Package 'ComplexHeatmap'

October 8, 2015

Type Package

Title Making Complex Heatmaps

Version 1.0.0

Date 2015-3-26

Author Zuguang Gu

Maintainer Zuguang Gu <z.gu@dkfz.de>

Depends R (>= 3.1.0), grid

Imports methods, circlize (>= 0.2.3), GetoptLong, colorspace,
RColorBrewer

Suggests testthat (>= 0.3), knitr, markdown, cluster, dendextend

VignetteBuilder knitr

Description Complex heatmaps are efficient to visualize associations
between different sources of data sets and reveal potential features.
Here the ComplexHeatmap package provides a highly flexible way to arrange
multiple heatmaps and supports self-defined annotation graphics.

biocViews Software, Visualization, Sequencing

URL https://github.com/jokergoo/ComplexHeatmap

License GPL (>= 2)

Repository Bioconductor

Date/Publication 2015-3-26 00:00:00

NeedsCompilation no

R topics documented:

ComplexHeatmap-package

Making complex heatmap

Description

Making complex heatmap

Details

This package aims to provide a simple and flexible way to arrange multiple heatmaps as well as self-defining annotation graphics.

The package is implemented in an object oriented way. Components of heatmap lists are abstracted into several classes.

- [Heatmap-class](#): a single heatmap containing heatmap body, row/column names, titles, dendrograms and column annotations.
- [HeatmapList-class](#): a list of heatmaps and row annotations.
- [HeatmapAnnotation-class](#): a list of row annotations or column annotations.

There are also several internal classes:

- [SingleAnnotation-class](#): a single row annotation or column annotation.
- [ColorMapping-class](#): mapping from values to colors.

For plotting one single heatmap, please go to the documentation page of [Heatmap](#). For plotting multiple heatmaps, please go to [HeatmapList-class](#) and +.AdditiveUnit.

The vignette provides detailed explanation of how to use this package.

+.AdditiveUnit   *Add heatmaps or row annotations to a heatmap list*

Description

Add heatmaps or row annotations to a heatmap list

Usage

```
## S3 method for class 'AdditiveUnit'
x + y
```

Arguments

x   a Heatmap-class object, a HeatmapAnnotation-class object or a HeatmapList-class
                object.

y   a Heatmap-class object, a HeatmapAnnotation-class object or a HeatmapList-class
                object.

Details

It is only a shortcut function. It actually calls add_heatmap,Heatmap-method, add_heatmap,HeatmapList-method
or add_heatmap,HeatmapAnnotation-method depending on the class of the input objects.

The HeatmapAnnotation-class object to be added should only be row annotation.

Value

A HeatmapList-class object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
ht + ht
ht + ht + ht

ht_list = ht + ht
ht + ht_list

ha = HeatmapAnnotation(points = anno_points(1:12, which = "row"),
```

```
    which = "row")
ht + ha
ht_list + ha

ha + ha + ht
```

AdditiveUnit   *Constructor method for AdditiveUnit class*

Description

Constructor method for AdditiveUnit class

Usage

```
AdditiveUnit(...)
```

Arguments

...	arguments.

Details

This method is not used in the package.

Value

No value is returned.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example
NULL
```

AdditiveUnit-class   *An internal class*

Description

An internal class

Details

This class is a super class for [Heatmap-class](), [HeatmapList-class]() and [HeatmapAnnotation-class]() classes. It is only designed for + generic method.

Examples

```
# no example
NULL
```

---

add_heatmap-dispatch   *Method dispatch page for add_heatmap*

Description

Method dispatch page for add_heatmap.

Dispatch

add_heatmap can be dispatched on following classes:

- [add_heatmap,HeatmapAnnotation-method](), [HeatmapAnnotation-class]() class method

- [add_heatmap,HeatmapList-method](), [HeatmapList-class]() class method

- [add_heatmap,Heatmap-method](), [Heatmap-class]() class method

Examples

```
# no example
NULL
```

---

add_heatmap-Heatmap-method

Add heatmaps or row annotations as a heatmap list

Description

Add heatmaps or row annotations as a heatmap list

Usage

```
## S4 method for signature 'Heatmap'
add_heatmap(object, x)
```

Arguments

object   a [Heatmap-class](#) object.

x   a [Heatmap-class](#) object, a [HeatmapAnnotation-class](#) object or a [HeatmapList-class](#) object.

Details

There is a shortcut function +.Heatmap.

Value

A [HeatmapList-class](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
add_heatmap(ht, ht)

ha = HeatmapAnnotation(points = anno_points(1:12, which = "row"),
    which = "row")
add_heatmap(ht, ha)
```

add_heatmap-HeatmapAnnotation-method

Add row annotations or heatmaps as a heatmap list

Description

Add row annotations or heatmaps as a heatmap list

Usage

```
## S4 method for signature 'HeatmapAnnotation'
add_heatmap(object, x)
```

Arguments

object   a HeatmapAnnotation-class object.

x   a Heatmap-class object, a HeatmapAnnotation-class object or a HeatmapList-class
                object.

Details

There is a shortcut function +.HeatmapAnnotation.

Value

A HeatmapList-class object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)

ha = HeatmapAnnotation(points = anno_points(1:12, which = "row"),
    which = "row")
add_heatmap(ha, ht)
```

add_heatmap-HeatmapList-method

Add heatmaps and row annotations to the heatmap list

Description

Add heatmaps and row annotations to the heatmap list

Usage

```
## S4 method for signature 'HeatmapList'
add_heatmap(object, x)
```

Arguments

object   a HeatmapList-class object.

x   a Heatmap-class object or a HeatmapAnnotation-class object or a HeatmapList-class
                object.

Details

There is a shortcut function +.HeatmapList.

