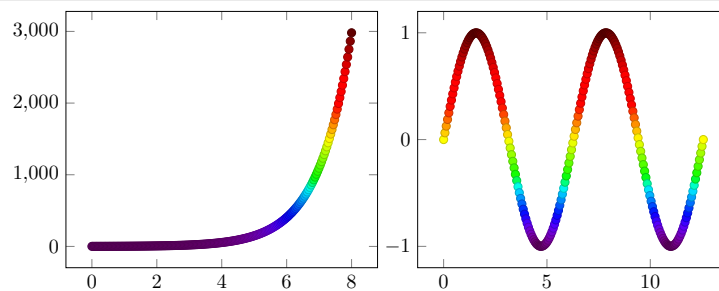
For these plots and other ones see [Using \pgfspectraplotshade and \pgfspectraplotmap with PGFPLOTS](#).

### ► 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`:

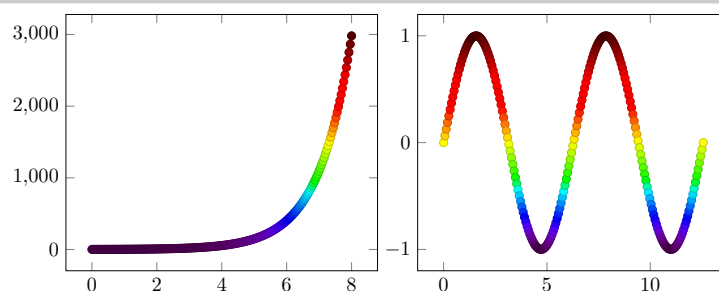
`\pgfspectraplotmap{myColorMap}` % low resolution (default value for optional parameter)

```
\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 a 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.

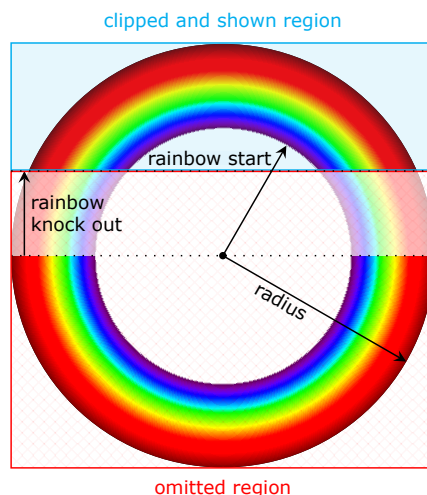
► **Utilization of `\pgfspectrarainbow<[tikz options]><(rainbow options)>\{radius\}`**

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.

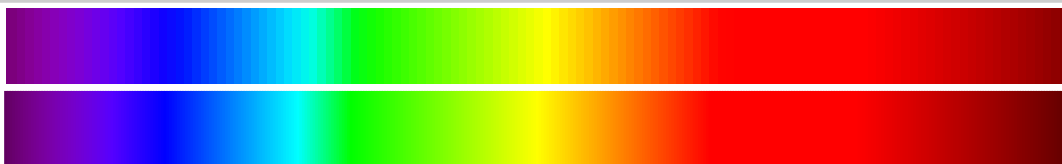
### use visible shading

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

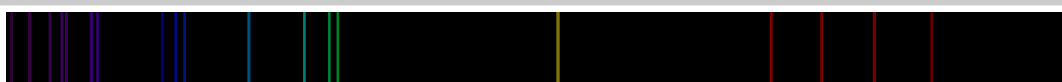
default: 100

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 ``lineintensity'`.*

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

**back**default: *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]
```

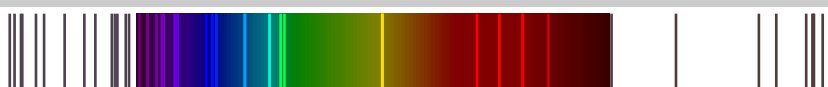
**backIRUV**default: *black*

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

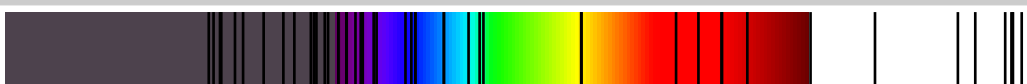
( $780nm < \lambda \leq 4000nm$ )

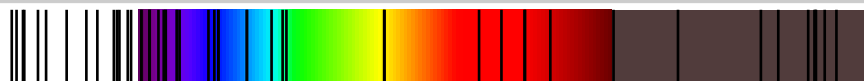
(new in v2.0.0)

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



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



**UVcolor**default: `rgb(0.3,0.2568,0.3)`Sets the color for emission lines and for background in absorption spectrum in the UV region ( $10\text{nm} \leq \lambda < 380\text{nm}$ )*(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  $[10; 4000]\text{nm}$ .*(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]`

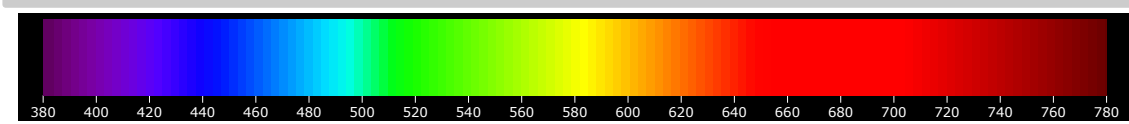
**Remark:** *it's obviously possible to set 'begin' and 'end' at the same time and if desired change the order of the wavelengths.*

`\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.

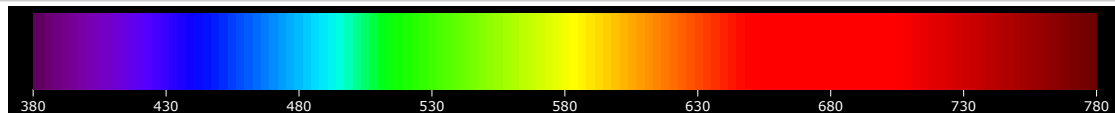
`\pgfspectra[axis]`

**axis step**

default: 20

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

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

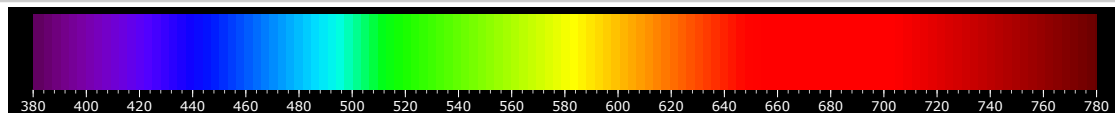
**axis ticks**

default: 0

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

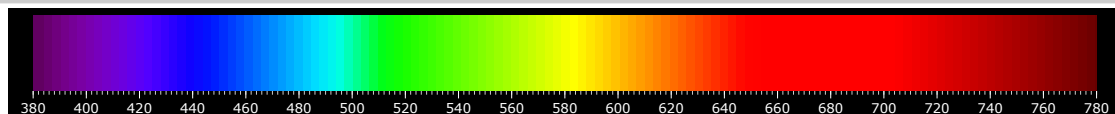
(new in v2.0.0)

```
\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

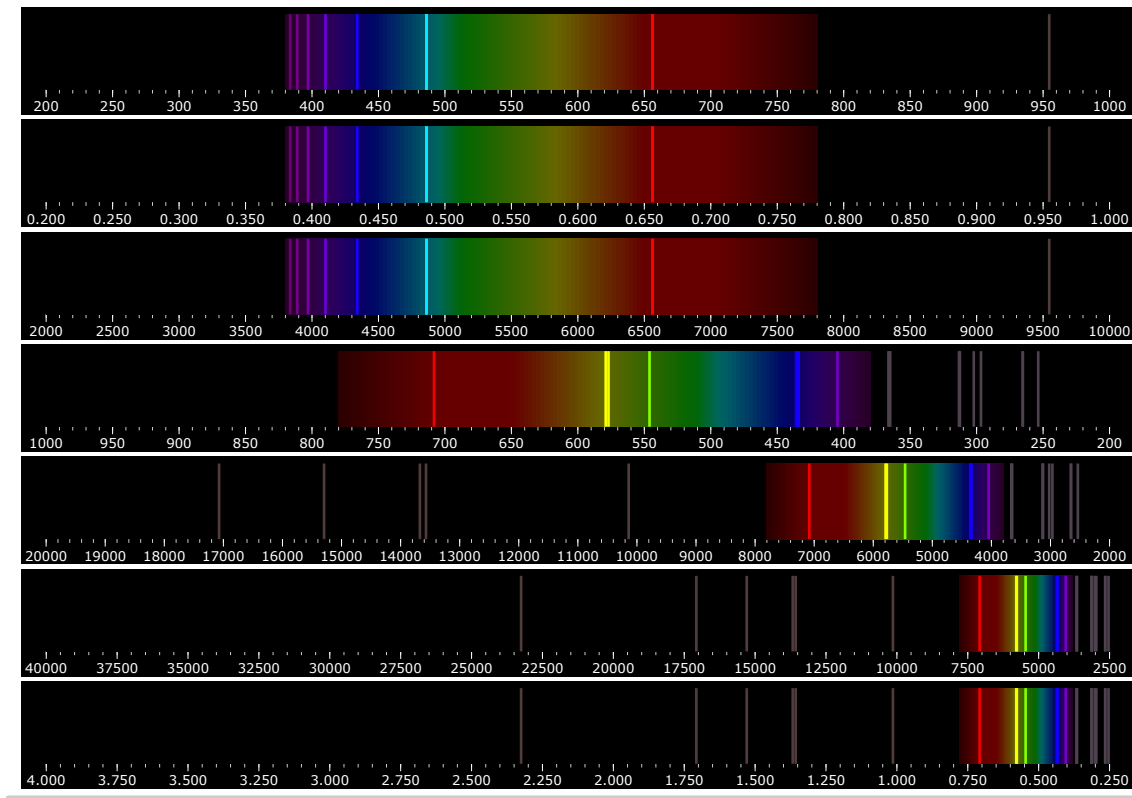
Sets the unit to use in the displayed values of wavelengths in the axis below the spectrum.

Available units are:

- nanometre (nm): `axis unit=nm`
- micrometre ( $\mu\text{m}$ ): `axis unit=micron`
- angstrom ( $\text{\AA}$ ): `axis unit=A`

(new in v2.1.0)

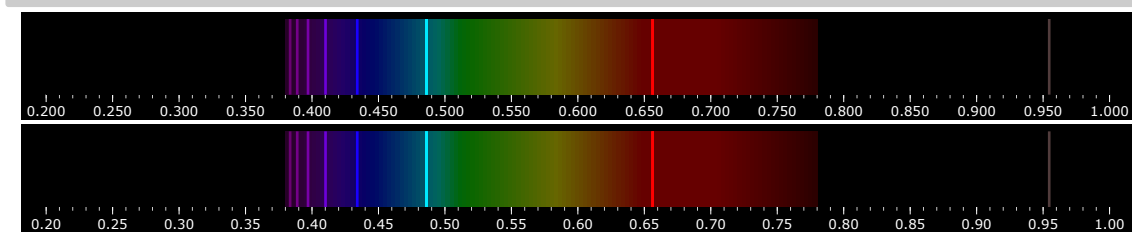
```
\pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=micron,back=visible40]
\\ \pgfspectra[element=H,begin=200,end=1000,axis,axis step=50,axis
ticks=4,axis unit=A,back=visible40]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

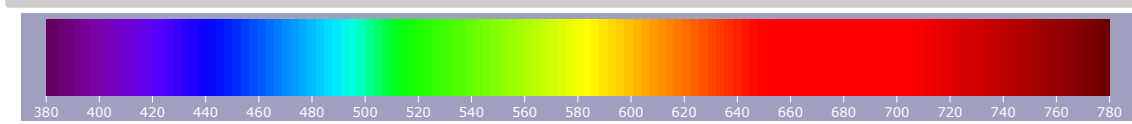
Sets the precision of the displayed wavelengths in the axis below the spectrum. *(new in v2.1.0)*

```
\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*

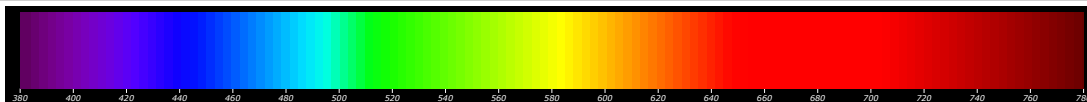
The color of the axis.

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

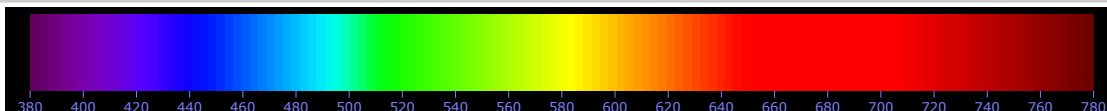


**axis font**default: `\tiny`

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 color**default: *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 text**default: `{}`

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 ( $\lambda_E$ ).

- 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  $\theta = \text{<numeric value 2>}$ .

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

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

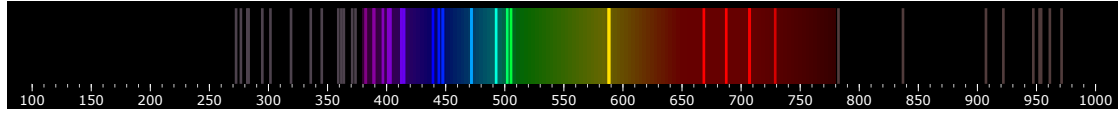
where  $\bar{v}$  is the *normalized velocity* (in units of the speed of light in vacuum,  $c$ ) of the emitter and  $\theta$  is the angle between the direction of relative motion and the direction of emission in the observer's frame (zero angle is directly away from the observer). So, if the source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



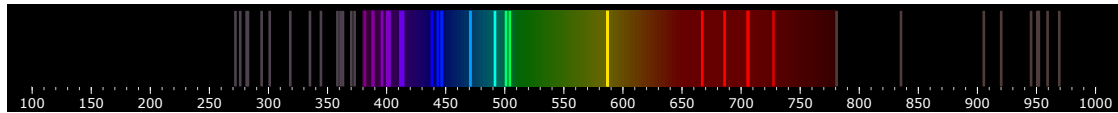
(new in v2.0.0)



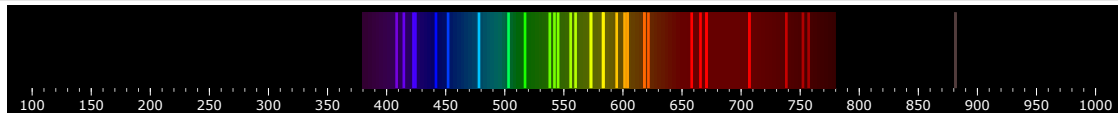
```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis
ticks=4,begin=100,end=1000,redshift={D=.001/0}]
```



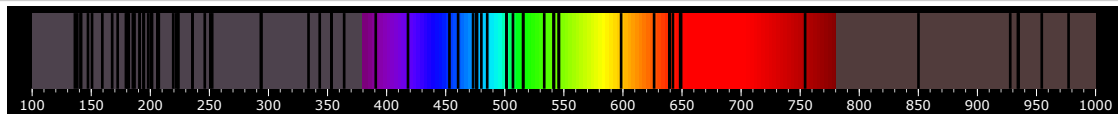
```
\pgfspectra[element=He,back=visible40,gamma=.6,axis,axis step=50,axis
ticks=4,begin=100,end=1000,redshift={D=.001/180}]
```



```
\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]
```



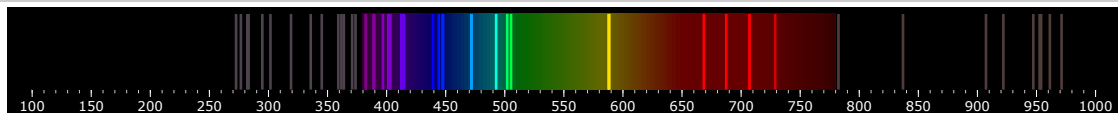
### show redshift value

default: *false*

Writes the value of the redshift (left below the spectrum).

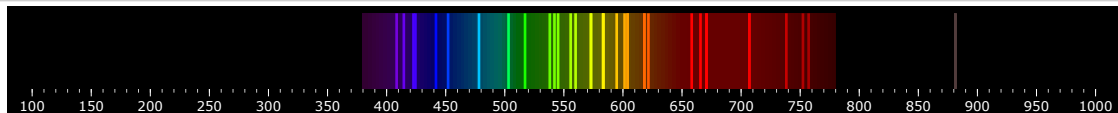
(*new in v2.0.0*)

```
\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 [PGFPLOTS package documentation](#) for more information). The range of accepted values goes from 0 nm to `(shade end-1)`. *(new in v2.1.1)*

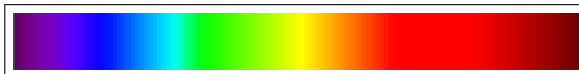
### `shade end`

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);}}
```



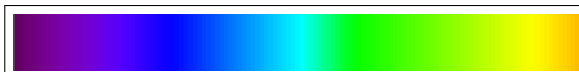
```
\pgfspectraplotshade[shade begin=600]{shadeBegin600}
```

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



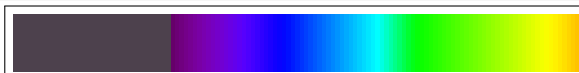
```
\pgfspectraplotshade[shade end=600]{shadeEnd600}
```