Value

A HeatmapList-class object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
ht_list = ht + ht
add_heatmap(ht_list, ht)

ha = HeatmapAnnotation(points = anno_points(1:12, which = "row"),
    which = "row")
add_heatmap(ht_list, ha)
```

annotation_legend_size-HeatmapList-method

Size of the annotation legend viewport

Description

Size of the annotation legend viewport

Usage

```
## S4 method for signature 'HeatmapList'
annotation_legend_size(object, annotation_legend_list = list(), ...)
```

Arguments

object   a HeatmapList-class object.

annotation_legend_list
                      a list of self-defined legend, should be wrapped into grob objects.

...   graphic parameters passed to color_mapping_legend,ColorMapping-method.

Details

Legends for all heatmaps or legends for all annotations will be put in one viewport. This function calculates the size of such viewport. Note graphic parameters for legends will affect the size.

This function is only for internal use.

Value

A unit object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

anno_barplot *Using barplot as annotation*

Description

Using barplot as annotation

Usage

```
anno_barplot(x, which = c("column", "row"),
    gp = gpar(fill = "#CCCCCC"), ...)
```

Arguments

x	a vector of values.
which	is the annotation a column annotation or a row annotation?
gp	graphic parameters.
...	for future use.

Value

A graphic function which can be set in [HeatmapAnnotation](#) constructor method.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
f = anno_barplot(rnorm(10))
grid.newpage(); f(1:10)

f = anno_barplot(rnorm(10), which = "row")
grid.newpage(); f(1:10)
```

---

anno_boxplot *Using boxplot as annotation*

Description

Using boxplot as annotation

Usage

```
anno_boxplot(x, which = c("column", "row"), gp = gpar(fill = "#CCCCCC"),
    pch = 16, size = unit(2, "mm"))
```

Arguments

x	a matrix or a list. If x is a matrix and if which is column, statistics for boxplot is calculated by columns, if which is row, the calculation is by rows.
which	is the annotation a column annotation or a row annotation?
gp	graphic parameters
pch	point type
size	point size

Value

A graphic function which can be set in [HeatmapAnnotation](#) constructor method.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(32), nrow = 4)
f = anno_boxplot(mat)
grid.newpage(); f(1:8)

f = anno_boxplot(mat, which = "row")
grid.newpage(); f(1:4)

lt = lapply(1:4, function(i) rnorm(8))
f = anno_boxplot(lt)
grid.newpage(); f(1:4)
```

---

anno_density   *Using kernel density as annotation*

Description

Using kernel density as annotation

Usage

```
anno_density(x, which = c("column", "row"), gp = gpar(fill = "#CCCCCC"),
    type = c("lines", "violin", "heatmap"), ...)
```

Arguments

x	a matrix or a list. If x is a matrix and if which is column, statistics for density is calculated by columns, if which is row, the calculation is by rows.
which	is the annotation a column annotation or a row annotation?
gp	graphic parameters. Note it is ignored if type equals to heatmap.
type	which type of graphics is used to represent density distribution.
...	pass to density

Value

A graphic function which can be set in HeatmapAnnotation constructor method.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(32), nrow = 4)
f = anno_density(mat)
grid.newpage(); f(1:8)

f = anno_density(mat, which = "row", type = "violin")
grid.newpage(); f(1:4)

lt = lapply(1:4, function(i) rnorm(8))
f = anno_density(lt, type = "heatmap")
grid.newpage(); f(1:4)
```

---

anno_histogram   *Using histogram as annotation*

Description

Using histogram as annotation

Usage

```
anno_histogram(x, which = c("column", "row"), gp = gpar(fill = "#CCCCCC"), ...)
```

Arguments

x	a matrix or a list. If x is a matrix and if which is column, statistics for histogram is calculated by columns, if which is row, the calculation is by rows.
which	is the annotation a column annotation or a row annotation?
gp	graphic parameters
...	pass to hist

Value

A graphic function which can be set in HeatmapAnnotation constructor method.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(32), nrow = 4)
f = anno_histogram(mat)
grid.newpage(); f(1:8)

f = anno_histogram(mat, which = "row")
grid.newpage(); f(1:4)

lt = lapply(1:4, function(i) rnorm(8))
f = anno_histogram(lt)
grid.newpage(); f(1:4)
```

---

anno_points   *Using points as annotation*

Description

Using points as annotation

Usage

```
anno_points(x, which = c("column", "row"), gp = gpar(), pch = 16,
    size = unit(2, "mm"), ...)
```

Arguments

x	a vector of values.
which	is the annotation a column annotation or a row annotation?
gp	graphic parameters.
pch	point type.
size	point size.
...	for future use.

Value

A graphic function which can be set in HeatmapAnnotation constructor method.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
f = anno_points(rnorm(10))
grid.newpage(); f(1:10)
```

ColorMapping   *Constructor methods for ColorMapping class*

Description

Constructor methods for ColorMapping class

Usage

```
ColorMapping(name, colors = NULL, levels = NULL,
    col_fun = NULL, breaks = NULL)
```

Arguments

name	name for this color mapping. It is used for drawing the title of the legend.
colors	discrete colors.
levels	levels that correspond to `colors`. If `colors` is name indexed, `levels` can be ignored.
col_fun	color mapping function that maps continuous values to colors.
breaks	breaks for the continuous color mapping. If col_fun is generated by [colorRamp2](), breaks can be ignored.

Details

`colors` and `levels` are used for discrete color mapping, `col_fun` and `breaks` are used for continuous color mapping.

Value

A [ColorMapping-class]() object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# discrete color mapping for characters
cm = ColorMapping(name = "test",
    colors = c("blue", "white", "red"),
    levels = c("a", "b", "c"))
cm

# discrete color mapping for numeric values
cm = ColorMapping(name = "test",
    colors = c("blue", "white", "red"),
    levels = c(1, 2, 3))
cm

# continuous color mapping
require(circlize)
cm = ColorMapping(name = "test",
    col_fun = colorRamp2(c(0, 0.5, 1), c("blue", "white", "red")))
cm
```

ColorMapping-class   *Class to map values to colors*

Description

Class to map values to colors

Details

The [ColorMapping-class](#) handles color mapping with both discrete values and continuous values. Discrete values are mapped by setting a vector of colors and continuous values are mapped by setting a color mapping function.

Methods

The [ColorMapping-class](#) provides following methods:

- [ColorMapping](#): contructor methods.
- [map_to_colors,ColorMapping-method](#): mapping values to colors.
- [color_mapping_legend,ColorMapping-method](#): draw legend or get the size of the legend.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# for examples, please go to `ColorMapping` method page
NULL
```

color_mapping_legend-ColorMapping-method

Draw legend based on color mapping

Description

Draw legend based on color mapping

Usage

```
## S4 method for signature 'ColorMapping'
color_mapping_legend(object, ..., plot = TRUE, legend_grid_height = unit(3, "mm"),
    legend_grid_width = unit(3, "mm"),
    legend_title_gp = gpar(fontsize = 10, fontface = "bold"),
    legend_label_gp = gpar(fontsize = 10))
```

Arguments

object	a [ColorMapping-class](#) object.
...	pass to [viewport](#).
plot	whether to plot or just return the size of the legend viewport.
legend_grid_height	
	height of each legend grid.
legend_grid_width	
	width of each legend grid.
legend_title_gp	
	graphic parameter for legend title.
legend_label_gp	
	graphic parameter for legend label.