```
\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);}}
```

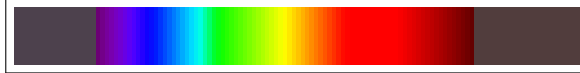


```
\pgfspectraplotshade[begin=600,shade end=900]{shade600to900}
```

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



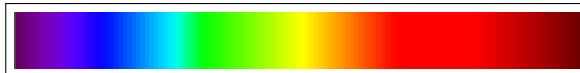
```
\pgfspectraplotshade[begin=300,shade end=900]{shade300to900}
\fbbox{\tikz{\fill[shading=shade300to900] (0,0) rectangle (7.5,.75);}}
```

**shade opacity**

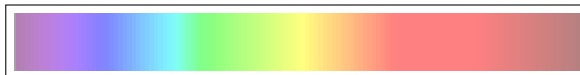
default: 1

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



```
\pgfspectraplotshade[shade opacity=.5]{shadeOpacity50}
\fbbox{\tikz{\fill[shading=shadeOpacity50] (0,0) rectangle (7.5,.75);}}
```



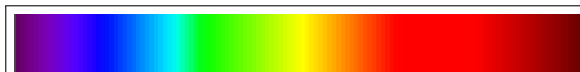
```
\pgfspectraplotshade[shade opacity=0]{shadeOpacity0}
\fbbox{\tikz{\fill[shading=shadeOpacity0] (0,0) rectangle (7.5,.75);}}
```

**shade opacity color**

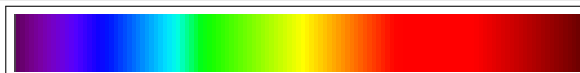
default: white

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

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



```
\pgfspectraplotshade[shade opacity color=black]{shadeOpacityBlack}
\fbbox{\tikz{\fill[shading=shadeOpacityBlack] (0,0) rectangle (7.5,.75);}}
```



```
\pgfspectraplotshade[shade opacity color=black, shade
opacity=.5]{shadeOpacityBlack50}
\fbbox{\tikz{\fill[shading=shadeOpacityBlack50] (0,0) rectangle (7.5,.75);}}
```

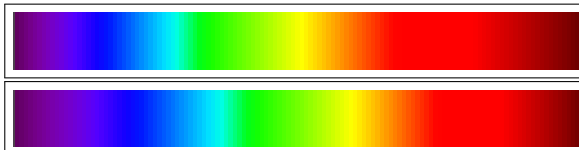


**logarithmic**default: *false*

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)

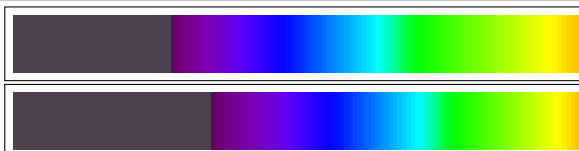
```
\pgfspectraplotshade[logarithmic]{logshadeDefault}
```

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



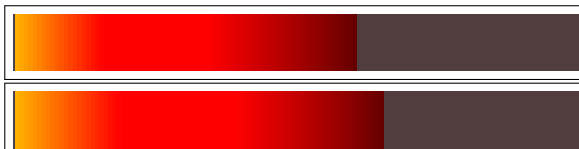
```
\pgfspectraplotshade[logarithmic,begin=300,shade end=600]{logshade300to600}
```

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



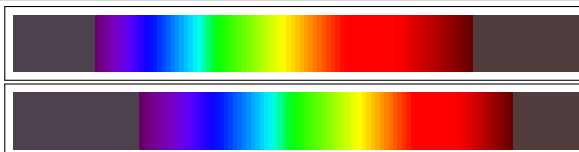
```
\pgfspectraplotshade[logarithmic,begin=600,shade end=900]{logshade600to900}
```

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



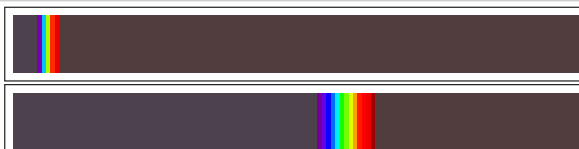
```
\pgfspectraplotshade[logarithmic,begin=300,shade end=900]{logshade300to900}
```

```
\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}
```

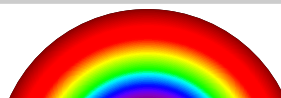


### **rainbow start**

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
```

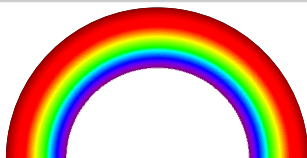


### **rainbow knock out**

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
```



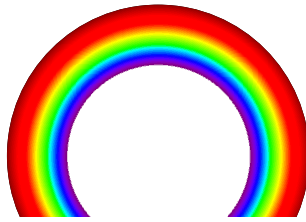
```
\pgfspectrarainbow(rainbow knock out=.4){2cm}% the default value
```



```
\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 transparency**default: `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}
```



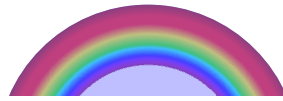
Some of the TikZ keys that affect the rainbow:

```
\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\}`.

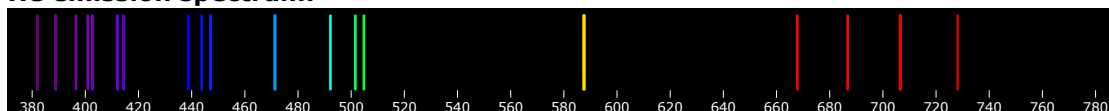
## Examples

### ► Using `\pgfspectra`

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

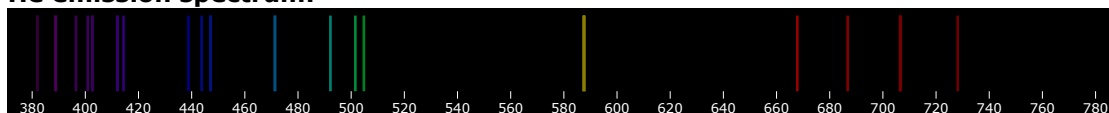
```
\pgfspectra[element=He,axis,label,label position=north west,
label after text=\ emission spectrum:]
```

**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:**

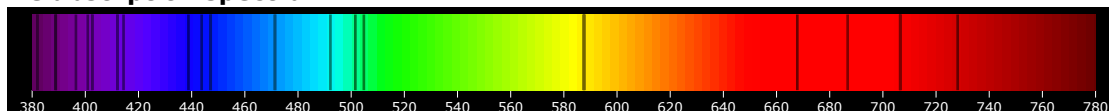


```
\pgfspectra[element=He,charge=all,line intensity=50,Imin=.05]
```



```
\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]
```



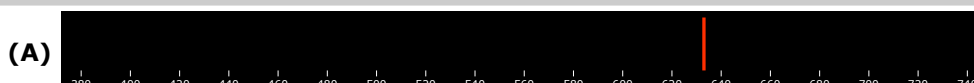
```
\pgfspectra[element=He,charge=all,relative intensity,back=visible75,gamma=2]
```



When the lines are manually inserted it's possible to use ``labelbefore text'` only with personalized text. In the next three examples ``labelbefore 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]
```





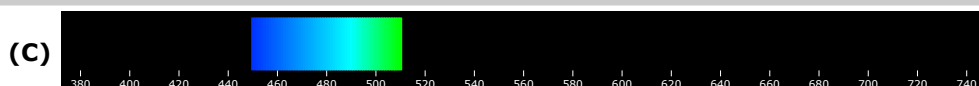
## ✓ Fluorescent lamp

```
\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]
```



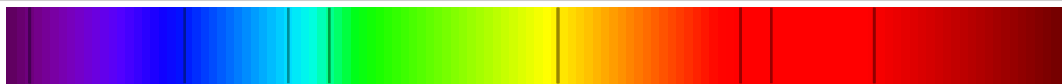
## ✓ Sun like spectrum

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



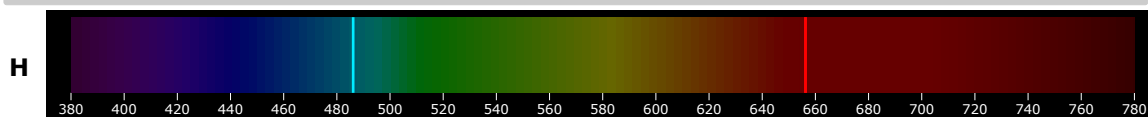
## ✓ Sirius like spectrum

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

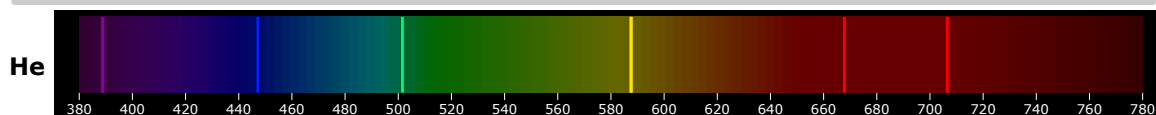


## ✓ "Classical" emission spectra of elements:

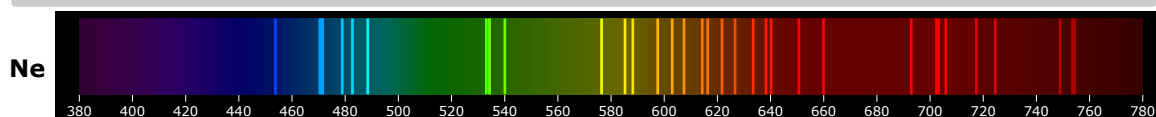
```
\pgfspectra[element=H,back=visible40,gamma=.6,label,axis,Imin=.05]
```



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

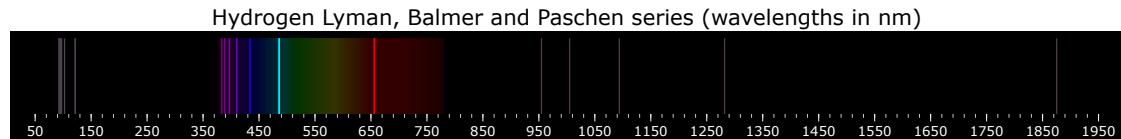


```
\pgfspectra[element=Ne,back=visible40,gamma=.6,label,axis,Imin=.05]
```



## ✓ Series of hydrogen:

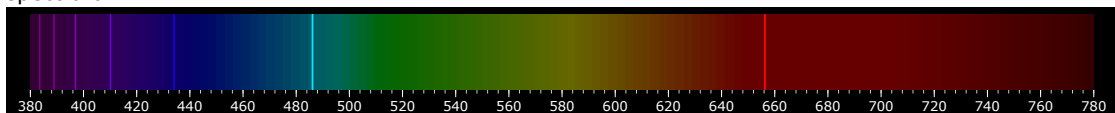
```
\pgfspectra[element=H,line width=.5pt,begin=50,end=1950,axis,axis
step=100,axis ticks=4,back=visible40,gamma=.6,brightness=.5,label,label
position=north,label font=\footnotesize,label after text={ydrogen Lyman, Balmer
and Paschen series (wavelengths in nm)}]
```



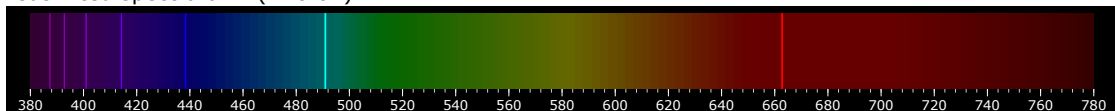
## ✓ Redshifted &amp; Blueshifted lines of hydrogen using the \foreach statement:

```
\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]
}
```

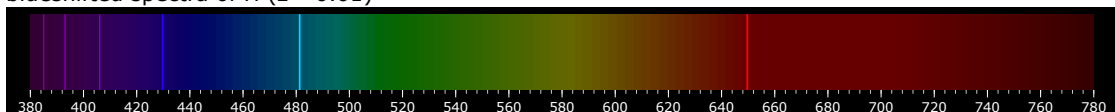
spectra of H



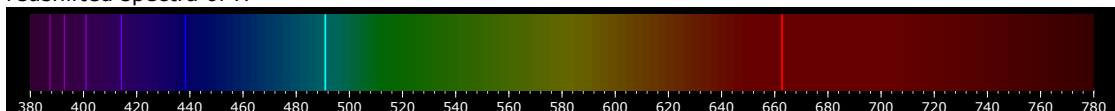
redshifted spectra of H (z=0.01)



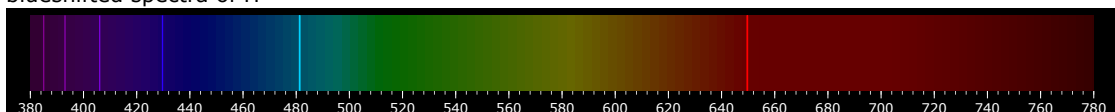
blueshifted spectra of H (z=-0.01)



redshifted spectra of H

Relativistic Doppler redshift z=0.01 ( $v=0.01c$ ;  $\theta=0^\circ$ )

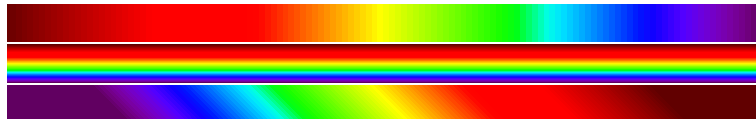
blueshifted spectra of H

Relativistic Doppler redshift z=-0.01 ( $v=0.01c$ ;  $\theta=180^\circ$ )

### ► 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
```



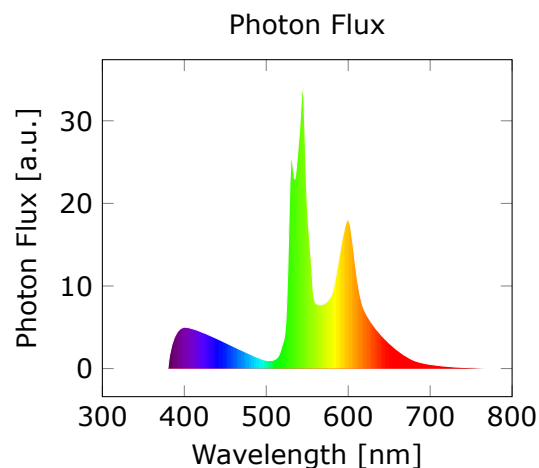
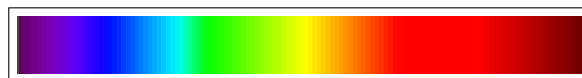
### ► Using `\pgfspectraplotshade` and `\pgfspectraplotmap` with PGFPLOTS

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);`**

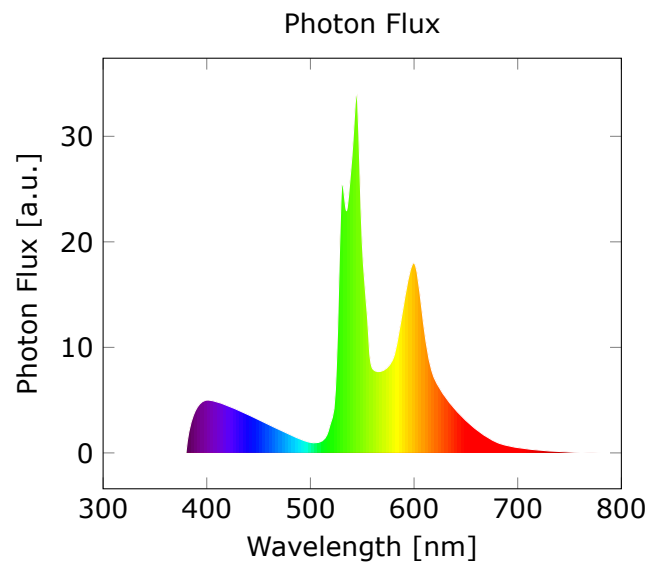
```
\pgfspectraplotshade{visiblespectrum}% default shading [380;780]nm
```

```
\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}%
```



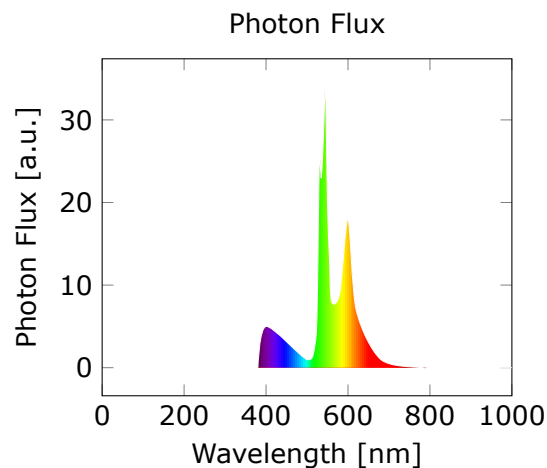
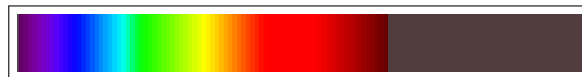
The above example could be obtained with the following *improved* code, based on a suggestion made by Stefan Pinnow:

```
\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}
```