Details

A viewport is created which contains a legend title, legend grids and corresponding labels.

Value

A [unit](#) object which corresponds to the width and height of the legend viewport.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# discrete color mapping for characters
cm = ColorMapping(name = "test",
    colors = c("blue", "white", "red"),
    levels = c("a", "b", "c"))
grid.newpage()
color_mapping_legend(cm)

# discrete color mapping for numeric values
cm = ColorMapping(name = "test",
    colors = c("blue", "white", "red"),
    levels = c(1, 2, 3))
grid.newpage()
color_mapping_legend(cm)

# continuous color mapping
require(circlize)
cm = ColorMapping(name = "test",
    col_fun = colorRamp2(c(0, 0.5, 1), c("blue", "white", "red")))
grid.newpage()
color_mapping_legend(cm, legend_title_gp = gpar(fontsize = 16))
```

---

component_height-dispatch

Method dispatch page for component_height

Description

Method dispatch page for component_height.

Dispatch

component_height can be dispatched on following classes:

- [component_height,HeatmapList-method](), [HeatmapList-class]() class method
- [component_height,Heatmap-method](), [Heatmap-class]() class method

Examples

```
# no example for this internal method
NULL
```

---

component_height-Heatmap-method

Height of each heatmap component

Description

Height of each heatmap component

Usage

```
## S4 method for signature 'Heatmap'
component_height(object, k = 1:9)
```

Arguments

object   a [Heatmap-class](#) object.

k   which component in the heatmap, see [Heatmap-class](#).

Details

This function is only for internal use.

Value

A [unit](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
```

---

component_height-HeatmapList-method

Height of each heatmap list component

Description

Height of each heatmap list component

Usage

```
## S4 method for signature 'HeatmapList'
component_height(object, k = 1:7)
```

Arguments

object   a [HeatmapList-class](#) object.

k   which component in the heatmap list, see [HeatmapList-class](#).

Value

A [unit](#) object

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
```

---

component_width-dispatch

Method dispatch page for component_width

Description

Method dispatch page for component_width.

Dispatch

component_width can be dispatched on following classes:

- [component_width,HeatmapList-method](#), [HeatmapList-class](#) class method
- [component_width,Heatmap-method](#), [Heatmap-class](#) class method

Examples

```
# no example
NULL
```

component_width–Heatmap-method

Width of each heatmap component

Description

Width of each heatmap component

Usage

```
## S4 method for signature 'Heatmap'
component_width(object, k = 1:7)
```

Arguments

object	a Heatmap-class object.
k	which component in the heatmap, see Heatmap-class.

Details

This function is only for internal use.

Value

A unit object.

Detials

This function is only for internal use.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
```

---

component_width-HeatmapList-method
Width of each heatmap list component

Description

Width of each heatmap list component

Usage

```
## S4 method for signature 'HeatmapList'
component_width(object, k = 1:7)
```

Arguments

object   a [HeatmapList-class](#) object.

k   which component in the heatmap list, see [HeatmapList-class](#).

Details

This function is only for internal use.

Value

A [unit](#) object

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
```

---

dist2   *Calculate distance from a matrix*

Description

Calculate distance from a matrix

Usage

```
dist2(mat, pairwise_fun = function(x, y) sqrt(sum((x - y)^2)), ...)
```

Arguments

mat	a matrix. The distance is calculated by rows.
pairwise_fun	a function which calculates distance between two vectors.
...	pass to [dist]().

Details

You can construct any type of distance measurements by defining a pair-wise distance function. The function is implemented by two nested `for` loops, thus the efficiency may not be so good.

Value

A [dist]() object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(40), nr = 4, ncol = 10)
rownames(mat) = letters[1:4]
colnames(mat) = letters[1:10]

d2 = dist2(mat)
d2 = dist2(mat, pairwise_fun = function(x, y) 1 - cor(x, y))
d2 = dist2(mat, pairwise_fun = function(x, y) {
    l = is.na(x) & is.na(y)
    sqrt(sum((x[l] - y[l])^2))
})
```

---

draw-dispatch *Method dispatch page for draw*

Description

Method dispatch page for `draw`.

Dispatch

`draw` can be dispatched on following classes:

- [draw,HeatmapAnnotation-method](), [HeatmapAnnotation-class]() class method
- [draw,SingleAnnotation-method](), [SingleAnnotation-class]() class method
- [draw,HeatmapList-method](), [HeatmapList-class]() class method
- [draw,Heatmap-method](), [Heatmap-class]() class method

Examples

```
# no example
NULL
```

---

draw-Heatmap-method   *Draw a single heatmap*

Description

Draw a single heatmap

Usage

```
## S4 method for signature 'Heatmap'
draw(object, internal = FALSE, test = FALSE, ...)
```

Arguments

object	a Heatmap-class object.
internal	only used inside the calling of draw,HeatmapList-method. Only heatmap without legends will be drawn.
test	only for testing
...	pass to draw,HeatmapList-method.

Details

The function creates a HeatmapList-class object which only contains a single heatmap and call draw,HeatmapList-method to make the final heatmap.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
draw(ht, heatmap_legend_side = "left")
```

draw-HeatmapAnnotation-method
Draw the heatmap annotations

Description

Draw the heatmap annotations

Usage

```
## S4 method for signature 'HeatmapAnnotation'
draw(object, index, ...)
```

Arguments

object	a [HeatmapAnnotation-class](#) object.
index	a vector of order.
...	pass to [viewport](#) which contains all annotations.

Details

A viewport is created. Mostly, this method is used inside [draw,HeatmapList-method](#).

Value

No value is returned.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
df = data.frame(type = c("a", "a", "a", "b", "b", "b"))
ha = HeatmapAnnotation(df = df)
grid.newpage(); draw(ha, 1:6)
grid.newpage(); draw(ha, 6:1)

ha = HeatmapAnnotation(df = df, col = list(type = c("a" =  "red", "b" = "blue")))
grid.newpage(); draw(ha, 1:6)

ha = HeatmapAnnotation(df = df, col = list(type = c("a" =  "red", "b" = "blue")),
    which = "row")
grid.newpage(); draw(ha, 1:6)

ha = HeatmapAnnotation(points = anno_points(1:6))
grid.newpage(); draw(ha, 1:6)
```

```
ha = HeatmapAnnotation(histogram = anno_barplot(1:6))
grid.newpage(); draw(ha, 1:6)

mat = matrix(rnorm(36), 6)
ha = HeatmapAnnotation(boxplot = anno_boxplot(mat))
grid.newpage(); draw(ha, 1:6)
```

draw-HeatmapList-method

Draw a list of heatmaps

Description

Draw a list of heatmaps

Usage

```
## S4 method for signature 'HeatmapList'
draw(object, ..., newpage= TRUE)
```

Arguments

object	a [HeatmapList-class](#) object
...	pass to [make_layout,HeatmapList-method](#)
newpage	whether to create a new page

Details

The function first calls [make_layout,HeatmapList-method](#) to calculate the layout of the heatmap list and the layout of every single heatmap, then makes the plot by re-calling the graphic functions which are already recorded in the layout.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
```

```
ht_list = ht + ht
draw(ht_list)
draw(ht_list, row_title = "row title", column_title = "column title",
heatmap_legend_side = "top")
```

```
draw-SingleAnnotation-method
```
Draw the single annotation

Description

Draw the single annotation

Usage

```
## S4 method for signature 'SingleAnnotation'
draw(object, index)
```

Arguments

object	a [SingleAnnotation-class](#) object.
index	a vector of orders

Details

A viewport is created.

The graphics would be different depending the annotation is a row annotation or a column annotation.

Value

No value is returned.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
anno = SingleAnnotation(name = "test", value = c("a", "a", "a", "b", "b", "b"))
grid.newpage(); draw(anno, 1:5)
grid.newpage(); draw(anno, c(1, 4, 3, 5, 2))

anno = SingleAnnotation(value = c("a", "a", "a", "b", "b", "b"),
    col = c("a" = "red", "b" = "blue"))
grid.newpage(); draw(anno, 1:5)
grid.newpage(); draw(anno, c(1, 4, 3, 5, 2))
```

```
anno = SingleAnnotation(value = c("a", "a", "a", "b", "b", "b"),
    col = c("a" = "red", "b" = "blue"), which = "row")
grid.newpage(); draw(anno, 1:5)

anno = SingleAnnotation(value = 1:10)
grid.newpage(); draw(anno, 1:10)

require(circlize)
anno = SingleAnnotation(value = 1:10, col = colorRamp2(c(1, 10), c("blue", "red")))
grid.newpage(); draw(anno, 1:10)

anno = SingleAnnotation(fun = anno_points(1:10))
grid.newpage(); draw(anno, 1:10)
```

draw_annotation-Heatmap-method
                            Draw column annotations

Description

Draw column annotations

Usage

```
## S4 method for signature 'Heatmap'
draw_annotation(object, which = c("top", "bottom"))
```

Arguments

object   a [Heatmap-class](#) object.

which   are the annotations put on the top or bottom of the heatmap?

Details

A viewport is created which contains column annotations.

Since the column annotations is a [HeatmapAnnotation-class](#) object, the function calls [draw,HeatmapAnnotation-method](#) to draw the annotations.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

draw_annotation_legend-HeatmapList-method
Draw legends for all column annotations

Description

Draw legends for all column annotations

Usage

```
## S4 method for signature 'HeatmapList'
draw_annotation_legend(object, annotation_legend_list = list(), ...)
```

Arguments

object   a [HeatmapList-class](#) object

annotation_legend_list
                   a list of self-defined legend, should be wrapped into [grob](#) objects.

...   graphic parameters passed to [color_mapping_legend,ColorMapping-method](#).

Details

A viewport is created which contains annotation legends.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

draw_dimnames-Heatmap-method

Draw row names or column names

Description

Draw row names or column names

Usage

```
## S4 method for signature 'Heatmap'
draw_dimnames(object,
    which = c("row", "column"), k = 1, ...)
```

Arguments

object	a [Heatmap-class](#) object.
which	are names put on the row or on the column of the heatmap?
k	a matrix may be split by rows, the value identifies which row-slice.
...	pass to [viewport](#), basically for defining the position of the viewport.

Details

A viewport is created which contains row names or column names.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

draw_hclust-Heatmap-method

Draw dendrogram on row or column

Description

Draw dendrogram on row or column

Usage

```
## S4 method for signature 'Heatmap'
draw_hclust(object,
    which = c("row", "column"), k = 1, max_height = NULL, ...)
```

Arguments

object	a [Heatmap-class](#) object.
which	is dendrogram put on the row or on the column of the heatmap?
k	a matrix may be splitted by rows, the value identifies which row-slice.
max_height	maximum height of the dendrograms.
...	pass to [viewport](#), basically for defining the position of the viewport.

Details

If the matrix is split into several row slices, a list of dendrograms will be drawn by the heatmap that each dendrogram corresponds to its row slices.

A viewport is created which contains dendrograms.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

See Also

[grid.dendrogram](#)

Examples

```
# no example for this internal method
NULL
```

draw_heatmap_body-Heatmap-method

Draw the heatmap body

Description

Draw the heatmap body

Usage

```
## S4 method for signature 'Heatmap'
draw_heatmap_body(object, k = 1, ...)
```

Arguments

object	a Heatmap-class object.
k	a matrix may be split by rows, the value identifies which row-slice.
...	pass to viewport, basically for defining the position of the viewport.

Details

The matrix can be split into several parts by rows if km or split is specified when initializing the Heatmap object. If the matrix is split, there will be gaps between rows to identify different row-slice.

A viewport is created which contains subset rows of the heatmap.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

draw_heatmap_legend-HeatmapList-method

Draw legends for all heatmaps

Description

Draw legends for all heatmaps

Usage

```
## S4 method for signature 'HeatmapList'
draw_heatmap_legend(object, ...)
```

Arguments

object	a [HeatmapList-class](#) object
...	graphic parameters passed to [color_mapping_legend,ColorMapping-method](#).

Details

A viewport is created which contains heatmap legends.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

draw_heatmap_list-HeatmapList-method
Draw the list of heatmaps

Description

Draw the list of heatmaps

Usage

```
## S4 method for signature 'HeatmapList'
draw_heatmap_list(object)
```

Arguments

object   a [HeatmapList-class](#) object

Details

A viewport is created which contains heatmaps.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

draw_title-dispatch   *Method dispatch page for draw_title*

Description

Method dispatch page for draw_title.