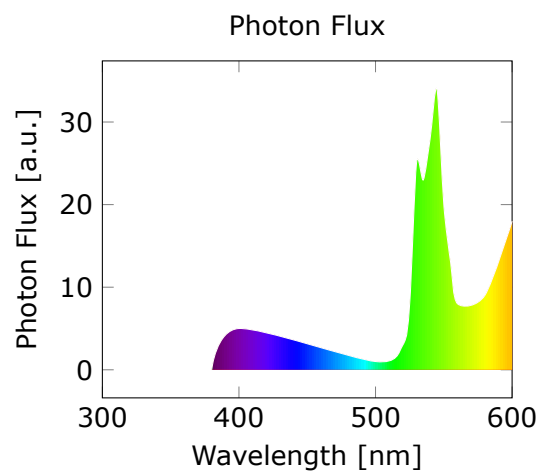
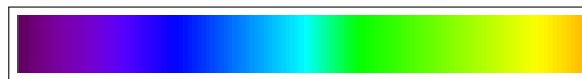
```
\pgfspectraplotshade[shade end=1000]{visiblespectrum}
```

```
\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}%
```



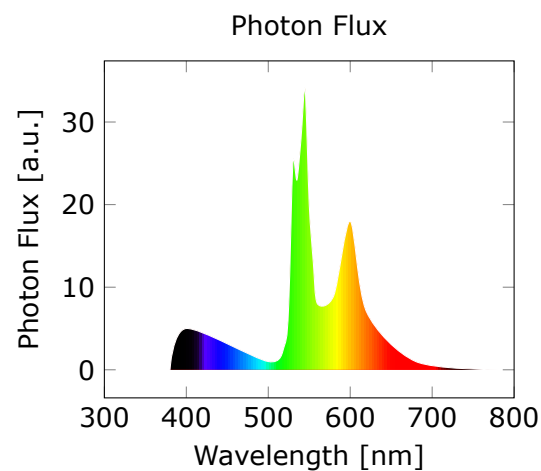
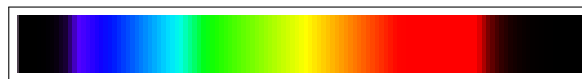
```
\pgfspectraplotshade[shade end=600]{visiblespectrum}
```

```
\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}%
```

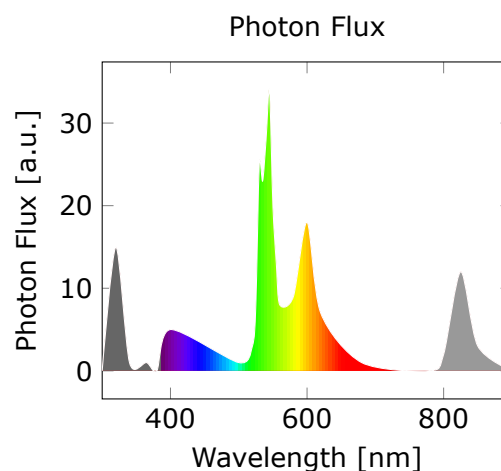
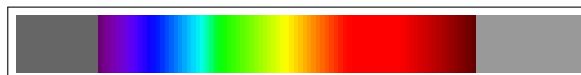




**Note:** when setting the color for IR or UV within `\pgfspectraplotshade` make sure it doesn't end with `!<number>` like `black!40`; use `black!40!white` instead.

```
\pgfspectraplotshade[IRcolor=black!40!white,UVcolor=black!60!white,shade be-
gin=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:

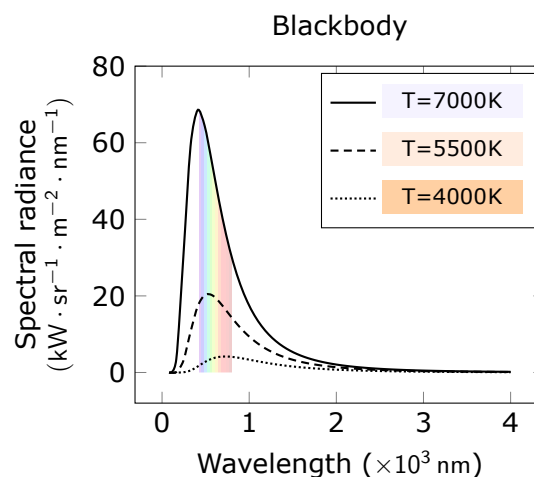
- $c = 3 \times 10^{14}$  microns  $\cdot$  s $^{-1}$  – speed of light
- $h = 6.626 \times 10^{22}$  kg  $\cdot$  microns $^2 \cdot$  s $^{-1}$  – Planck constant
- $k_B = 1.38 \times 10^{-11}$  kg  $\cdot$  microns $^2 \cdot$  s $^{-2} \cdot$  K $^{-1}$  – Boltzmann constant
- $\lambda$  – wavelength (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}};%
```

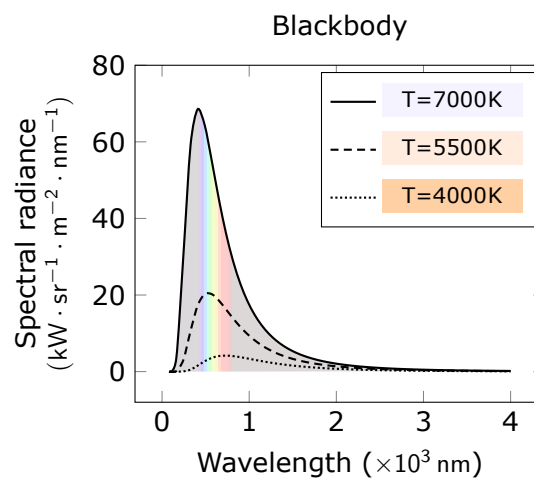
```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=white,UVcolor=white,
gamma=.6,shade opacity=.2]{BBody}
```

```
\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{(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:4,0);
\addplot+ [white,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```

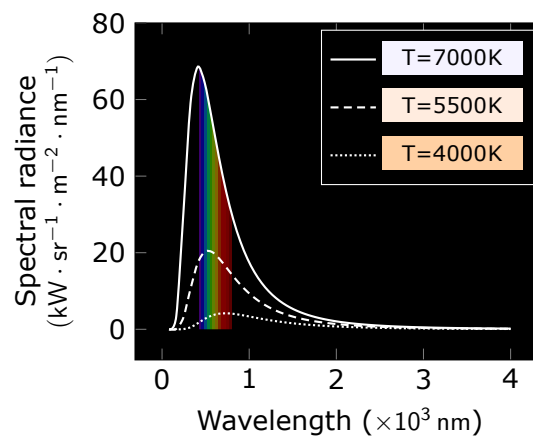


```
\pgfspectraplotshade[shade begin=0,shade end=4000,IRcolor=black,UVcolor=black,
gamma=.6,shade opacity=.5,shade opacity color=black]{BBody}
```

```
\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 ($\mathsf{\times 10^3 nm}$)},%
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:4,0);
\addplot+ [black,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```



Blackbody

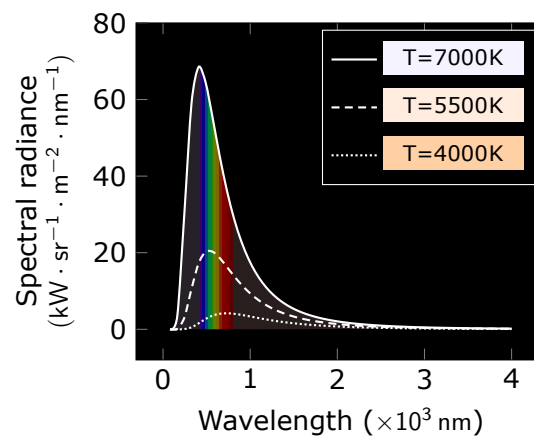


```
\pgfspectraplotshade[shade begin=0,shade end=4000,shade opacity=.5,
gamma=.6,shade opacity color=black]{BBody}
```

```
\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 ($\mathsf{\times 10^3 nm}$)},%
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:4,0);
\addplot+ [black,shading=BBody] fill between[of=spectrum and axis];
\end{axis}\end{tikzpicture}%
```

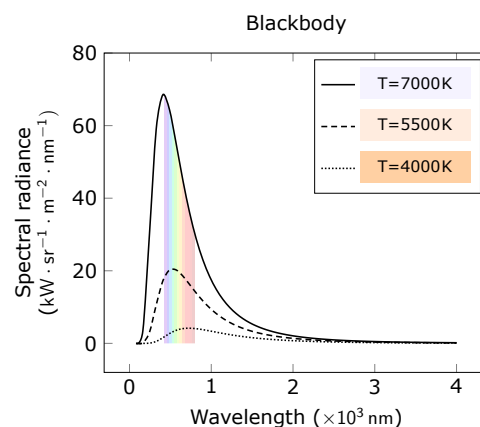


Blackbody



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)**

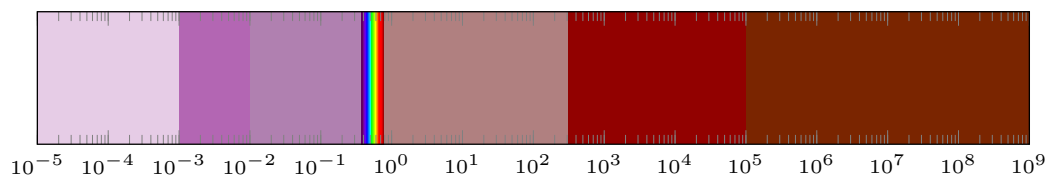
The original code lines that was replaced, in this possible answer, are commented.

```
\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(0.00000000000000bp)=(violet);
%color(8.33333333333333bp)=(blue);
%color(16.66666666666667bp)=(cyan);
%color(25.00000000000000bp)=(green);
%color(33.33333333333330bp)=(yellow);
%color(41.66666666666667bp)=(orange);
%color(50.00000000000000bp)=(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}
]
```

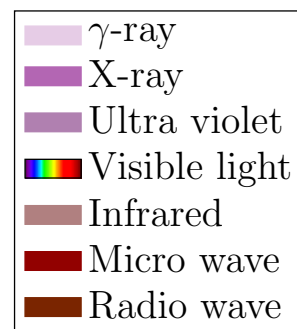
```

\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 [ $\mu\text{m}$ ]





**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.

```

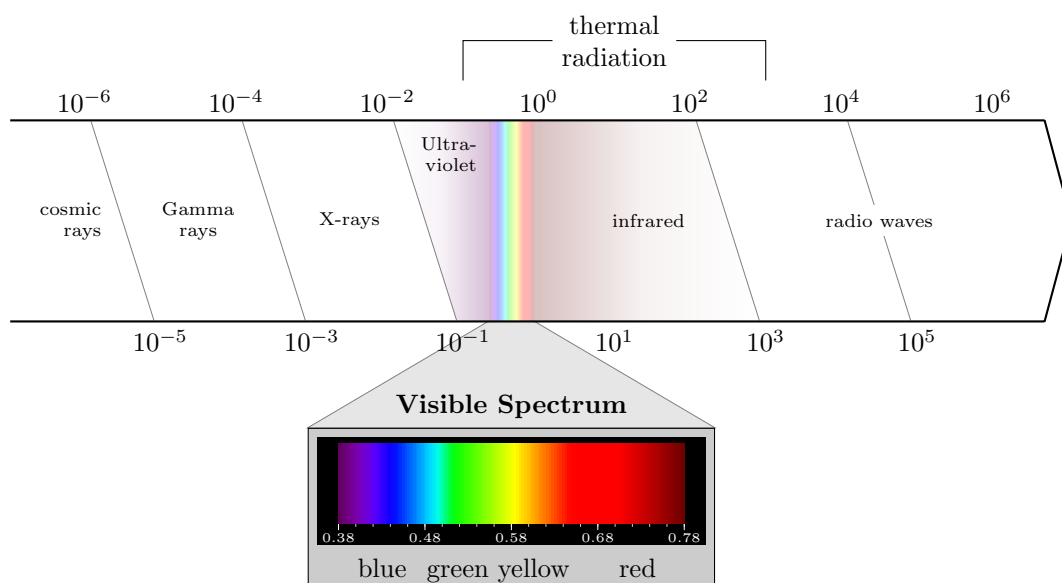
\documentclass{article}
\usepackage{tikz}
\usetikzlibrary{calc, positioning, shapes, backgrounds, fit, arrows}
\usepackage{pgf-spectra}
\usepackage{siunitx}
\usepackage{contour}
\begin{document}
\pgfdeclarehorizontalshading{visiblelight}{50bp}{%
    %https://tex.stackexchange.com/a/348492/120853
%   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
\wlcolor{380}\colorlet{UV}{wlcolor}%
\wlcolor{780}\colorlet{IR}{wlcolor}%
\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}

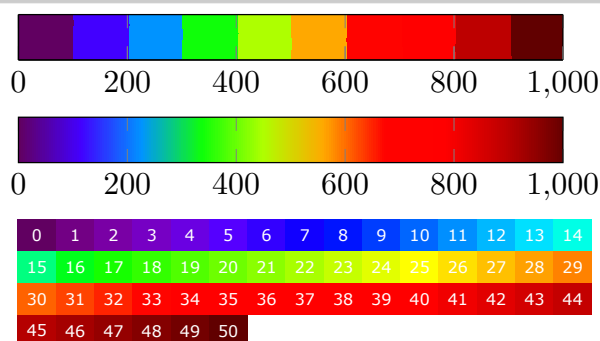
```



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

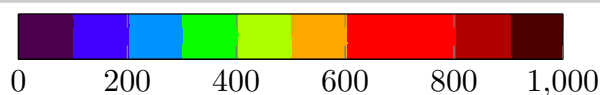
```
\pgfspectraplotmap{myColorMap}% default resolution (51 colors)
```

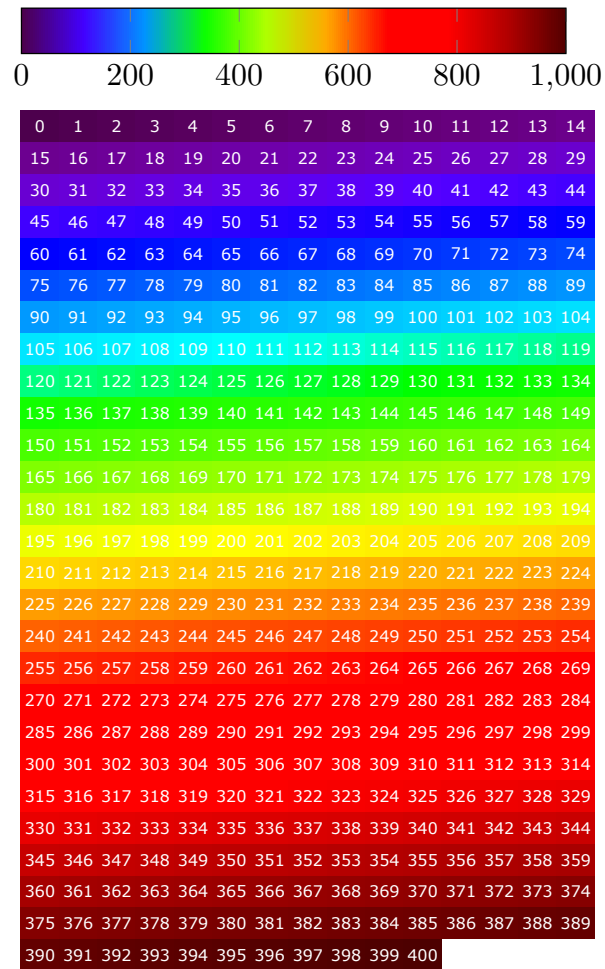
```
\pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMap)},
colorbar horizontal,colormap access=const]
\\ \pgfplotscolorbardrawstandalone[colormap={example}%
  {samples of colormap=(10 of myColorMap)},
colorbar horizontal,colormap access=map]
% --- code improved 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}
```





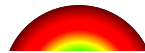
### ► Using `\pgfspectrarainbow`

Here are some examples of rainbows:

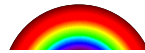
```
\pgfspectrarainbow{1cm}
```



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



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



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



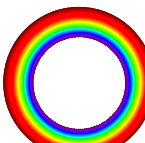
```
\pgfspectrarainbow(rainbow knock out=.8){1cm}
```