Dispatch

draw_title can be dispatched on following classes:

- draw_title,HeatmapList-method, HeatmapList-class class method
- draw_title,Heatmap-method, Heatmap-class class method

Examples

```
# no example
NULL
```

---

draw_title-Heatmap-method

Draw heatmap title

Description

Draw heatmap title

Usage

```
## S4 method for signature 'Heatmap'
draw_title(object,
    which = c("row", "column"), k = 1, ...)
```

Arguments

object	a Heatmap-class object.
which	is title put on the row or on the column of the heatmap?
k	a matrix may be split by rows, the value identifies which row-slice.
...	pass to viewport, basically for defining the position of the viewport.

Details

A viewport is created which contains heatmap title.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

```
draw_title-HeatmapList-method
```
Draw heatmap list title

Description

Draw heatmap list title

Usage

```
## S4 method for signature 'HeatmapList'
draw_title(object,
    which = c("column", "row"))
```

Arguments

object	a [HeatmapList-class](#) object
which	dendrogram on the row or on the column of the heatmap

Details

A viewport is created which contains heatmap list title.

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

get_color_mapping_list-HeatmapAnnotation-method
Get a list of color mapping objects

Description

Get a list of color mapping objects

Usage

```
## S4 method for signature 'HeatmapAnnotation'
get_color_mapping_list(object)
```

Arguments

object   a [HeatmapAnnotation-class](#) object.

Details

Color mapping for visible simple annotations are only returned.

This function is only for internal use.

Value

A list of [ColorMapping-class](#) objects or an empty list.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

grid.dendrogram   *Draw dendrogram under grid system*

Description

Draw dendrogram under grid system

Usage

```
grid.dendrogram(dend, facing = c("bottom", "top", "left", "right"),
    max_height = NULL, order = c("normal", "reverse"), ...)
```

Arguments

`dend`	a [dendrogram](#) object.
`facing`	facing of the dendrogram.
`max_height`	maximum height of the dendrogram. It is useful if you want to plot more than one dendrograms.
`order`	should leaves of dendrogram be put in the normal order or reverse order?
`...`	pass to [viewport](#) that contains the dendrogram.

Details

The dendrogram tree can be renderred (e.g. by dendextend package).

A viewport is created which contains the dendrogram.

Value

No value is returned.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
hc = hclust(dist(USArrests[1:5, ]))
dend = as.dendrogram(hc)

grid.newpage()
layout = grid.layout(nrow = 2, ncol = 2)
pushViewport(viewport(layout = layout))
grid.dendrogram(dend, layout.pos.row = 1, layout.pos.col = 1)
grid.dendrogram(dend, facing = "top", layout.pos.row = 1, layout.pos.col = 2)
grid.dendrogram(dend, facing = "top", order = "reverse", layout.pos.row = 2,
    layout.pos.col = 1)
grid.dendrogram(dend, facing = "left", layout.pos.row = 2, layout.pos.col = 2)
upViewport()
```

Heatmap *Constructor method for Heatmap class*

Description

Constructor method for Heatmap class

Usage

```
Heatmap(matrix, col, name, rect_gp = gpar(col = NA),
    cell_fun = function(j, i, x, y, width, height, fill) NULL,
    row_title = character(0), row_title_side = c("left", "right"),
    row_title_gp = gpar(fontsize = 14), column_title = character(0),
    column_title_side = c("top", "bottom"), column_title_gp = gpar(fontsize = 14),
    cluster_rows = TRUE, clustering_distance_rows = "euclidean",
    clustering_method_rows = "complete", row_hclust_side = c("left", "right"),
    row_hclust_width = unit(10, "mm"), show_row_hclust = TRUE,
    row_hclust_gp = gpar(), cluster_columns = TRUE,
    clustering_distance_columns = "euclidean", clustering_method_columns = "complete",
    column_hclust_side = c("top", "bottom"), column_hclust_height = unit(10, "mm"),
    show_column_hclust = TRUE, column_hclust_gp = gpar(),
    row_names_side = c("right", "left"), show_row_names = TRUE,
    row_names_max_width = unit(4, "cm"), row_names_gp = gpar(fontsize = 12),
    column_names_side = c("bottom", "top"),
    show_column_names = TRUE, column_names_max_height = unit(4, "cm"),
    column_names_gp = gpar(fontsize = 12),
    top_annotation = new("HeatmapAnnotation"),
    top_annotation_height = unit(5*length(top_annotation@anno_list), "mm"),
    bottom_annotation = new("HeatmapAnnotation"),
    bottom_annotation_height = unit(5*length(bottom_annotation@anno_list), "mm"),
    km = 1, split = NULL, gap = unit(1, "mm"),
    combined_name_fun = function(x) paste(x, collapse = "/"),
    width = NULL, show_heatmap_legend = TRUE)
```

Arguments

matrix   a matrix. Either numeric or character. If it is a simple vector, it will be converted to a one-column matrix.

col   a vector of colors if the color mapping is discrete or a color mapping function if the matrix is continuous numbers. Pass to ColorMapping.

name   name of the heatmap. The name is used as the title of the heatmap legend.

rect_gp   graphic parameters for drawing rectangles (for heatmap body).

cell_fun   self-defined function to add graphics on each cell. Six parameters will be passed into this function: i, j, x, y, width, height which are row index, column index in matrix, coordinate of the middle points in the heatmap body viewport, and the width and height of the cell.

row_title   title on row.

row_title_side will the title be put on the left or right of the heatmap?

row_title_gp   graphic parameters for drawing text.

column_title   title on column.

column_title_side

                will the title be put on the top or bottom of the heatmap?

column_title_gp

                graphic parameters for drawing text.

cluster_rows   If the value is a logical, it means whether make cluster on rows. The value
                   can also be a [hclust](#) or a [dendrogram](#) that already contains clustering infor-
                   mation. This means you can use any type of clustering methods and render the
                   [dendrogram](#) object with self-defined graphic settings.

clustering_distance_rows
                   it can be a pre-defined character which is in ("euclidean", "maximum", "man-
                   hattan", "canberra", "binary", "minkowski", "pearson", "spearman", "kendall").
                   It can also be a function. If the function has one argument, the input argument
                   should be a matrix and the returned value should be a [dist](#) object. If the func-
                   tion has two arguments, the input arguments are two vectors and the function
                   calculates distance between these two vectors.

clustering_method_rows
                   method to make cluster, pass to [hclust](#).

row_hclust_side
                   should the row cluster be put on the left or right of the heatmap?

row_hclust_width
                   width of the row cluster, should be a [unit](#) object.

show_row_hclust
                   whether show row clusters.

row_hclust_gp   graphics parameters for drawing lines. If users already provide a [dendrogram](#)
                   object with edges rendered, this argument will be ignored.

cluster_columns
                   whether make cluster on columns. Same settings as cluster_rows.

clustering_distance_columns
                   same setting as clustering_distance_rows.

clustering_method_columns
                   method to make cluster, pass to [hclust](#).

column_hclust_side
                   should the column cluster be put on the top or bottom of the heatmap?

column_hclust_height
                   height of the column cluster, should be a [unit](#) object.

show_column_hclust
                   whether show column clusters.

column_hclust_gp
                   graphic parameters for drawling lines. Same settings as row_hclust_gp.

row_names_side   should the row names be put on the left or right of the heatmap?

show_row_names   whether show row names.

row_names_max_width
                   maximum width of row names viewport. Because some times row names can
                   be very long, it is not reasonable to show them all.

row_names_gp   graphic parameters for drawing text.

column_names_side
                   should the column names be put on the top or bottom of the heatmap?

column_names_max_height
                   maximum height of column names viewport.

show_column_names

> whether show column names.

column_names_gp

> graphic parameters for drawing text.

top_annotation a HeatmapAnnotation object which contains a list of annotations.

top_annotation_height

> total height of the column annotations on the top.

bottom_annotation

> a HeatmapAnnotation object.

bottom_annotation_height

> total height of the column annotations on the bottom.

km

> do k-means clustering on rows. If the value is larger than 1, the heatmap will be split by rows according to the k-means clustering. For each row-clusters, hierarchical clustering is still applied with parameters above.

split

> a vector or a data frame by which the rows are split.

gap

> gap between row-slices if the heatmap is split by rows, should be unit object.

combined_name_fun

> if the heatmap is split by rows, how to make a combined row title for each slice? The input parameter for this function is a vector which contains level names under each column in split.

width

> the width of the single heatmap, should be a fixed unit object. It is used for the layout when the heatmap is appended to a list of heatmaps.

show_heatmap_legend

> whether show heatmap legend?

Details

The initialization function only applies parameter checking and fill values to each slot with proper ones. Then it will be ready for clustering and layout.

Following methods can be applied on the Heatmap-class object:

- show,Heatmap-method: draw a single heatmap with default parameters
- draw,Heatmap-method: draw a single heatmap.
- add_heatmap,Heatmap-method append heatmaps and row annotations to a list of heatmaps.

The constructor function pretends to be a high-level graphic function because the show method of the Heatmap-class object actually plots the graphics.

Value

A Heatmap-class object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

require(circlize)

Heatmap(mat)
Heatmap(mat, col = colorRamp2(c(-3, 0, 3), c("green", "white", "red")))
Heatmap(mat, name = "test")
Heatmap(mat, column_title = "blablabla")
Heatmap(mat, row_title = "blablabla")
Heatmap(mat, column_title = "blablabla", column_title_side = "bottom")
Heatmap(mat, column_title = "blablabla", column_title_gp = gpar(fontsize = 20,
    fontface = "bold"))
Heatmap(mat, cluster_rows = FALSE)
Heatmap(mat, clustering_distance_rows = "pearson")
Heatmap(mat, clustering_distance_rows = function(x) dist(x))
Heatmap(mat, clustering_distance_rows = function(x, y) 1 - cor(x, y))
Heatmap(mat, clustering_method_rows = "single")
Heatmap(mat, row_hclust_side = "right")
Heatmap(mat, row_hclust_width = unit(1, "cm"))
Heatmap(mat, row_names_side = "left", row_hclust_side = "right",
    column_names_side = "top", column_hclust_side = "bottom")
Heatmap(mat, show_row_names = FALSE)

mat2 = mat
rownames(mat2) = NULL
colnames(mat2) = NULL
Heatmap(mat2)

Heatmap(mat, row_names_gp = gpar(fontsize = 20))
Heatmap(mat, km = 2)
Heatmap(mat, split = rep(c("A", "B"), 6))
Heatmap(mat, split = data.frame(rep(c("A", "B"), 6), rep(c("C", "D"), each = 6)))
Heatmap(mat, split = data.frame(rep(c("A", "B"), 6), rep(c("C", "D"), each = 6)),
    combined_name_fun = function(x) paste(x, collapse = "\n"))

annotation = HeatmapAnnotation(df = data.frame(type = c(rep("A", 6), rep("B", 6))))
Heatmap(mat, top_annotation = annotation)

annotation = HeatmapAnnotation(df = data.frame(type1 = rep(c("A", "B"), 6),
    type2 = rep(c("C", "D"), each = 6)))
Heatmap(mat, bottom_annotation = annotation)

annotation = data.frame(value = rnorm(10))
annotation = HeatmapAnnotation(df = annotation)
Heatmap(mat, top_annotation = annotation)

annotation = data.frame(value = rnorm(10))
value = 1:10
```

```
ha = HeatmapAnnotation(df = annotation, points = anno_points(value),
    annotation_height = c(1, 2))
Heatmap(mat, top_annotation = ha, top_annotation_height = unit(2, "cm"),
    bottom_annotation = ha)

# character matrix
mat3 = matrix(sample(letters[1:6], 100, replace = TRUE), 10, 10)
rownames(mat3) = {x = letters[1:10]; x[1] = "aaaaaaaaaaaaaaaaaaaaaaaa";x}
Heatmap(mat3, rect_gp = gpar(col = "white"))

mat = matrix(1:9, 3, 3)
rownames(mat) = letters[1:3]
colnames(mat) = letters[1:3]

Heatmap(mat, rect_gp = gpar(col = "white"),
    cell_fun = function(i, j, x, y, width, height, fill) {
        grid.text(mat[i, j], x = x, y = y)
    },
    cluster_rows = FALSE, cluster_columns = FALSE, row_names_side = "left",
    column_names_side = "top")
```

Heatmap-class   *Class for a single heatmap*

Description

Class for a single heatmap

Details

The components for a single heamtap are placed into a 9 x 7 layout:

```
        +------+ (1)
        +------+ (2)
        +------+ (3)
        +------+ (4)
  +-+-+-+------+-+-+-+
  |1|2|3| 4(5) |5|6|7|
  +-+-+-+------+-+-+-+
        +------+ (6)
        +------+ (7)
        +------+ (8)
        +------+ (9)
```

From top to bottom in column 4, the regions are:

- title which is put on the top of the heatmap, graphics are drawn by draw_title,Heatmap-method.