```
\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)
<b>absorption</b>	minimum intensity for the lines in the spectrum when using their relative intensities	boolean	false	{true, false}
<b>axis</b>	show or hide the axis	boolean	false	{true, false}
<b>axis color</b>	color of the axis	color	black	any named color or user defined color
<b>axis font</b>	font of the axis labels	font commands	{\tiny}	T <sub>E</sub> X font commands
<b>axis font color</b>	color of the axis labels	color	white	any named color or user defined color
<b>axis step</b>	interval in nanometres between two major axis ticks	integer	20	[0; end-begin ]nm
<b>axis ticks</b>	number of minor ticks	integer	0	{0,1,2,3,...}
<b>axis unit</b>	unit of the axis labels	text	nm	nm or micron or A
<b>axis unit precision</b>	number of significant digits (for values in nanometres) shown in axis labels	integer	3	{0,1,2,3,...}
<b>back</b>	spectrum background color	color	black	any named color or user defined color
<b>backIRUV</b>	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
<b>backVIS</b>	visible region background color in emission spectrum or emission lines color in absorption spectrum	color	black	any named color or user defined color
<b>begin</b>	first wavelength, in nanometres	integer	380	[10;4000]nm
<b>brightness</b>	brightness color correction as in the CMYK color model	decimal	1	[0;1]
<b>charge</b>	charge of the element(s)	integer	0	LSE Data: {0,1,2,3,4} NIST Data: {0,1}
<b>element</b>	chemical symbol of one element or comma sparated list of chemical symbols elements	text	NONE	H to Es except Fr
<b>end</b>	last wavelength, in nanometres	integer	780	[10;4000]nm
<b>gamma</b>	gamma color correction at the edges of the visible region	decimal	0.8	[0;1]
<b>height</b>	spectrum height	length	1cm	up to maximum T <sub>E</sub> X dimension (16384pt)
<b>Imin</b>	minimum intensity of the lines	decimal	0	[0;1]
<b>IRcolor</b>	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
<b>label</b>	show or hide the axis labels	boolean	false	{true, false}
<b>label after text</b>	extra text to place after the label of the spectrum	text	{}	
<b>label before text</b>	extra text to place before the label of the spectrum	text	{}	
<b>label font</b>	font of the spectrum label	font commands	{\bfseries\small}	
<b>label font color</b>	color of the font of the spectrum label	color	black	any named color or user defined color
<b>label position</b>	position of the label of the spectrum	text	{west}	{west, north west, north, north east, east, south east, south, south west}
<b>line intensity</b>	draw all lines with the same intensity value	integer	100	{0,1,2,...,99,100}
<b>line width</b>	width of each line drawn in the spectrum	length	1pt	up to maximum T <sub>E</sub> 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 $\text{\TeX}$ 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 $\text{\textit{TikZ}}$ 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 $\text{\TeX}$ '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  $\text{\TeX}$  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 %%%%%%%%%%%
6 \NeedsTeXFormat{LaTeX2e}%
7 \ProvidesPackage{pgf-spectra}[27/06/2021 pgf-spectra v2.1.1]%
8 \RequirePackage{tikz}%
9 \DeclareOption{LSE}{\input{spectra.data.LSE.tex}}%
10 \DeclareOption{NIST}{\input{spectra.data.NIST.tex}}%
11 \ExecuteOptions{NIST}%
12 \ProcessOptions\relax%
13 %%%%%%%%%%%
14 \definecolor{wliRcolor}{rgb}{.3157,.2373,.2373}% NEW v2.0.0
15 \definecolor{wliUVcolor}{rgb}{.3,.2568,.3}% NEW v2.0.0
16 %
17 \newcount\wl@counta% tmp counter
18 \newcount\wl@countb% tmp counter
19 \newcount\wl@countc% tmp counter
20 \newcount\wl@countd% tmp counter
21 %
22 \newif\ifwl@absorption%
23 \newif\ifcur@elem@exist%
24 \newif\ifwl@drawaxis%
25 \newif\ifwl@axislabel%
26 \newif\ifwl@intensity%
27 \newif\ifwl@redshift% NEW v2.0.0
28 \newif\ifwl@RSvalue% NEW v2.0.0
29 \newif\ifwl@usevisibleshade% NEW v2.1.0
30 % defining PGF keys
31 \pgfkeys{/wl/.cd,%
32 element/.get=\wl@element,%
33 element/.store in=\wl@element,%
34 element/.default=NONE,%
35 width/.get=\wl@width,%

```

```

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

```

```

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

```

---

```

161 % ----- The main command to draw the spectra
162 % \pgfspectra[options]
163 \def\pgfspectra{\@ifnextchar[\wl@pgfspectra@withoptions{\wl@pgfspectra@nooptions}}%
164 \def\wl@pgfspectra@nooptions{\wl@pgfspectra@continuous(0.9\textwidth,1cm)}%
165 % #####
166 \def\wl@pgfspectra@continuous(#1,#2){\ignorespaces%
167 \ifwl@usevisibleshade% NEW v2.1.0
168 \pgfspectrashade(380,780){\wl@visibleshade}%
169 \tikz{\fill[shading=\wl@visibleshade] (0,0) rectangle (#1,#2);}
170 \else%
171 \begin{tikzpicture}%
172 \pgfmathparse{#1/400}\edef\xscale{\pgfmathresult}
173 \pgfmathparse{1.4*\xscale+.09*\linewidth/\wl@width}\edef\wl@linewidth{\pgfmathresult}% NEW v
174 2.0.0 {\xscale} -> {1.4*\xscale+.09*\linewidth/\wl@width}
175 \foreach \x in {380,...,780}%
176 {
177 \wlcolor{\x}%
178 \pgfmathparse{(\x-380)*\xscale}\edef\wl@currentx{\pgfmathresult pt}%
179 \draw[\wl@temp,line width=\wl@linewidth] (\wl@currentx,0) -- ++(0,#2);%
180 }%
181 \end{tikzpicture}%
182 \fi%
183 % #####
184 \def\wl@pgfspectra@withoptions[#1]{\ignorespaces%
185 % setting default values or user style
186 \ifpgfspectra@StyleIsDef\pgfspectraStyleReset\pgfspectra@DoStyle\pgfspectra@StyleIsDeftrue
187 \relax\else\pgfspectraStyleReset\relax\fi% NEW v2.0.0
188 % process options (key values)
189 \pgfkeys{/wl/.cd,#1}%
190 % axis height
191 \setbox0=\hbox{\wl@axisfont\selectfont380}\edef\wl@axis@height{\the\ht0}%
192 % process visible background (visible+opacity)
193 \wl@counta=0%
194 \wl@countb=-1%
195 \@for\@myarg:=\wl@visible@list\do{%
196 \ifx\wl@back\@myarg\wl@countb=\wl@counta\fi%
197 \advance\wl@counta by1%
198 }%
199 \ifnum\wl@countb=-1\edef\@visible@opacity{1}\else% NEW v2.0.0
200 \ifnum\wl@countb=0\let\wl@back\wl@visible\edef\@visible@opacity{.5}\else%
201 \ifnum\wl@countb>0\let\wl@back\wl@visible\pgfmathparse{.05*\wl@countb}\edef\@visible@opacity
202 {\pgfmathresult}\fi\fi\fi%
203 %
204 % -----
205 % check limits ... % NEW v2.0.0
206 \ifnum\wl@end<10\relax\def\wlde{10}\let\wl@end\wlde\fi%
207 \ifnum\wl@end>4000\relax\def\wlquatomil{4000}\let\wl@end\wlquatomil\fi%
208 \ifnum\wl@begin<10\relax\def\wlde{10}\let\wl@begin\wlde\fi%
209 \ifnum\wl@begin>4000\relax\def\wlquatomil{4000}\let\wl@begin\wlquatomil\fi%
210 %
211 % -----
212 % verifying redshift key
213 \ifx\wl@redshift\@empty\relax%
214 \wl@redshiftfalse%
215 \else%
216 \wl@processredshiftkey\wl@redshift\relax%
217 \fi%
218 %
219 % -----
220 % if no element provided draws continuous spectrum with options or user list of lines
221 \ifx\wl@element\wl@NE%no element by the user
222 \ifx\wl@elt@chemsym\undefined\else\let\wl@elt@chemsym\undefined\fi%
223 \ifx\wl@lines\@empty%no lines by the user => continuous spectrum
224 % draws the continuous spectrum width options (default or by the user)
225 \begin{tikzpicture}%

```

---

```

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

```

59 of 90



```

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

```

```

411 \@for\@myarg:=#1\do{\expandafter\wl@get@line@info\@myarg\ifx\@currentline@charge\wl@charge
    \advance\wl@countb by 1\fi}%count only if is the desired charge
412     \ifwl@intensity%
413     \@for\@myarg:=#1%
414     \do{%
415     \expandafter\wl@get@line@info\@myarg%
416     \ifx\@currentline@charge\wl@charge%add to list if is the desired charge
417         \pgfmathparse{notless(\@currentline@int,\wl@actual@int)}\edef\wl@int@result{
            \pgfmathresult}%
418         \ifnum\wl@int@result=1%
419         \pgfmathparse{\wl@relintthresh+(1-\wl@relintthresh)*\@currentline@int/#2}\edef
            \wl@intensity@to@list{\pgfmathresult}%
420         \ifwl@first\wl@addt@list{\wl@list@@}{\@currentline@wl/\wl@intensity@to@list}
            \else%
421             \wl@addt@list{\wl@list@@}{\@currentline@wl/\wl@intensity@to@list}\fi%
422             \ifwl@first\wl@firstfalse\fi%
423         \fi%
424     \advance\wl@countb by 1%
425     \fi%
426 }%END do
427 \else%
428 \@for\@myarg:=#1%
429 \do{%
430 \expandafter\wl@get@line@info\@myarg%
431 \ifx\@currentline@charge\wl@charge%add to list if is the desired charge
432     \pgfmathparse{notless(\@currentline@int,\wl@actual@int)}\edef\wl@int@result{
        \pgfmathresult}%
433     \ifnum\wl@int@result=1%
434     \ifwl@first\wl@addt@list{\wl@list@@}{\@currentline@wl}\else%
435     \wl@addt@list{\wl@list@@}{\@currentline@wl}\fi%
436     \ifwl@first\wl@firstfalse\fi%
437     \fi%
438 \advance\wl@countb by 1%
439 \fi%
440 }%END do
441 \fi%
442 \fi%
443 \fi%
444 }%
445 % add to list
446 \def\wl@addt@list#1#2{\edef\wl@list@@{\#1#2}}%
447 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
448 % internal utils
449 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
450 \def\wl@utils@draw@axis{\ignorespaces%
451     % axis unit -> NEW v2.1.0
452     \pgfkeys{/pgf/number format/.cd,fixed,precision=\wl@axisunitprecision,set
        thousands separator={},assume math mode=true}
453     \ifx\wl@axisunit\wl@axisunit@nm\relax%
454         \def\wl@axisunit@scale{1}\def\wl@axisunit@addzeros{}%
455     \else\ifx\wl@axisunit\wl@axisunit@um\relax%
456         \def\wl@axisunit@scale{1000}\def\wl@axisunit@addzeros{}\pgfkeys{/pgf/
        number format/.cd,fixed zerofill}%
457     \else\ifx\wl@axisunit\wl@axisunit@A\relax%
458         \def\wl@axisunit@scale{1}\def\wl@axisunit@addzeros{0}% probably not the
459         best solution to overcome the TeX dimension limit values (16384pt)...
        but works!
460     \fi\fi\fi%
461     \ifnum\wl@begin>\wl@end%
462     % New xshift={-2.5*\wl@axis@height} to hold bigger numbers, e.g. 2500
463     \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);%
464     % minor ticks -> NEW v2.0.0
465     \ifnum\wl@axisticks>0\relax%
466         \pgfmathparse{\wl@end+\wl@axisstep/(\wl@axisticks+1)}\pgfmathparse{
            int(\pgfmathresult)}%
467         \edef\@axis@list{\wl@end,\pgfmathresult,...,\wl@begin}%
468         \foreach \x in \@axis@list

```



```

469         {%
470         \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult
         pt}%
471         \draw[\wl@axisfontcolor!80!transparent,line width=.25pt] (
         \wl@currentx,-.375*\wl@axis@height) — ++(0,.375*\wl@axis@height
         );%
472         }%
473         \fi%
474         \pgfmathparse{\wl@end+\wl@axisstep}\pgfmathparse{int(\pgfmathresult)}%
475         \edef\@axis@list{\wl@end,\pgfmathresult,...,\wl@begin}%
476         \foreach \x in \@axis@list%
477         {%
478         \pgfmathparse{(\wl@end-\x)*\xscale}\edef\wl@currentx{\pgfmathresult pt}%
479         \draw[\wl@axisfontcolor,line width=.25pt] (\wl@currentx,-.75*
         \wl@axis@height) — ++(0,.75*\wl@axis@height);%
480         \pgfmathparse{\x/\wl@axisunit@scale}\edef\wl@xscaledvalue{\pgfmathresult}%
         NEW v2.1.0
481         \node[\wl@axisfontcolor,font=\wl@axisfont,above,inner sep=0pt] at (
         \wl@currentx,-2.25*\wl@axis@height) {\pgfmathprintnumber{\wl@xscaledvalue
         }\wl@axisunit@addzeros};%
482         }%
483         \else%
484         \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);%
485         % minor ticks → NEW v2.0.0
486         \ifnum\wl@axisticks>0\relax%
487         \pgfmathparse{\wl@begin+\wl@axisstep/(\wl@axisticks+1)}\pgfmathparse
         {int(\pgfmathresult)}%
488         \edef\@axis@list{\wl@begin,\pgfmathresult,...,\wl@end}%
489         \foreach \x in \@axis@list%
490         {%
491         \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{
         \pgfmathresult pt}%
492         \draw[\wl@axisfontcolor!80!transparent,line width=.25pt] (
         \wl@currentx,-.375*\wl@axis@height) — ++(0,.375*\wl@axis@height
         );%
493         }%
494         \fi%
495         \pgfmathparse{\wl@begin+\wl@axisstep}\pgfmathparse{int(\pgfmathresult)}%
496         \edef\@axis@list{\wl@begin,\pgfmathresult,...,\wl@end}%
497         \foreach \x in \@axis@list%
498         {%
499         \pgfmathparse{(\x-\wl@begin)*\xscale}\edef\wl@currentx{\pgfmathresult pt}%
500         \draw[\wl@axisfontcolor,line width=.25pt] (\wl@currentx,-.75*
         \wl@axis@height) — ++(0,.75*\wl@axis@height);%
501         \pgfmathparse{\x/\wl@axisunit@scale}\edef\wl@xscaledvalue{\pgfmathresult}%
         NEW v2.1.0
502         \node[\wl@axisfontcolor,font=\wl@axisfont,above,inner sep=0pt] at (
         \wl@currentx,-2.25*\wl@axis@height) {\pgfmathprintnumber{\wl@xscaledvalue
         }\wl@axisunit@addzeros};%
503         }%
504         \fi%
505     }%
506     \def\wl@utils@put@label{\ignorespaces%
507     \ifx\wl@elt@chemsym\undefined\def\wl@elt@chemsym{}\fi%
508     \wl@get@label@position%
509     \ifnum\wl@begin>\wl@end%
510     \ifcase\wl@label@position%
511     %west
512     \ifwl@drawaxis%\ifwl@axislabel%
513     \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};%
514     \else%
515     \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};%
516     \fi%

```

```

517 \or%north west
518 \ifwl@drawaxis%
519 \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};%
520 \else%
521 \node[\wl@labelfontcolor,font=\wl@labelfont,above right,inner
xsep=0pt] at (-\wl@width,\wl@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
522 \fi%
523 \or%north
524 \node[\wl@labelfontcolor,font=\wl@labelfont,above] at (-0.5*
\wl@width,\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
};%
525 \or%north east
526 \ifwl@drawaxis%
527 \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};%
528 \else%
529 \node[\wl@labelfontcolor,font=\wl@labelfont,above left,inner xsep=0
pt] at (0,\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
};%
530 \fi
531 \or%east
532 \ifwl@drawaxis%
533 \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};%
534 \else%
535 \node[\wl@labelfontcolor,font=\wl@labelfont,right] at (0,0.5*
\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
536 \fi%
537 \or%south east
538 \ifwl@drawaxis%
539 \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};%
540 \else%
541 \node[\wl@labelfontcolor,font=\wl@labelfont,below left,inner
xsep=0pt] at (0,0) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%
542 \fi%
543 \or%south
544 \ifwl@drawaxis%
545 \node[\wl@labelfontcolor,font=\wl@labelfont,below] at (-0.5*
\wl@width,-2.5*\wl@axis@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
546 \else%
547 \node[\wl@labelfontcolor,font=\wl@labelfont,below] at (-0.5*
\wl@width,0) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
548 \fi%
549 \or%south west
550 \ifwl@drawaxis%
551 \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
};%
552 \else%
553 \node[\wl@labelfontcolor,font=\wl@labelfont,below right,inner
xsep=0pt] at (-\wl@width,0) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%
554 \fi%
555 \else%
556 \ifcase \wl@label@position%
557 %west
558 \ifwl@drawaxis%
559 \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
};%
561 \else%
562 \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};%
563 \fi%
564 \or%north west
565 \ifwl@drawaxis%
566 \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};%
567 \else%
568 \node[\wl@labelfontcolor,font=\wl@labelfont,above right,inner
xsep=0pt] at (0,\wl@height) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%
569 \fi%
570 \or%north
571 \node[\wl@labelfontcolor,font=\wl@labelfont,above] at (0.5*\wl@width
,\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
572 \or%north east
573 \ifwl@drawaxis%
574 \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};%
575 \else%
576 \node[\wl@labelfontcolor,font=\wl@labelfont,above left,inner
xsep=0pt] at (\wl@width,\wl@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
577 \fi%
578 \or%east
579 \ifwl@drawaxis%
580 \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
};%
581 \else%
582 \node[\wl@labelfontcolor,font=\wl@labelfont,right] at (\wl@width
,0.5*\wl@height) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext
};%
583 \fi%
584 \or%south east
585 \ifwl@drawaxis%
586 \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
};%
587 \else%
588 \node[\wl@labelfontcolor,font=\wl@labelfont,below left,inner
xsep=0pt] at (\wl@width,0) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%
589 \fi%
590 \or%south
591 \ifwl@drawaxis%
592 \node[\wl@labelfontcolor,font=\wl@labelfont,below] at (0.5*
\wl@width,-2.5*\wl@axis@height) {\wl@labelbtext
\wl@elt@chemsym\wl@labelatext};%
593 \else%
594 \node[\wl@labelfontcolor,font=\wl@labelfont,below] at (0.5*
\wl@width,0) {\wl@labelbtext\wl@elt@chemsym\wl@labelatext};%
595 \fi%
596 \or%south west
597 \ifwl@drawaxis%
598 \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};%
599 \else%
600 \node[\wl@labelfontcolor,font=\wl@labelfont,below right,inner
xsep=0pt] at (0,0) {\wl@labelbtext\wl@elt@chemsym
\wl@labelatext};%

```

```

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

```

```

654         \else% 3
655             \draw[draw=none, fill=\wl@back] (0,0) rectangle (
656                 \wl@pointB, \wl@height);%
657         \fi% 3
658     \fi% 2
659     \fi% 1
660 \else% 0
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             \else% 2
707                 \pgfmathparse{((\wl@end-\wl@begin)*\xscale)\edef\wl@pointB{
708                     \pgfmathresult pt}%
709                 \ifx\wl@back\wl@visible% 3 visible background
710                     \pgfspectrashade(\wl@begin, \wl@end){\wl@visibleshade}%
711                     \fill[shading=\wl@visibleshade] (0,0) rectangle (
712                         \wl@pointB, \wl@height);%
713                 \else% 3
714                     \draw[draw=none, fill=\wl@back] (0,0) rectangle (
715                         \wl@pointB, \wl@height);%
716                 \fi% 3
717             \fi% 2
718         \fi% 1
719     \fi%0
720 \else% \wl@usevisibleshade (don't use shading, draw line by line...)
721     \pgfmathparse{int(#1*100)}\edef\wl@bright{\pgfmathresult}%

```

```

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

```



```

752         \wcolor{\x}%
753         \colorlet{wcolor}{wl@temp!wl@bright!wl@backvisible}%
754         CHANGED v2.1.0
755         \pgfmathparse{\wl@pointB+(\wl@begin-\x)*\xscale}\edef
756         \wl@currentx{\pgfmathresult pt}%
757         \draw[wcolor,line width=\wl@linewidth] (\wl@currentx
758         ,0) — ++(0,\wl@height);}%
759     \else% 3
760         \draw[draw=none,fill=\wl@back] (0,0) rectangle (
761         \wl@pointB,\wl@height);%
762     \fi% 3
763 \fi% 2
764 \else% 1
765 \ifnum\wl@begin<380\relax% 1
766     \pgfmathparse{(380-\wl@begin)*\xscale}\edef\wl@pointA{\pgfmathresult
767     pt}%
768     \ifnum\wl@end>780\relax% 2
769         \pgfmathparse{(780-\wl@begin)*\xscale}\edef\wl@pointB{
770         \pgfmathresult pt}%
771         \draw[draw=none,fill=\wl@background@UVcolor] (0,0) rectangle (
772         \wl@pointA,\wl@height);%
773         \draw[draw=none,fill=\wl@background@IRcolor] (\wl@pointB,0)
774         rectangle (\wl@width,\wl@height);%
775         \ifx\wl@back\wl@visible% 3 visible background
776             \foreach \x in {380,...,780}{%
777                 \wcolor{\x}%
778                 \colorlet{wcolor}{wl@temp!wl@bright!wl@backvisible}%
779                 CHANGED v2.1.0
780                 \pgfmathparse{\wl@pointB-(780-\x)*\xscale}\edef
781                 \wl@currentx{\pgfmathresult pt}%
782                 \draw[wcolor,line width=\wl@linewidth] (\wl@currentx
783                 ,0) — ++(0,\wl@height);}%
784             \else% 3
785                 \draw[draw=none,fill=\wl@back] (\wl@pointA,0) rectangle
786                 (\wl@pointB,\wl@height);%
787             \fi% 3
788         \fi% 2
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     \fi% 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}%

```

```

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

```



```

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

```

```

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

```

```

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

```

```

1030 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1031 % NEW v2.1.0 ----->
1032 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1033 % \pgfspectrashade[ type=<h|v>]( start ,end){name} ----- type: h—horizontal (default) | v—
    vertical
1034 \def\pgfspectrashade{\ignorespaces\@ifnextchar[\wl@pgfspectrashade{\wl@pgfspectrashade[h]}}%
1035 \def\wl@pgfspectrashade[#1](#2,#3)#4{\ignorespaces%
1036 \ifnum#2>#3\relax\PackageError{pgf-spectra}%
1037 {in \textbackslash pgfspectrashade the starting wavelength (#2) must be lesser then the
    finishing wavelength (#3)}%
1038 {Try typing the wavelengths in the correct order: \textbackslash pgfspectrashade(#3,#2)...}
    \else%
1039 \ifnum#3<381\relax\PackageError{pgf-spectra}{in \textbackslash pgfspectrashade the finishing
    wavelength (#3) must be at least 381...}%
1040 {Please type a finishing wavelength greater then 380...}\else%
1041 \ifnum#2<380\relax\def\wl@shade@begin{380}\else\pgfmathparse{int(#2)}\edef\wl@shade@begin{
    \pgfmathresult}\fi%
1042 \ifnum#3>780\relax\def\wl@shade@end{780}\else\pgfmathparse{int(#3)}\edef\wl@shade@end{
    \pgfmathresult}\fi%
1043 \pgfmathparse{((\wl@shade@end-\wl@shade@begin)/50)}\edef\wl@shadecolor@step{\pgfmathresult}%
1044 %\foreach \n in {1,...,51}{%
1045 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1046          21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,%
1047          39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1048 \pgfmathparse{\wl@shade@begin+(\n-1)*\wl@shadecolor@step}\edef\wl@currentwl{\pgfmathresult}%
1049 \wlcolor{\wl@currentwl}%
1050 \edef\wl@colname{wlshcol\@Roman\n}\relax\colorlet{\wl@colname}{\wlcolor}%
1051 }%
1052 \def\wl@test{#1}\def\wl@v{v}\def\wl@h{h}%
1053 \ifx\wl@test\wl@h\relax%
1054 % the horizontal pgfshading
1055 \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)}%
1056 \else\ifx\wl@test\wl@v\relax%
1057 % the vertical pgfshading
1058 \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)}%
1059 \fi\fi%
1060 \fi\fi%
1061 }%

```

---

```

1062 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1063 \newif\ifwl@logshade% NEW v2.1.1
1064 \pgfkeys{/wl/.cd,%
1065 shade begin/.get=\wl@shadebegin,% NEW v2.1.1
1066 shade begin/.store in=\wl@shadebegin,% NEW v2.1.1
1067 shade begin/.default=380,% NEW v2.1.1
1068 shade end/.get=\wl@shadeend,%
1069 shade end/.store in=\wl@shadeend,%
1070 shade end/.default=780,%
1071 shade opacity/.get=\wl@shade@opacity,%
1072 shade opacity/.store in=\wl@shade@opacity,%
1073 shade opacity/.default=1,%
1074 shade opacity color/.get=\wl@shade@opacitycolor,%
1075 shade opacity color/.store in=\wl@shade@opacitycolor,%
1076 shade opacity color/.default=white,%
1077 logarithmic/.is if=wl@logshade% NEW v2.1.1
1078 }%
1079 \pgfkeys{/wl/.cd,shade begin,shade end,shade opacity,shade opacity color,logarithmic=false}%
1080 % \pgfspectraplotshade{name} or
1081 % \pgfspectraplotshade{options}{name}
1082 % — recoded in v2.1.1 —
1083 \def\pgfspectraplotshade{\@ifnextchar[\wl@pgfspectraplotshade{\wl@pgfspectraplotshade[]}}%
1084 \def\wl@pgfspectraplotshade[#1]#2{\ignorespaces%
1085 \pgfkeys{/wl/.cd,shade begin=380,shade end=780,shade opacity=1,shade opacity color=white,
1086 logarithmic=false}%
1087 \pgfkeys{/wl/.cd,UVcolor=\wl@UVcolor,IRcolor=\wl@IRcolor,gamma=.8}%
1088 \pgfmathparse{100*\wl@shade@opacity}\edef\@wl@shade@opacity{\pgfmathresult}%
1089 %

```

---

```

1090 % verifying start & end
1091 \edef\wl@shadebegin@pt{\wl@shadebegin pt}%
1092 \edef\wl@shadeend@pt{\wl@shadeend pt}%
1093 \ifdim\wl@shadebegin@pt<0pt\relax%
1094 \PackageError{pgf-spectra}{\pgfspectraplotshade — minimum wavelength is 0nm!}{}%
1095 \else\ifdim\wl@shadeend@pt>16000pt\relax%
1096 \PackageError{pgf-spectra}{\pgfspectraplotshade — maximum wavelength is 16000nm!}{}%
1097 \else\ifdim\wl@shadeend@pt<\wl@shadebegin@pt\relax%
1098 \PackageError{pgf-spectra}{\pgfspectraplotshade — shade end must be greater then shade begin
1099 !}{}%
1100 \else\relax%
1101 \ifdim\wl@shadeend@pt<380pt\relax%
1102 \pgfdeclarehorizontalshading{#2}(100bp){color(0bp)=(\wl@UVcolor!\@wl@shade@opacity!
1103 \wl@shade@opacitycolor);color(100bp)=(\wl@UVcolor!\@wl@shade@opacity!
1104 \wl@shade@opacitycolor)}%
1105 \else%
1106 \ifdim\wl@shadebegin@pt>780pt\relax%
1107 \pgfdeclarehorizontalshading{#2}(100bp){color(0bp)=(\wl@IRcolor!\@wl@shade@opacity!
1108 \wl@shade@opacitycolor);color(100bp)=(\wl@IRcolor!\@wl@shade@opacity!
1109 \wl@shade@opacitycolor)}%
1110 \else%
1111 \pgfkeys{/pgf/number format/.cd,fixed,precision=4,set thousands separator={},assume math
1112 mode=true}%
1113 %
1114 \ifdim\wl@shadebegin@pt<380pt\relax%
1115 \ifdim\wl@shadeend@pt>780pt\relax%
1116 %

```

---

```

1117 % \wl@shadebegin<380nm & \wl@shadeend>780nm

```

---

```

1118 % Shade structure:
1119 % xShade(bp) = f(wl)
1120 % x_from_\wl@shadebegin[bp]@25bp — x_to_\wl@shadeend[bp]@75bp:
1121 % x_UV_from_\wl@shadebegin[bp]@25bp — xUV_to_379.9nm@x_end_UV[bp] —
1122 % — x_VIS_from_380nm@x_start_VIS[bp] — xVIS_to_780nm@x_end_VIS[bp]
1123 %
1124 % — x_IR_from_780.1nm@x_start_IR[bp] — — x\wl@shadeend[bp]@75bp

```

---



```

1118 %
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 %
1125
1126 \ifwl@logshade% \wl@logshadetrue
1127 \ifnum\wl@shadebegin=0\relax\edef\wl@shadebegin{0.00001}\fi%
1128 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1129 \pgfmathprintnumber to {\pgfmathresult}{\wl@shadestep}%
1130 \pgfmathparse{50/(\log 10(\wl@shadeend)-\log 10(\wl@shadebegin))}%
1131 \pgfmathprintnumber to {\pgfmathresult}{\wl@shade@logscale@factor}%
1132 \pgfmathparse{25+(\log 10(379.9)-\log 10(\wl@shadebegin))*
1133 \wl@shade@logscale@factor}%
1134 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1135 \edef\wl@X@UV{\wl@X@tmp bp}%
1136 \pgfmathparse{25+(\log 10(780.1)-\log 10(\wl@shadebegin))*
1137 \wl@shade@logscale@factor}%
1138 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1139 \edef\wl@X@IR{\wl@X@tmp bp}%
1140 \pgfmathparse{25+(\log 10(380)-\log 10(\wl@shadebegin))*
1141 \wl@shade@logscale@factor}%
1142 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1143 \edef\wl@x@start@VIS@log{\wl@X@tmp}%
1144 \@for\n:= {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1145 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1146 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1147 \pgfmathparse{372+8*\n}%
1148 \pgfmathprintnumber to {\pgfmathresult}{\wl@currentwl}%
1149 \wlcolor{\wl@currentwl}%
1150 \colorlet{\wlcolor}{\wl@temp!\wl@shade@opacity!
1151 \wl@shade@opacitycolor}%
1152 \edef\wl@colorname{\wlshcol\@Roman\n}\relax\colorlet{
1153 \wl@colorname}{\wlcolor}%
1154 \pgfmathparse{\wl@x@start@VIS@log+(\log 10(372+8*\n)-\log 10(380))*
1155 \wl@shade@logscale@factor}%
1156 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1157 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1158 }%
1159 \else% \wl@logshadefalse
1160 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1161 \edef\wl@shade@scale{\pgfmathresult}%
1162 \pgfmathparse{25+(379.9-\wl@shadebegin)/\wl@shade@scale}%
1163 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1164 \edef\wl@X@UV{\wl@X@tmp bp}%
1165 \pgfmathparse{25+(780.1-\wl@shadebegin)/\wl@shade@scale}%
1166 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1167 \edef\wl@X@IR{\wl@X@tmp bp}%
1168 \@for\n:= {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1169 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1170 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1171 \pgfmathparse{372+8*\n}% MANTER !!!
1172 \pgfmathprintnumber to {\pgfmathresult}{\wl@currentwl}%
1173 \wlcolor{\wl@currentwl}%
1174 \colorlet{\wlcolor}{\wl@temp!\wl@shade@opacity!
1175 \wl@shade@opacitycolor}%
1176 \edef\wl@colorname{\wlshcol\@Roman\n}\relax\colorlet{
1177 \wl@colorname}{\wlcolor}%
1178 \pgfmathparse{25+(372-\wl@shadebegin+8*\n)/\wl@shade@scale}%
1179 \pgfmathprintnumber to {\pgfmathresult}{\wl@X@tmp}%
1180 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1181 }%
1182 \fi% \ifwl@logshade
1183 % the horizontal shading
1184 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
1185 \wl@shade@opacity!\wl@shade@opacitycolor);color(\wl@X@UV)=(\wl@UVcolor!

```

```

1174 \wl@shade@opacity!\wl@shade@opacitycolor);%
color(\xI)=(wlshcolI);color(\xII)=(wlshcolII);color(\xIII)=(wlshcolIII);color(
\xIV)=(wlshcolIV);color(\xV)=(wlshcolV);color(\xVI)=(wlshcolVI);color(\xVII)=(
wlshcolVII);color(\xVIII)=(wlshcolVIII);color(\xIX)=(wlshcolIX);color(\xX)=(
wlshcolX);color(\xXI)=(wlshcolXI);color(\xXII)=(wlshcolXII);color(\xXIII)=(
wlshcolXIII);color(\xXIV)=(wlshcolXIV);color(\xXV)=(wlshcolXV);color(\xXVI)=(
wlshcolXVI);color(\xXVII)=(wlshcolXVII);color(\xXVIII)=(wlshcolXVIII);color(
\xXIX)=(wlshcolXIX);color(\xXX)=(wlshcolXX);color(\xXXI)=(wlshcolXXI);color(
\xXXII)=(wlshcolXXII);color(\xXXIII)=(wlshcolXXIII);color(\xXXIV)=(wlshcolXXIV
);color(\xXXV)=(wlshcolXXV);color(\xXXVI)=(wlshcolXXVI);color(\xXXVII)=(
wlshcolXXVII);color(\xXXVIII)=(wlshcolXXVIII);color(\xXXIX)=(wlshcolXXIX);
color(\xXXX)=(wlshcolXXX);color(\xXXXI)=(wlshcolXXXI);color(\xXXXII)=(
wlshcolXXXII);color(\xXXXIII)=(wlshcolXXXIII);color(\xXXXIV)=(wlshcolXXXIV);
color(\xXXXV)=(wlshcolXXXV);color(\xXXXVI)=(wlshcolXXXVI);color(\xXXXVII)=(
wlshcolXXXVII);color(\xXXXVIII)=(wlshcolXXXVIII);color(\xXXXIX)=(wlshcolXXXIX
);color(\xXL)=(wlshcolXL);color(\xXLI)=(wlshcolXLI);color(\xXLII)=(wlshcolXLII
);color(\xXLIII)=(wlshcolXLIII);color(\xXLIV)=(wlshcolXLIV);color(\xXLV)=(
wlshcolXLV);color(\xXLVI)=(wlshcolXLVI);color(\xXLVII)=(wlshcolXLVII);color(
\xXLVIII)=(wlshcolXLVIII);color(\xXLIX)=(wlshcolXLIX);color(\xL)=(wlshcolL);
color(\xLI)=(wlshcolLI);%
1175 color(\wl@X@IR)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);color
(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);%
1176 \else% \wl@shadeendt@pt>780pt
1177 %
%
% \wl@shadebegin<380nm & \wl@shadeend<=780nm
%
% Shade structure:
% 25bp — UV — \wl@X@UV — \wl@X@VIS@i — xVIS_to_ \wl@shadeend[bp]@75bp
% VIS region in shade (51 points) -> \n=1,2,...,51
%
%
1183 \pgfmathparse{int(\wl@shadeend-380)}\pgfmathprintnumberto{\pgfmathresult}{
\wl@shade@VIS@amp@int}%
1184 \ifwl@logshade% \wl@logshadetrue
1185 \ifnum\wl@shadebegin=0\relax\edef\wl@shadebegin{0.00001}\fi%
1186 \ifnum\wl@shade@VIS@amp@int>20\relax%
1187 \pgfmathparse{50/((log10(\wl@shadeend)-log10(\wl@shadebegin)))}%
1188 \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1189 \pgfmathparse{25+((log10(379.9)-log10(\wl@shadebegin))*
\wl@shade@logscale@factor)%
\pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
\edef\wl@X@UV{\wl@X@tmp bp}%
1192 \pgfmathparse{25+((log10(380)-log10(\wl@shadebegin))*
\wl@shade@logscale@factor)%
\pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
\edef\wl@X@VIS{\wl@X@tmp bp}%
1195 \pgfmathparse{((\wl@shadeend-380)/50)%
\pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
\edef\wl@step@VIS{\wl@X@tmp}%
1197 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
38,39,40,41,42,43,44,45,46,47,48,49,50,51\do{%
1201 \pgfmathparse{380+\wl@step@VIS*(\n-1)}%
1202 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1203 \wlcolor{\wl@currentwl}%
1204 \colorlet{\wlcolor}{\wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1205 \edef\wl@colorname{\wlshcol\@Roman\n}\relax\colorlet{
\wl@colorname}{\wlcolor}%
1206 \pgfmathparse{\wl@X@VIS+log10(1+(\n-1)*(\wl@step@VIS/380))*
\wl@shade@logscale@factor}% correction factor for log10 calc
1207 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1208 \pgfmathparse{\wl@X@tmp*0.99638}% correction factor for log10
calc
1209 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1210 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1211 }%

```