- column cluster on the top, graphics are drawn by `draw_hclust,Heatmap-method`.
- column annotation on the top, graphics are drawn by `draw_annotation,Heatmap-method`.
- column names on the top, graphics are drawn by `draw_dimnames,Heatmap-method`.
- heatmap body, graphics are drawn by `draw_heatmap_body,Heatmap-method`.
- column names on the bottom, graphics are drawn by `draw_dimnames,Heatmap-method`.
- column annotation on the bottom, graphics are drawn by `draw_annotation,Heatmap-method`.
- column cluster on the bottom, graphics are drawn by `draw_hclust,Heatmap-method`.
- title on the bottom, graphics are drawn by `draw_title,Heatmap-method`.

From left to right in row 5, the regions are:

- title which is put in the left of the heatmap, graphics are drawn by `draw_title,Heatmap-method`.
- row cluster on the left, graphics are drawn by `draw_hclust,Heatmap-method`.
- row names on the left, graphics are drawn by `draw_dimnames,Heatmap-method`.
- heatmap body
- row names on the right, graphics are drawn by `draw_dimnames,Heatmap-method`.
- row cluster on the right, graphics are drawn by `draw_hclust,Heatmap-method`.
- title on the right, graphics are drawn by `draw_title,Heatmap-method`.

The `Heatmap-class` is not responsible for heatmap legend and annotation legends. The `draw,Heatmap-method` method will construct a `HeatmapList-class` object which only contains one single heatmap and call `draw,HeatmapList-method` to make a complete heatmap.

Methods

The `Heatmap-class` provides following methods:

- `Heatmap`: constructor method.
- `draw,Heatmap-method`: draw a single heatmap.
- `add_heatmap,Heatmap-method` append heatmaps and row annotations to a list of heatmaps.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# for examples, please go to `Heatmap` method page
NULL
```

HeatmapAnnotation   *Constructor method for HeatmapAnnotation class*

Description

Constructor method for HeatmapAnnotation class

Usage

```
HeatmapAnnotation(df, name, col, show_legend, ...,
    which = c("column", "row"), annotation_height = 1, annotation_width = 1,
    height = unit(1, "cm"), width = unit(1, "cm"), gp = gpar(col = NA))
```

Arguments

df	a data frame. Each column will be treated as a simple annotation. The data frame must have column names.
name	name of the heatmap annotation
col	a list of colors which contains color mapping to columns in df. See [SingleAnnotation](#) for how to set colors.
show_legend	whether show legend for each column in df.
...	functions which define complex annotations. Values should be named arguments.
which	are the annotations row annotations or column annotations?
annotation_height	
	height of each annotation if annotations are column annotations.
annotation_width	
	width of each annotation if annotations are row annotations.
height	not using currently.
width	width of the whole heatmap annotations, only used for column annotation when appending to the list of heatmaps.
gp	graphic parameters for simple annotations.

Details

The simple annotations are defined by df and col arguments. Complex annotations are defined by the function list.

Value

A [HeatmapAnnotation-class](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
df = data.frame(type = c("a", "a", "a", "b", "b", "b"))
ha = HeatmapAnnotation(df = df)

ha = HeatmapAnnotation(df = df, col = list(type = c("a" =  "red", "b" = "blue")))

ha = HeatmapAnnotation(df = df, col = list(type = c("a" =  "red", "b" = "blue")),
    which = "row")

ha = HeatmapAnnotation(points = anno_points(1:6))

ha = HeatmapAnnotation(histogram = anno_points(1:6))

mat = matrix(rnorm(36), 6)
ha = HeatmapAnnotation(boxplot = anno_boxplot(mat))
```

HeatmapAnnotation-class

Class for heatmap annotations

Description

Class for heatmap annotations

Details

A complex heatmap contains a list of annotations which represent as different graphics placed on
rows and columns. The HeatmapAnnotation-class is a category of single annotations which are
by a list of SingleAnnotation-class objects with same number of rows or columns.

Methods

The HeatmapAnnotation-class provides following methods:

- HeatmapAnnotation: constructor method
- draw,HeatmapAnnotation-method: draw the annotations

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# for examples, please go to `HeatmapAnnotation` method page
NULL
```

HeatmapList   *Constructor method for HeatmapList class*

Description

Constructor method for HeatmapList class

Usage

```
HeatmapList(...)
```

Arguments

   ...   arguments

Details

There is no public constructor method for the HeatmapList-class.

Value

No value is returned.

Detailes

There is no public constructor method for the HeatmapList-class.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example
NULL
```

HeatmapList-class   *Class for a list of heatmaps*

Description

Class for a list of heatmaps

Details

A heatmap list is defined as a list of heatmaps and row annotations.

The components for the heamtap list are placed into a 7 x 7 layout:

```
        +------+(1)
        +------+(2)
        +------+(3)
   +-+-+-+------+-+-+-+
   |1|2|3| 4(4) |5|6|7|
   +-+-+-+------+-+-+-+
        +------+(5)
        +------+(6)
        +------+(7)
```

From top to bottom in column 4, the regions are:

- annotation legend on the top, graphics are drawn by `draw_annotation_legend,HeatmapList-method`.
- heatmap legend on the top, graphics are drawn by `draw_heatmap_legend,HeatmapList-method`.
- title for the heatmap list which is put on the top, graphics are drawn by `draw_title,HeatmapList-method`.
- the list of heatmaps and row annotations
- title for the heatmap list which is put on the bottom, graphics are drawn by `draw_title,HeatmapList-method`.
- heatmap legend on the bottom, graphics are drawn by `draw_heatmap_legend,HeatmapList-method`.
- annotation legend on the bottom, graphics are drawn by `draw_annotation_legend,HeatmapList-method`.

From left to right in row 4, the regions are:

- annotation legend on the left, graphics are drawn by `draw_annotation_legend,HeatmapList-method`.
- heatmap legend on the left, graphics are drawn by `draw_heatmap_legend,HeatmapList-method`.
- title for the heatmap list which is put on the left, graphics are drawn by `draw_title,HeatmapList-method`.
- the list of heatmaps and row annotations
- title for the heatmap list which is put on the right, graphics are drawn by `draw_title,HeatmapList-method`.
- heatmap legend on the right, graphics are drawn by `draw_heatmap_legend,HeatmapList-method`.
- annotation legend on the right, graphics are drawn by `draw_annotation_legend,HeatmapList-method`.