```

1212 \edef\xLI{75bp}% not always at 75bp... so forcing it!
1213 % the horizontal shading
1214 \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);%
1215 color(\xI)=(wlschcolI);color(\xII)=(wlschcolII);color(\xIII)=(wlschcolIII);
    color(\xIV)=(wlschcolIV);color(\xV)=(wlschcolV);color(\xVI)=(wlschcolVI);
    color(\xVII)=(wlschcolVII);color(\xVIII)=(wlschcolVIII);color(\xIX)=(
    wlschcolIX);color(\xX)=(wlschcolX);color(\xXI)=(wlschcolXI);color(\xXII)
    =(wlschcolXII);color(\xXIII)=(wlschcolXIII);color(\xXIV)=(wlschcolXIV);
    color(\xXV)=(wlschcolXV);color(\xXVI)=(wlschcolXVI);color(\xXVII)=(
    wlschcolXVII);color(\xXVIII)=(wlschcolXVIII);color(\xXIX)=(wlschcolXIX);
    color(\xXX)=(wlschcolXX);color(\xXXI)=(wlschcolXXI);color(\xXXII)=(
    wlschcolXXII);color(\xXXIII)=(wlschcolXXIII);color(\xXXIV)=(wlschcolXXIV);
    color(\xXXV)=(wlschcolXXV);color(\xXXVI)=(wlschcolXXVI);color(\xXXVII)
    =(wlschcolXXVII);color(\xXXVIII)=(wlschcolXXVIII);color(\xXXIX)=(
    wlschcolXXIX);color(\xXXX)=(wlschcolXXX);color(\xXXXI)=(wlschcolXXXI);
    color(\xXXXII)=(wlschcolXXXII);color(\xXXXIII)=(wlschcolXXXIII);color(
    \xXXXIV)=(wlschcolXXXIV);color(\xXXXV)=(wlschcolXXXV);color(\xXXXVI)=(
    wlschcolXXXVI);color(\xXXXVII)=(wlschcolXXXVII);color(\xXXXVIII)=(
    wlschcolXXXVIII);color(\xXXXIX)=(wlschcolXXXIX);color(\xXL)=(wlschcolXL);
    color(\xXLI)=(wlschcolXLI);color(\xXLII)=(wlschcolXLII);color(\xXLIII)=(
    wlschcolXLIII);color(\xXLIV)=(wlschcolXLIV);color(\xXLV)=(wlschcolXLV);
    color(\xXLVI)=(wlschcolXLVI);color(\xXLVII)=(wlschcolXLVII);color(
    \xXLVIII)=(wlschcolXLVIII);color(\xXLIX)=(wlschcolXLIX);color(\xL)=(
    wlschcolL);color(\xLI)=(wlschcolLI);%
1216 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
    color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor)}%
1217 \else% VIS@amp@int<=20%
1218 \pgfmathparse{1/(log10(\wl@shadeend)-log10(\wl@shadebegin))}%
1219 \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1220 \pgfmathparse{25+(log10(379.999)-log10(\wl@shadebegin))*50*
    \wl@shade@logscale@factor}%
1221 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1222 \ifnum\wl@shade@VIS@amp@int=0\relax%
1223 \pgfmathparse{\wl@X@tmp*0.99621-.1898}% correction factor for log10
    calc
1224 \else%
1225 \pgfmathparse{\wl@X@tmp*0.99621}% correction factor for log10 calc
1226 \fi%
1227 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1228 \edef\wl@X@UV{\wl@X@tmp bp}%
1229 \pgfmathparse{25+(log10(380)-log10(\wl@shadebegin))*50*
    \wl@shade@logscale@factor}%
1230 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1231 \ifnum\wl@shade@VIS@amp@int=0\relax%
1232 \pgfmathparse{\wl@X@tmp*0.99621-.1898}% correction factor for log10
    calc
1233 \else%
1234 \pgfmathparse{\wl@X@tmp*0.99621}% correction factor for log10 calc
1235 \fi%
1236 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1237 \edef\wl@X@VIS{\wl@X@tmp bp}%
1238 \pgfmathparse{((\wl@shadeend-380)/50)%}
1239 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1240 \edef\wl@step@VIS{\wl@X@tmp}%
1241 \@for\n:={1,6,11,16,21,26,31,36,41,46,51}\do{%
1242 \pgfmathparse{380+\wl@step@VIS*(\n-1)}%
1243 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1244 \wcolor{\wl@currentwl}%
1245 \colorlet{wcolor}{\wl@temp!\wl@shade@opacity!
    \wl@shade@opacitycolor}%
1246 \edef\wl@colname{wlschcol\@Roman\n}\relax\colorlet{
    \wl@colname}{wcolor}%
1247 \pgfmathparse{\wl@X@VIS+log10(1+(\n-1)*(\wl@step@VIS/380))*50*
    \wl@shade@logscale@factor}%
1248 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1249 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1250 }%
1251 \edef\xLI{75bp}%

```



```

1252 % the horizontal shading
1253 \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);%
1254 color(\xl)=(wlshcolI); color(\xVI)=(wlshcolVI); color(\xXI)=(wlshcolXI);
color(\xXVI)=(wlshcolXVI); color(\xXXI)=(wlshcolXXI); color(\xXXVI)=(
wlshcolXXVI); color(\xXXI)=(wlshcolXXI); color(\xXXVI)=(wlshcolXXVI)
; color(\xXLI)=(wlshcolXLI); color(\xXLVI)=(wlshcolXLVI); color(\xLI)=(
wlshcolLI);%
1255 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);%
1256 \fi%
1257 \else% \wl@logshadefalse
1258 \ifnum\wl@shade@VIS@amp@int=0\relax%
1259 \wlcolor{380}%
1260 \colorlet{wlcolor}{\wl@temp!\wl@shade@opacity!\wl@shade@opacitycolor}%
1261 \pgfmathparse{((\wl@shadeend-\wl@shadebegin)/50}%
1262 \edef\wl@shade@scale{\pgfmathresult}%
1263 \pgfmathparse{25+(379.9-\wl@shadebegin)/\wl@shade@scale-.1898}%
1264 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@UV}%
1265 \pgfmathparse{25+(379.99-\wl@shadebegin)/\wl@shade@scale-.1898}%
1266 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@VIS}%
1267 % the horizontal shading
1268 \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);%
1269 color(\wl@X@VIS)=(wlcolor); color(75bp)=(wlcolor);%
1270 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);%
1271 \else%
1272 \pgfmathparse{((\wl@shadeend-\wl@shadebegin)/50}%
1273 \edef\wl@shade@scale{\pgfmathresult}%
1274 \pgfmathparse{25+(379.99-\wl@shadebegin)/\wl@shade@scale}%
1275 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1276 \edef\wl@X@UV{\wl@X@tmp bp}%
1277 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1278 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1279 38,39,40,41,42,43,44,45,46,47,48,49,50,51\do{%
1280 \pgfmathparse{380+(\wl@shadeend-380)/50*(\n-1)}%
1281 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1282 \wlcolor{\wl@currentwl}%
1283 \colorlet{wlcolor}{\wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1284 \edef\wl@colorname{wlshcol\@Roman\n}\relax\colorlet{
\wl@colorname}{wlcolor}%
1285 \pgfmathparse{25+(380-\wl@shadebegin+(\wl@shadeend-380)/50*(\n
-1))/\wl@shade@scale}%
1286 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1287 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1288 }%
1289 % the horizontal shading
1290 \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);%
1291 color(\xl)=(wlshcolI); color(\xII)=(wlshcolII); color(\xIII)=(wlshcolIII);
color(\xIV)=(wlshcolIV); color(\xV)=(wlshcolV); color(\xVI)=(wlshcolVI);
color(\xVII)=(wlshcolVII); color(\xVIII)=(wlshcolVIII); color(\xIX)=(
wlshcolIX); color(\xX)=(wlshcolX); color(\xXI)=(wlshcolXI); color(\xXII)=(
wlshcolXII); color(\xXIII)=(wlshcolXIII); color(\xXIV)=(wlshcolXIV);
color(\xXV)=(wlshcolXV); color(\xXVI)=(wlshcolXVI); color(\xXVII)=(
wlshcolXVII); color(\xXVIII)=(wlshcolXVIII); color(\xXIX)=(wlshcolXIX);
color(\xXX)=(wlshcolXX); color(\xXXI)=(wlshcolXXI); color(\xXXII)=(
wlshcolXXII); color(\xXXIII)=(wlshcolXXIII); color(\xXXIV)=(wlshcolXXIV);
color(\xXXV)=(wlshcolXXV); color(\xXXVI)=(wlshcolXXVI); color(\xXXVII)=(
wlshcolXXVII); color(\xXXVIII)=(wlshcolXXVIII); color(\xXXIX)=(
wlshcolXXIX); color(\xXXX)=(wlshcolXXX); color(\xXXXI)=(wlshcolXXXI);
color(\xXXXII)=(wlshcolXXXII); color(\xXXXIII)=(wlshcolXXXIII); color(
\lshcolXXXIV)=(wlshcolXXXIV); color(\lshcolXXXV)=(wlshcolXXXV); color(\lshcolXXXVI)=(
wlshcolXXXVI); color(\lshcolXXXVII)=(wlshcolXXXVII); color(\lshcolXXXVIII)=(
wlshcolXXXVIII); color(\lshcolXXXIX)=(wlshcolXXXIX); color(\xXL)=(wlshcolXL);

```

```

1292     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);%
        color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
        color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor)%
1293     \fi%
1294     \fi% \ifwl@logshade
1295     \fi% \wl@shadeendt@pt>780pt
1296 \else% \wl@shadebegin@pt<380pt
1297     \ifdim \wl@shadeend@pt>780pt \relax%
1298     %


---


1299     % \wl@shadebegin>=380nm & \wl@shadeend>780nm


---


1300     % Shade structure:
1301     % xShade(bp) = f(wl)
1302     % xVIS_from_\wl@shadebegin[bp]@25bp — xVIS_to_\wl@shadeend[bp]@75bp
1303     % VIS region in shade (51 points) -> \n=1,2,...,51
1304     %


---


1305 \ifwl@logshade% \wl@logshadetrue
1306 \pgfmathparse{int(780-\wl@shadebegin)}\pgfmathprintnumberto{\pgfmathresult}{
        \wl@shade@VIS@amp@int}%
1307     \ifnum \wl@shade@VIS@amp@int>20 \relax%
1308         \pgfmathparse{50/(log10(\wl@shadeend)-log10(\wl@shadebegin))}%
1309         \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1310         \pgfmathparse{25+(log10(780.1)-log10(\wl@shadebegin))*
            \wl@shade@logscale@factor}%
1311         \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1312         \edef\wl@X@IR{\wl@X@tmp bp}%
1313         \@for \n:= {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1314             21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1315             38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1316             \pgfmathparse{\wl@shadebegin+(780-\wl@shadebegin)/50*(\n-1)}%
1317             \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1318             \wcolor{\wl@currentwl}%
1319             \colorlet{\wcolor}{\wl@temp!\wl@shade@opacity!
                \wl@shade@opacitycolor}%
1320             \edef\wl@colorname{wlscol@\Roman\n}\relax\colorlet{
                \wl@colorname}{\wcolor}%
1321             \pgfmathparse{25+(log10(\wl@shadebegin+(780-\wl@shadebegin)/50*(
                \n-1))-log10(\wl@shadebegin))*\wl@shade@logscale@factor}%
1322             \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1323             \expandafter\edef\csname x@\Roman\n\endcsname{\wl@X@tmp bp}%
1324             }%
1325     % the horizontal shading
1326     \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
        \@wl@shade@opacity!\wl@shade@opacitycolor); color(24.99bp)=(\wl@UVcolor
        !\wl@shade@opacity!\wl@shade@opacitycolor);%
1327     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);%
1328 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)}
%
1329 \else% amp<=20
1330 \pgfmathparse{1/(log10(\wl@shadeend)-log10(\wl@shadebegin))}%
1331 \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1332 \ifnum\wl@shade@VIS@amp=int=0\relax%
1333 \pgfmathparse{25+(log10(780.1)-log10(\wl@shadebegin))*50*
\wl@shade@logscale@factor}%
1334 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1335 \edef\wl@X@IR{\wl@X@tmp bp}%
1336 \else%
1337 \pgfmathparse{25+(log10(781)-log10(\wl@shadebegin))*50*
\wl@shade@logscale@factor}%
1338 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1339 \edef\wl@X@IR{\wl@X@tmp bp}%
1340 \fi%
1341 \@for\n:=1,6,11,16,21,26,31,36,41,46,51\do{%
1342 \pgfmathparse{\wl@shadebegin+(780-\wl@shadebegin)/50*(\n-1)}%
1343 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1344 \wcolor{\wl@currentwl}%
1345 \colorlet{\wcolor}{\wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1346 \edef\wl@colorname{wlscol\@Roman\n}\relax\colorlet{
\wl@colorname}{\wcolor}%
1347 \pgfmathparse{25+(log10(\wl@shadebegin+(780-\wl@shadebegin)/50*(
\n-1))-log10(\wl@shadebegin))*50*\wl@shade@logscale@factor}%
1348 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1349 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1350 }%
1351 % the horizontal shading
1352 \pgfdeclarehorizontalshading{#2}{100bp}{color(0bp)=(\wl@UVcolor!
\wl@shade@opacity!\wl@shade@opacitycolor); color(24.99bp)=(\wl@UVcolor
!\wl@shade@opacity!\wl@shade@opacitycolor);%
1353 color(\xI)=(wlscolI); color(\xVI)=(wlscolVI); color(\xXI)=(wlscolXI);
color(\xXVI)=(wlscolXVI); color(\xXXI)=(wlscolXXI); color(\xXXVI)=(
wlscolXXVI); color(\xXXXI)=(wlscolXXXI); color(\xXXXVI)=(wlscolXXXVI)
; color(\xXLI)=(wlscolXLI); color(\xXLVI)=(wlscolXLVI); color(\xLI)=(
wlscolLI);%
1354 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)}
%
1355 \fi%
1356 \else% \wl@logshadefalse
1357 \pgfmathparse{(\wl@shadeend-\wl@shadebegin)/50}%
1358 \edef\wl@shade@scale{\pgfmathresult}%
1359 \pgfmathparse{25+(780.01-\wl@shadebegin)/\wl@shade@scale}%
1360 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1361 \edef\wl@X@IR{\wl@X@tmp bp}%
1362 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1363 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1364 38,39,40,41,42,43,44,45,46,47,48,49,50,51\do{%
1365 \pgfmathparse{\wl@shadebegin+(780-\wl@shadebegin)/50*(\n-1)}%
1366 \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1367 \wcolor{\wl@currentwl}%
1368 \colorlet{\wcolor}{\wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1369 \edef\wl@colorname{wlscol\@Roman\n}\relax\colorlet{
\wl@colorname}{\wcolor}%
1370 \pgfmathparse{25+(780-\wl@shadebegin)/50*(\n-1)/\wl@shade@scale}
%
1371 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1372 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1373 }%

```

```

1374 % the horizontal shading
1375 \pgfdeclarehorizontalshading{#2}{100bp}{ color(0bp)=(\wl@UVcolor!
\wl@shade@opacity!\wl@shade@opacitycolor); color(24.99bp)=(\wl@UVcolor!
\wl@shade@opacity!\wl@shade@opacitycolor);%
1376 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);%
1377 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)}%
1378 \fi% \ifwl@logshade
1379 \else% \wl@shadeend@pt>780pt
1380 %


---


1381 % \wl@shadebegin>=380nm & \wl@shadeend<=780nm
1382 % Shade structure:
1383 % xShade(bp) = f(wl)
1384 % xVIS_from_\wl@shadebegin[bp]@25bp — xVIS_to_\wl@shadeend[bp]@75bp
1385 % VIS region in shade (51 points) -> \n=1,2,...,51
1386 %


---


1387 \pgfmathparse{((\wl@shadeend-\wl@shadebegin)/50)\pgfmathprintnumberto{
\pgfmathresult}}{\wl@shade@step}%
1388 \ifwl@logshade% \wl@logshadetrue
1389 % scale (25 to 75 bp): \wl@shade@amp=\wl@shadeend-\wl@shadebegin ;
\wl@shade@scale=\wl@shade@amp/50
1390 \pgfmathparse{\wl@shadeend-\wl@shadebegin}\edef\wl@shade@amp{\pgfmathresult}
%
1391 \pgfmathprintnumberto{\wl@shade@amp}{\wl@shade@amp@int}%
1392 \ifnum\wl@shade@amp@int>5\relax%
1393 \pgfmathparse{50/(log10(\wl@shadeend)-log10(\wl@shadebegin))}
\pgfmathprintnumberto{\pgfmathresult}}{\wl@shade@logscale@factor}%
1394 \@for\n:=1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1395 38,39,40,41,42,43,44,45,46,47,48,49,50,51\do{%
1396 \pgfmathparse{\wl@shadebegin+(\n-1)*\wl@shade@step}%
\pgfmathprintnumberto{\pgfmathresult}}{\wl@currentwl}%
1397 \wlcolor{\wl@currentwl}%
\colorlet{wlcolor}{\wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1401 \edef\wl@colname{w!shcol\@Roman\n}\relax\colorlet{
\wl@colname}{wlcolor}%
1402 \pgfmathparse{25+((log10(\wl@currentwl)-log10(\wl@shadebegin))*
\wl@shade@logscale@factor}%
\pgfmathprintnumberto{\pgfmathresult}}{\wl@X@tmp}%
1403 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1404 }%
1405 \edef\xLI{75bp}%
1406 % the horizontal shading
1407 \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)=(wlshcolI);
%
1409 color(\xI)=(wlshcolI); color(\xII)=(wlshcolII); color(\xIII)=(
    wlshcolIII); color(\xIV)=(wlshcolIV); color(\xV)=(wlshcolV); color(
    \xVI)=(wlshcolVI); color(\xVII)=(wlshcolVII); color(\xVIII)=(
    wlshcolVIII); color(\xIX)=(wlshcolIX); color(\xX)=(wlshcolX); color(
    \xXI)=(wlshcolXI); color(\xXII)=(wlshcolXII); color(\xXIII)=(
    wlshcolXIII); color(\xXIV)=(wlshcolXIV); color(\xXV)=(wlshcolXV);
    color(\xXVI)=(wlshcolXVI); color(\xXVII)=(wlshcolXVII); color(
    \xXVIII)=(wlshcolXVIII); color(\xXIX)=(wlshcolXIX); color(\xXX)=(
    wlshcolXX); color(\xXXI)=(wlshcolXXI); color(\xXXII)=(wlshcolXXII);
    color(\xXXIII)=(wlshcolXXIII); color(\xXXIV)=(wlshcolXXIV); color(
    \xXXV)=(wlshcolXXV); color(\xXXVI)=(wlshcolXXVI); color(\xXXVII)=(
    wlshcolXXVII); color(\xXXVIII)=(wlshcolXXVIII); color(\xXXIX)=(
    wlshcolXXIX); color(\xXXX)=(wlshcolXXX); color(\xXXXI)=(wlshcolXXXI);
    ; color(\xXXXII)=(wlshcolXXXII); color(\xXXXIII)=(wlshcolXXXIII);
    color(\xXXXIV)=(wlshcolXXXIV); color(\xXXXV)=(wlshcolXXXV); color(
    \xXXXVI)=(wlshcolXXXVI); color(\xXXXVII)=(wlshcolXXXVII); color(
    \xXXXVIII)=(wlshcolXXXVIII); color(\xXXXIX)=(wlshcolXXXIX); color(
    \xXL)=(wlshcolXL); color(\xXLI)=(wlshcolXLI); color(\xXLII)=(
    wlshcolXLII); color(\xXLIII)=(wlshcolXLIII); color(\xXLIV)=(
    wlshcolXLIV); color(\xXLV)=(wlshcolXLV); color(\xXLVI)=(wlshcolXLVI);
    ; color(\xXLVII)=(wlshcolXLVII); color(\xXLVIII)=(wlshcolXLVIII);
    color(\xXLIX)=(wlshcolXLIX); color(\xL)=(wlshcolL); color(\xLI)=(
    wlshcolLI);%
1410 color(75.01bp)=(\wl@IRcolor!\@wl@shade@opacity!
    \wl@shade@opacitycolor); color(100bp)=(\wl@IRcolor!
    \@wl@shade@opacity!\wl@shade@opacitycolor)}%
1411 \else% \wl@shade@amp@int<5
1412 \pgfmathparse{((log 10(\wl@shadeend)-log 10(\wl@shadebegin)))}
    \pgfmathprintnumberto{\pgfmathresult}{\wl@shade@logscale@factor}%
1413 \wlcolor{\wl@shadebegin}%
1414 \colorlet{wlcolor}{\wl@temp!\@wl@shade@opacity!\wl@shade@opacitycolor}%
1415 \colorlet{wlshcolI}{wlcolor}%
1416 \edef\xI{25bp}%
1417 \@for\n:={11,21,31,41}\do{%
1418     \pgfmathparse{\wl@shadebegin+(\n-1)*\wl@shade@step}%
1419     \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1420     \wlcolor{\wl@currentwl}%
1421     \colorlet{wlcolor}{\wl@temp!\@wl@shade@opacity!
        \wl@shade@opacitycolor}%
1422     \edef\wl@colorname{wlshcol\@Roman\n}\relax\colorlet{
        \wl@colorname}{wlcolor}%
1423     \pgfmathparse{25+(log 10(\wl@currentwl)*50-log 10(\wl@shadebegin)
        *50)/\wl@shade@logscale@factor}%
1424     \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1425     \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1426    }%
1427 \wlcolor{\wl@shadeend}%
1428 \colorlet{wlcolor}{\wl@temp!\@wl@shade@opacity!\wl@shade@opacitycolor}%
1429 \colorlet{wlshcolLI}{wlcolor}%
1430 \edef\xLI{75bp}%
1431 % the horizontal shading
1432 \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)=(wlshcolI);
    %
1433 color(\xI)=(wlshcolI); color(\xXI)=(wlshcolXI); color(\xXXI)=(
    wlshcolXXI); color(\xXXXI)=(wlshcolXXXI); color(\xXLI)=(wlshcolXLI);
    color(\xLI)=(wlshcolLI);%
1434 color(75.01bp)=(\wl@IRcolor!\@wl@shade@opacity!
    \wl@shade@opacitycolor); color(100bp)=(\wl@IRcolor!
    \@wl@shade@opacity!\wl@shade@opacitycolor)}%
1435 \fi% amp>5
1436 \else% \wl@logshadefalse
1437 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
    21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
    38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1438     \pgfmathparse{\wl@shadebegin+(\n-1)*\wl@shade@step}%
1439     \pgfmathprintnumberto{\pgfmathresult}{\wl@currentwl}%
1440
1441

```



```

1442 \wcolor{\wl@currentwl}%
1443 \colorlet{wcolor}{wl@temp!\wl@shade@opacity!
\wl@shade@opacitycolor}%
1444 \edef\wl@colname{\wlshcol\@Roman\n}\relax\colorlet{
\wl@colname}{wcolor}%
1445 \pgfmathparse{24+\n}%
1446 \pgfmathprintnumberto{\pgfmathresult}{\wl@X@tmp}%
1447 \expandafter\edef\csname x\@Roman\n\endcsname{\wl@X@tmp bp}%
1448 }%
1449 % the horizontal shading
1450 \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)=(\wlshcolI);%
1451 color(\xI)=(\wlshcolI);color(\xII)=(\wlshcolII);color(\xIII)=(\wlshcolIII);
color(\xIV)=(\wlshcolIV);color(\xV)=(\wlshcolV);color(\xVI)=(\wlshcolVI);
color(\xVII)=(\wlshcolVII);color(\xVIII)=(\wlshcolVIII);color(\xIX)=(
\wlshcolIX);color(\xX)=(\wlshcolX);color(\xXI)=(\wlshcolXI);color(\xXII)
=(\wlshcolXII);color(\xXIII)=(\wlshcolXIII);color(\xXIV)=(\wlshcolXIV);
color(\xXV)=(\wlshcolXV);color(\xXVI)=(\wlshcolXVI);color(\xXVII)=(
\wlshcolXVII);color(\xXVIII)=(\wlshcolXVIII);color(\xXIX)=(\wlshcolXIX);
color(\xXX)=(\wlshcolXX);color(\xXXI)=(\wlshcolXXI);color(\xXXII)=(
\wlshcolXXII);color(\xXXIII)=(\wlshcolXXIII);color(\xXXIV)=(\wlshcolXXIV);
color(\xXXV)=(\wlshcolXXV);color(\xXXVI)=(\wlshcolXXVI);color(\xXXVII)
=(\wlshcolXXVII);color(\xXXVIII)=(\wlshcolXXVIII);color(\xXXIX)=(
\wlshcolXXIX);color(\xXXX)=(\wlshcolXXX);color(\xXXXI)=(\wlshcolXXXI);
color(\xXXXII)=(\wlshcolXXXII);color(\xXXXIII)=(\wlshcolXXXIII);color(
\xXXXIV)=(\wlshcolXXXIV);color(\xXXXV)=(\wlshcolXXXV);color(\xXXXVI)=(
\wlshcolXXXVI);color(\xXXXVII)=(\wlshcolXXXVII);color(\xXXXVIII)=(
\wlshcolXXXVIII);color(\xXXXIX)=(\wlshcolXXXIX);color(\xXL)=(\wlshcolXL);
color(\xXLI)=(\wlshcolXLI);color(\xXLII)=(\wlshcolXLII);color(\xXLIII)=(
\wlshcolXLIII);color(\xXLIV)=(\wlshcolXLIV);color(\xXLV)=(\wlshcolXLV);
color(\xXLVI)=(\wlshcolXLVI);color(\xXLVII)=(\wlshcolXLVII);color(
\xXLVIII)=(\wlshcolXLVIII);color(\xXLIX)=(\wlshcolXLIX);color(\xL)=(
\wlshcolL);color(\xLI)=(\wlshcolLI);%
1452 color(75.01bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor);
color(100bp)=(\wl@IRcolor!\wl@shade@opacity!\wl@shade@opacitycolor)}%
1453 \fi% \ifwl@logshade
1454 \fi% \wl@shadeend@pt>780pt
1455 \fi% \wl@shadebegin@pt<380pt\relax
1456 \fi% \wl@shadeend@pt<380pt || \wl@shadebegin@pt>780pt
1457 \fi% \wl@shadeend@pt<\wl@shadebegin@pt
1458 \fi\fi% \wl<0nm && \wl > 16000 nm
1459 }%
1460 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1461 % \pgfspectraplotmap<|h>{name}
1462 % l -> low resolution (51 colors -> 380 to 780 nm ; step 8 nm) ! Default
1463 % h -> high resolution (401 colors -> 380 to 780 nm ; step 1 nm)
1464 \def\pgfspectraplotmap{\ignorespaces\ifnextchar[\wl@pgfspectraplotmap{\wl@pgfspectraplotmap
[ ]}}%
1465 \def\wl@pgfspectraplotmap[#1]#2{\ignorespaces%
1466 \def\wl@test{#1}\def\wl@l{ } \def\wl@h{h}%
1467 \ifx\wl@test\wl@l\relax%
1468 \@for\n:=\{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1469 38,39,40,41,42,43,44,45,46,47,48,49,50,51\}\do{%
1470 %\pgfmathparse{380+(\n-1)*8}\edef\wl@currentwl{\pgfmathresult}%
1471 \pgfmathparse{372+8*\n}\edef\wl@currentwl{\pgfmathresult}%
1472 \wcolor{\wl@currentwl}%
1473 \edef\wl@colname{\wlshcol\@Roman\n}\relax\colorlet{\wl@colname}{wcolor}%
1474 }%
1475 % the pgfplots colormap
1476 \pgfplotsset{/pgfplots/colormap={#2}{color(0bp)=(\wlshcolI); color(1bp)=(\wlshcolII); color(2
bp)=(\wlshcolIII); color(3bp)=(\wlshcolIV); color(4bp)=(\wlshcolV); color(5bp)=(\wlshcolVI);
color(6bp)=(\wlshcolVII); color(7bp)=(\wlshcolVIII); color(8bp)=(\wlshcolIX); color(9bp)=(
\wlshcolX); color(10bp)=(\wlshcolXI); color(11bp)=(\wlshcolXII); color(12bp)=(\wlshcolXIII);
color(13bp)=(\wlshcolXIV); color(14bp)=(\wlshcolXV); color(15bp)=(\wlshcolXVI); color(16bp)=(
\wlshcolXVII); color(17bp)=(\wlshcolXVIII); color(18bp)=(\wlshcolXIX); color(19bp)=(\wlshcolXX
); color(20bp)=(\wlshcolXXI); color(21bp)=(\wlshcolXXII); color(22bp)=(\wlshcolXXIII); color
(23bp)=(\wlshcolXXIV); color(24bp)=(\wlshcolXXV); color(25bp)=(\wlshcolXXVI); color(26bp)=(
\wlshcolXXVII); color(27bp)=(\wlshcolXXVIII); color(28bp)=(\wlshcolXXIX); color(29bp)=(

```

```

wlscolXXX); color(30bp)=(wlscolXXXI); color(31bp)=(wlscolXXXII); color(32bp)=(
wlscolXXXIII); color(33bp)=(wlscolXXXIV); color(34bp)=(wlscolXXXV); color(35bp)=(
wlscolXXXVI); color(36bp)=(wlscolXXXVII); color(37bp)=(wlscolXXXVIII); color(38bp)=(
wlscolXXXIX); color(39bp)=(wlscolXL); color(40bp)=(wlscolXLI); color(41bp)=(wlscolXLII
); color(42bp)=(wlscolXLIII); color(43bp)=(wlscolXLIV); color(44bp)=(wlscolXLV); color
(45bp)=(wlscolXLVI); color(46bp)=(wlscolXLVII); color(47bp)=(wlscolXLVIII); color(48bp)
=(wlscolXLIX); color(49bp)=(wlscolL); color(50bp)=(wlscolLI)}}%
1478 \else \ifx \w@test\w@h\relax%%
1479 \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); %
1480 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); %
1481 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); %
1482 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); %
1483 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); %
1484 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); %
1485 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); %
1486 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); %
1487 }}%
1488 \fi\fi%
1489 }%
1490 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1491 % \pgfspectrarainbow[<tikz options>](<rainbow fade>,<rainbow start>,<rainbow knock out>,<
    rainbow background>,<rainbow transparency>){radius}
1492 % tikz options -> color, opacity, scope fading
1493 % rainbow clip -> applies a scope fading in clipped region
1494 % ...
1495 \pgfkeys{/wl/.cd,%
1496 rainbow fade/.get=\wl@rainbowfade,%
1497 rainbow fade/.store in=\wl@rainbowfade,%
1498 rainbow fade/.default={},%
1499 rainbow start/.get=\wl@rainbow@start,%
1500 rainbow start/.store in=\wl@rainbow@start,%
1501 rainbow start/.default=.6,% -> 60%
1502 rainbow knock out/.get=\wl@rainbow@KO,%
1503 rainbow knock out/.store in=\wl@rainbow@KO,%
1504 rainbow knock out/.default=.4,% -> 40%
1505 rainbow background/.get=\wl@rainbowback,%
1506 rainbow background/.store in=\wl@rainbowback,%
1507 rainbow background/.default=white,%

```



```

1508 rainbow transparency/.get=\wl@rainbowtransp,%
1509 rainbow transparency/.store in=\wl@rainbowtransp,%
1510 rainbow transparency/.default=0}% → 0%
1511 %
1512 \def\pgfspectrarainbow{\ignorespaces\@ifnextchar{\pgf@spectrarainbow{\pgf@spectrarainbow[]}}
1513 %
1514 \def\pgf@spectrarainbow[#1]{\ignorespaces\@ifnextchar({\pgf@spectra@rainbow{#1}}{
1515 \pgf@spectra@rainbow{#1}())}%
1516 %
1517 \def\pgf@spectra@rainbow#1(#2)#3{\ignorespaces%
1518 \pgfkeys{/wl/.cd,rainbow fade,rainbow start,rainbow knock out,rainbow background,rainbow
1519 transparency}%
1520 \pgfkeys{/wl/.cd,#2}%
1521 \pgfmathparse{100-\wl@rainbowtransp*100}\edef\wl@rainbow@transp{\pgfmathresult}%
1522 \pgfmathparse{\wl@rainbowtransp*100}\edef\wl@rainbow@transp@w{\pgfmathresult}%
1523 \edef\wl@rainbowend{.8875cm}\pgfmathparse{\wl@rainbow@start*\wl@rainbowend/1cm}\edef
1524 \wl@rainbowstart{\pgfmathresult cm}%
1525 \pgfmathparse{\wl@rainbow@KO*#3/1cm}\edef\wl@rainbowKO{\pgfmathresult cm}%\edef
1526 \wl@rainbowKO{\wl@rainbow@KO cm}%
1527 \ifdim\wl@rainbowstart<\wl@rainbowend\relax%
1528 \ifdim\wl@rainbowstart<.0175cm\relax\edef\wl@rainbowstart{.0175cm}\fi% ensuring there is no
1529 error in radial shading
1530 \pgfkeys{/pgf/number format/.cd,fixed,precision=3,set thousands separator={},assume math
1531 mode=true}%
1532 \pgfmathparse{\wl@rainbowstart-1/50*(\wl@rainbowend-\wl@rainbowstart)}%
1533 \pgfmathprintnumberto{\pgfmathresult}{\wl@rainbowresult}%
1534 \edef\rO{\wl@rainbowresult pt}%
1535 \@for\n:={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,%
1536 21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,%
1537 38,39,40,41,42,43,44,45,46,47,48,49,50,51}\do{%
1538 %\pgfmathparse{380+(\n-1)*8}\edef\wl@currentwl{\pgfmathresult}%
1539 \pgfmathparse{372+8*\n}\edef\wl@currentwl{\pgfmathresult}%
1540 \wlcolor{\wl@currentwl}%
1541 \edef\wl@colname{\wlshcol\@Roman\n}\relax\colorlet{\wl@colname}{\wlcolor!100!transparent!
1542 \wl@rainbow@transp}%
1543 \pgfmathparse{\wl@rainbowstart+1/50*(\n-1)*(\wl@rainbowend-\wl@rainbowstart)}%
1544 \pgfmathprintnumberto{\pgfmathresult}{\wl@rainbowresult}%
1545 \expandafter\edef\csname r\@Roman\n\endcsname{\wl@rainbowresult pt}%
1546 }%
1547 \pgfdeclareradialshading{pgfspectrarainbow}{\pgfpoint{0pt}{0pt}}{%
1548 color(0cm)=(\wl@rainbowback!\wl@rainbow@transp@w!white);color(4/5*\rO)=(\wl@rainbowback!
1549 \wl@rainbow@transp@w!white!50);color(\rO)=(white);%
1550 color(\rI)=(\wlshcolI);color(\rII)=(\wlshcolII);color(\rIII)=(\wlshcolIII);color(\rIV)=(
1551 \wlshcolIV);color(\rV)=(\wlshcolV);color(\rVI)=(\wlshcolVI);color(\rVII)=(\wlshcolVII);color(
1552 \rVIII)=(\wlshcolVIII);color(\rIX)=(\wlshcolIX);color(\rX)=(\wlshcolX);%
1553 color(\rXI)=(\wlshcolXI);color(\rXII)=(\wlshcolXII);color(\rXIII)=(\wlshcolXIII);color(\rXIV)=(
1554 \wlshcolXIV);color(\rXV)=(\wlshcolXV);color(\rXVI)=(\wlshcolXVI);color(\rXVII)=(\wlshcolXVII);
1555 color(\rXVIII)=(\wlshcolXVIII);color(\rXIX)=(\wlshcolXIX);color(\rXX)=(\wlshcolXX);%
1556 color(\rXXI)=(\wlshcolXXI);color(\rXXII)=(\wlshcolXXII);color(\rXXIII)=(\wlshcolXXIII);color(
1557 \rXXIV)=(\wlshcolXXIV);color(\rXXV)=(\wlshcolXXV);color(\rXXVI)=(\wlshcolXXVI);color(\rXXVII)
1558 =( \wlshcolXXVII);color(\rXXVIII)=(\wlshcolXXVIII);color(\rXXIX)=(\wlshcolXXIX);color(\rXXX)=(
1559 \wlshcolXXX);%
1560 color(\rXXXI)=(\wlshcolXXXI);color(\rXXXII)=(\wlshcolXXXII);color(\rXXXIII)=(\wlshcolXXXIII);
1561 color(\rXXXIV)=(\wlshcolXXXIV);color(\rXXXV)=(\wlshcolXXXV);color(\rXXXVI)=(\wlshcolXXXVI);
1562 color(\rXXXVII)=(\wlshcolXXXVII);color(\rXXXVIII)=(\wlshcolXXXVIII);color(\rXXXIX)=(
1563 \wlshcolXXXIX);color(\rXL)=(\wlshcolXL);%
1564 color(\rXLI)=(\wlshcolXLI);color(\rXLII)=(\wlshcolXLII);color(\rXLIII)=(\wlshcolXLIII);color(
1565 \rXLIV)=(\wlshcolXLIV);color(\rXLV)=(\wlshcolXLV);color(\rXLVI)=(\wlshcolXLVI);color(\rXLVII)
1566 =( \wlshcolXLVII);color(\rXLVIII)=(\wlshcolXLVIII);color(\rXLIX)=(\wlshcolXLIX);color(\rL)=(
1567 \wlshcolL);color(\rLI)=(\wlshcolLI);%
1568 color(.95cm)=(\wlshcolLI)%
1569 }%
1570 \ifx\wl@rainbowfade\@empty\relax%
1571 \tikz{\clip(-#3,\wl@rainbowKO) rectangle ++(2*#3,#3-\wl@rainbowKO);%
1572 \fill[#1,shading=pgfspectrarainbow](0,0) circle(#3);}%
1573 \else%
1574 \tikz{\clip[scope fading=\wl@rainbowfade](-#3,\wl@rainbowKO) rectangle ++(2*#3,#3-
1575 \wl@rainbowKO);%
1576 \fill[shading=pgfspectrarainbow,\wl@rainbowback,#1](0,0) circle(#3);}%
1577 \fi%

```

---

```

1555 \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...}%
1556 \fi
1557 }%
1558 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1559 % \tempercolor -> Convert a temperature in Kelvin to rgb color
1560 %
1561 % The original algorithm used in this macro can be found at
1562 %
1563 % https://github.com/neilbartlett/color-temperature
1564 %
1565 % More information at...
1566 %
1567 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1568 %

```

---

```

1569 % https://tannerhelland.com/2012/09/18/convert-temperature-rgb-algorithm-code.html
1570 %
1571 % «Start with a temperature, in Kelvin, somewhere between 1000 and 40000. (Other values may
      work,
1572 % but I can't make any promises about the quality of the algorithm's estimates above 40000 K
      .)»
1573 %

```

---

```

1574 % Tanner Helland
1575 % https://www.zombieprototypes.com/?p=210

```

---

```

1576 % and the improved algorithm at...
1577 %

```

---

```

1578 % https://github.com/neilbartlett/color-temperature
1579 %
1580 % neilbartlett/color-temperature is licensed under the
1581 %
1582 % MIT License
1583 % A short and simple permissive license with conditions only requiring preservation of
      copyright and license notices.
1584 % Licensed works, modifications, and larger works may be distributed under different terms
      and without source code.
1585 %
1586 % Permissions
1587 % Commercial use, Modification, Distribution, Private use
1588 % Limitations
1589 % Liability, Warranty
1590 %
1591 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1592 % The MIT License (MIT)
1593 %
1594 % Copyright (c) 2015 Neil Bartlett
1595 %
1596 % Permission is hereby granted, free of charge, to any person obtaining a copy
1597 % 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 % The above copyright notice and this permission notice shall be included in
1604 % all copies or substantial portions of the Software.
1605 %
1606 % THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
1607 % IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
1608 % FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
1609 % AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

```

```

1610 % LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
1611 % OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
1612 % THE SOFTWARE.
1613 % © 2021 GitHub, Inc.
1614 %
1615 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1616 \definecolor{tempercolor}{rgb}{0,0,0}%
1617 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1618 % \tempercolor{Kelvin}
1619 \def\tempercolor#1{\ignorespaces%
1620 %Set Temperature = Temperature \ 100
1621 \edef\wl@temperatura{#1sp}%
1622 \ifdim\wl@temperatura>16380sp\relax% for supporting large values of T
1623 \pgfkeys{/pgf/number format/.cd,fixed,precision=2,fixed zerofill,set thousands separator={},
1624   assume math mode=true}%
1625 \pgfmathparse{#1sp/100sp*1.0026931+.002}% correction factor for the division made in sp
1626 \pgfmathprintnumber to{\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
1636   greater temperature ...}%
1637 \else\ifdim\wl@temperatura@pt>400pt\relax%
1638 \PackageError{pgf-spectra}{\tempercolor: the temperature must be at most 40000K}{Type a
1639   smaller temperature ...}%
1640 \else%
1641 %Calculate Red:
1642 \ifdim\wl@temperatura@pt>66pt\relax%
1643 \pgfmathparse{\wl@temperatura-55}\edef\wl@tempercolor@r{\pgfmathresult}%
1644 \pgfmathparse{351.976906+ 0.114207*\wl@tempercolor@r-40.253663*ln(\wl@tempercolor@r)}%
1645 \edef\wl@tempercolor@r{\pgfmathresult}%
1646 \edef\wl@tempercolor@r@pt{\pgfmathresult pt}%
1647 \ifdim\wl@tempercolor@r@pt<0pt\edef\wl@tempercolor@r{0}\relax\fi%
1648 \ifdim\wl@tempercolor@r@pt>255pt\edef\wl@tempercolor@r{255}\relax\fi%
1649 \pgfmathparse{\wl@tempercolor@r/255}\edef\wl@tempercolor@r{\pgfmathresult}%
1650 \else%
1651 \edef\wl@tempercolor@r{1}%
1652 \fi%
1653 %Calculate Green:
1654 \ifdim\wl@temperatura@pt>66pt\relax%
1655 \pgfmathparse{\wl@temperatura-50}\edef\wl@tempercolor@g{\pgfmathresult}%
1656 \pgfmathparse{325.449413+0.079435*\wl@tempercolor@g-28.085296*ln(\wl@tempercolor@g)}%
1657 \edef\wl@tempercolor@g{\pgfmathresult}%
1658 \edef\wl@tempercolor@g@pt{\pgfmathresult pt}%
1659 \ifdim\wl@tempercolor@g@pt<0pt\edef\wl@tempercolor@g{0}\relax\fi%
1660 \ifdim\wl@tempercolor@g@pt>255pt\edef\wl@tempercolor@g{255}\relax\fi%
1661 \pgfmathparse{\wl@tempercolor@g/255}\edef\wl@tempercolor@g{\pgfmathresult}%
1662 \else%
1663 \pgfmathparse{\wl@temperatura-2}\edef\wl@tempercolor@g{\pgfmathresult}%
1664 \pgfmathparse{-155.254856-0.445970*\wl@tempercolor@g+104.492162*ln(\wl@tempercolor@g)}%
1665 \edef\wl@tempercolor@g{\pgfmathresult}%
1666 \edef\wl@tempercolor@g@pt{\pgfmathresult pt}%
1667 \ifdim\wl@tempercolor@g@pt<0pt\edef\wl@tempercolor@g{0}\relax\fi%
1668 \ifdim\wl@tempercolor@g@pt>255pt\edef\wl@tempercolor@g{255}\relax\fi%
1669 \pgfmathparse{\wl@tempercolor@g/255}\edef\wl@tempercolor@g{\pgfmathresult}%
1670 \fi%
1671 %Calculate Blue:
1672 \ifdim\wl@temperatura@pt<66pt\relax%
1673 \ifdim\wl@temperatura@pt>20pt\relax%
1674 \pgfmathparse{\wl@temperatura-10}\edef\wl@tempercolor@b{\pgfmathresult}%
1675 \pgfmathparse{-254.769352+0.827410*\wl@tempercolor@b+115.679944*ln(\wl@tempercolor@b)}%
1676 \edef\wl@tempercolor@b{\pgfmathresult}%
1677 \edef\wl@tempercolor@b@pt{\pgfmathresult pt}%
1678 \ifdim\wl@tempercolor@b@pt<0pt\edef\wl@tempercolor@b{0}\relax\fi%
1679 \ifdim\wl@tempercolor@b@pt>255pt\edef\wl@tempercolor@b{255}\relax\fi%

```

```

1677 \pgfmathparse{\wl@tempercolor@b/255}\edef\wl@tempercolor@b{\pgfmathresult}%
1678 \else%
1679 \edef\wl@tempercolor@b{0}%
1680 \fi%
1681 \else%
1682 \edef\wl@tempercolor@b{1}%
1683 \fi%
1684 \definecolor{wl@@tempercolor}{rgb}{\wl@tempercolor@r,\wl@tempercolor@g,\wl@tempercolor@b}%
1685 \colorlet{tempercolor}{wl@@tempercolor}%
1686 \fi\fi% check limits 1000K<T<40000K
1687 }%
1688 % <--- NEW v2.1.0 ---
1689 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1690 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1691 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1692 %
1693 % nm2rgb convert nanometre wavelength to rgb
1694 % (380 <= Lambda <= 780 ) -> r,g,b on stack
1695 %
1696 % BASED on FORTRAN Code
1697 % RGB VALUES FOR VISIBLE WAVELENGTHS by Dan Bruton (astro@tamu.edu)
1698 % This program can be found at
1699 % http://www.physics.sfasu.edu/astro/color.html
1700 % and was last updated on February 20, 1996.
1701 % The spectrum is generated using approximate RGB values for visible
1702 % wavelengths between 380 nm and 780 nm.
1703 % The red, green and blue values (RGB) are
1704 % assumed to vary linearly with wavelength (for GAMMA=1).
1705 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1706 \newdimen\wl%wavelength
1707 \newdimen\wl%i%intensity
1708 \newdimen\wl@gamma%gamma
1709 \newdimen\wlc@lorr%red (0. - 1)
1710 \newdimen\wlc@lorg%green (0. - 1)
1711 \newdimen\wlc@lorb%blue (0. - 1) % wavelength to rgb values
1712 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1713 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1714 % \wlc@color{wavelength}
1715 \def\wlc@color#1{\ignorespaces%
1716 \ifx\wl@bright\undefined\relax\def\wl@bright{100}\fi% NEW v2.1.0
1717 \wl=#1pt%
1718 \@wl@gamma=\wl@gamma pt%
1719 % compute the rgb components
1720 \ifdim\wl<10pt\relax\PackageWarning{pgf-spectra}{(#1nm) wavelength out of range ignored. The
1721 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
1724 \wlc@lorb=1pt\else%
1725 \ifdim\wl<490pt\wlc@lorr=0pt\wlc@lorg=\wl\advance\wlc@lorg by-440pt\divide\wlc@lorg by50
1726 \wlc@lorb=1pt\else%
1727 \ifdim\wl<510pt\wlc@lorr=0pt\wlc@lorg=1pt\wlc@lorb=510pt\advance\wlc@lorb by-\wl\divide
1728 \wlc@lorb by20\else%
1729 \ifdim\wl<580pt\wlc@lorr=\wl\advance\wlc@lorr by-510pt\divide\wlc@lorr by70\wlc@lorg=1pt
1730 \wlc@lorb=0pt\else%
1731 \ifdim\wl<645pt\wlc@lorr=1pt\wlc@lorg=645pt\advance\wlc@lorg by-\wl\divide\wlc@lorg by65
1732 \wlc@lorb=0pt\else%
1733 \ifdim\wl<780.00001pt\wlc@lorr=1pt\wlc@lorg=0pt\wlc@lorb=0pt\else%
1734 \ifdim\wl>4000pt\relax\PackageWarning{pgf-spectra}{invalid wavelength (#1nm). The wavelength
1735 must be lesser or equal to 4000nm (NIR)...}% NEW v2.0.0
1736 \relax%
1737 \fi\fi\fi\fi\fi\fi\fi\fi\fi
1738 % intensity correction at vision limits
1739 \ifdim\wl>700pt\ifdim\wl<780.00001pt\wl@i=780pt\advance\wl@i by-\wl\divide\wl@i by80
1740 \multiply\wl@i by7\advance\wl@i by3pt\divide\wl@i by10\fi\else% NEW
1741 \ifdim\wl<420pt\ifdim\wl>379.99999pt\wl@i=\wl\advance\wl@i by-380pt\divide\wl@i by40
1742 \multiply\wl@i by7\advance\wl@i by3pt\divide\wl@i by10\fi\else%
1743 \wl@i=1pt%
1744 \fi\fi%
1745 %apply intensity at vision limits correction and gamma

```

```

1737 \ifdim\wl<380pt\colorlet{wl@temp}{\wl@UVcolor}\else\ifdim\wl>780pt\colorlet{wl@temp}{
1738   \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 \def\wl@elt@search#1#2#3#4{\ignorespaces%
1752 % #1 Chemical Symbol, entered by USER
1753 % #2 Chemical Symbol to compare to, e.g. Na
1754 % #3 Emission Lines Data (or error message)
1755 % #4 lmax
1756 \edef\wl@CS@user{#1}\edef\wl@CS@comp{#2}\relax% New
1757 \ifx\wl@CS@user\wl@CS@comp\relax% New -> was \iftthenelse{...
1758 \def\wl@elt@chemsym{#2}% set chemical symbol
1759 \def\wl@elt@elemdata{#3}% set element lines data
1760 \def\wl@elt@lmax{#4}% set element lmax
1761 \fi%
1762 }%
1763 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1764 \endinput

```