For the list of heatmaps which are placed at (5, 5) in the layout, the heatmaps and row annotations are placed one after the other.

Methods

The [HeatmapList-class](#) provides following methods:

- [draw,HeatmapList-method](#): draw the list of heatmaps and row annotations.
- [add_heatmap,HeatmapList-method](#) add heatmaps to the list of heatmaps.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
ht + ht
ht + ht + ht

ht_list = ht + ht
ht + ht_list

ha = HeatmapAnnotation(points = anno_points(1:12, which = "row"),
    which = "row")
ht + ha
ht_list + ha
```

---

heatmap_legend_size-HeatmapList-method
Size of the heatmap legend viewport

Description

Size of the heatmap legend viewport

Usage

```
## S4 method for signature 'HeatmapList'
heatmap_legend_size(object, ...)
```

Arguments

object   a [HeatmapList-class](#) object

...   graphic parameters passed to [color_mapping_legend,ColorMapping-method](#).

Details

This function is only for internal use.

Value

A [unit](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

make_column_cluster-Heatmap-method

Make cluster on columns

Description

Make cluster on columns

Usage

```
## S4 method for signature 'Heatmap'
make_column_cluster(object, order = NULL)
```

Arguments

object   a [Heatmap-class](#) object.

order   a pre-defined order.

Details

The function will fill or adjust column_hclust and column_order slots.

This function is only for internal use.

Value

A [Heatmap-class](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

make_layout-dispatch   *Method dispatch page for make_layout*

Description

Method dispatch page for make_layout.

Dispatch

make_layout can be dispatched on following classes:

- make_layout,HeatmapList-method, HeatmapList-class class method
- make_layout,Heatmap-method, Heatmap-class class method

Examples

```
# no example
NULL
```

---

make_layout-Heatmap-method

Make the layout of a single heatmap

Description

Make the layout of a single heatmap

Usage

```
## S4 method for signature 'Heatmap'
make_layout(object)
```

Arguments

object   a Heatmap-class object.

Details

The layout of the single heatmap will be established by setting the size of each heatmap components. Also functions that make graphics for heatmap components will be recorded.

Whether apply row clustering or column clustering affects the layout, so clustering should be applied first before making the layout.

This function is only for internal use.

Value

A [Heatmap-class](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

make_layout-HeatmapList-method

Make layout for the complete plot

Description

Make layout for the complete plot

Usage

```
## S4 method for signature 'HeatmapList'
make_layout(object, row_title = character(0),
    row_title_side = c("left", "right"), row_title_gp = gpar(fontsize = 14),
    column_title = character(0), column_title_side = c("top", "bottom"),
    column_title_gp = gpar(fontsize = 14),
    heatmap_legend_side = c("right", "left", "bottom", "top"),
    show_heatmap_legend = TRUE,
    annotation_legend_side = c("right", "left", "bottom", "top"),
    show_annotation_legend = TRUE, annotation_legend_list = list(),
    gap = unit(3, "mm"), auto_adjust = TRUE,
    main_heatmap = which(sapply(object@ht_list, inherits, "Heatmap"))[1],
    row_hclust_side = c("original", "left", "right"),
    row_sub_title_side = c("original", "left", "right"), ...)
```

Arguments

object	a [HeatmapList-class](#) object.
row_title	title on the row.
row_title_side	will the title be put on the left or right of the heatmap.
row_title_gp	graphic parameters for drawing text.
column_title	title on the column.
column_title_side	
	will the title be put on the top or bottom of the heatmap.

column_title_gp

> graphic parameters for drawing text.

heatmap_legend_side

> side of the heatmap legend.

show_heatmap_legend

> whether show heatmap legend.

annotation_legend_side

> side of annotation legend.

show_annotation_legend

> whether show annotation legend.

annotation_legend_list

> a list of self-defined legend, should be wrapped into grob objects.

gap   gap between heatmaps, should be a unit object.

auto_adjust   auto adjust if the number of heatmap is larger than one.

main_heatmap   name or index for the main heatmap

row_hclust_side

> if auto adjust, where to put the row dendrograms for the main heatmap

row_sub_title_side

> if auto adjust, where to put sub row titles for the main heatmap

...   graphic parameters pass to color_mapping_legend,ColorMapping-method.

Details

It sets the size of each component of the heatmap list and adjusts graphic parameters for each heatmap if necessary.

The layout for the heatmap list and layout for each heatmap are calculated when drawing the heatmap list.

This function is only for internal use.

Value

A HeatmapList-class object in which settings for each heatmap are adjusted.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

make_row_cluster-Heatmap-method
Make cluster on rows

Description

Make cluster on rows

Usage

```
## S4 method for signature 'Heatmap'
make_row_cluster(object, order = NULL, km = object@matrix_param$km,
    split = object@matrix_param$split)
```

Arguments

object	a [Heatmap-class](#) object.
order	a pre-defined order.
km	if apply k-means clustering on rows, number of clusters.
split	a vector or a data frame by which the rows are be split.

Details

The function will fill or adjust row_hclust_list, row_order_list, row_title and matrix_param slots.

If order is defined, no clustering will be applied.

This function is only for internal use.

Value

A [Heatmap-class](#) object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

---

map_to_colors-ColorMapping-method

Map values to colors

Description

Map values to colors

Usage

```
## S4 method for signature 'ColorMapping'
map_to_colors(object, x)
```

Arguments

object	a [ColorMapping-class](#) object.
x	input values.

Details

It maps a vector of values to a vector of colors.

Value

A vector of colors.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# discrete color mapping for characters
cm = ColorMapping(name = "test",
    colors = c("blue", "white", "red"),
    levels = c("a", "b", "c"))
map_to_colors(cm, "a")
map_to_colors(cm, c("a", "a", "b"))

# discrete color mapping for numeric values
cm = ColorMapping(name = "test",
    colors = c("blue", "white", "red"),
    levels = c(1, 2, 3))
map_to_colors(cm, 1)
map_to_colors(cm, "1")
map_to_colors(cm, c(1, 1, 2, 2))

# continuous color mapping
require(circlize)
```

```
cm = ColorMapping(name = "test",
    col_fun = colorRamp2(c(0, 0.5, 1), c("blue", "white", "red")))
map_to_colors(cm, 0.2)
map_to_colors(cm, seq(0.2, 0.8, by = 0.1))
```

---

prepare-Heatmap-method

Prepare the heatmap

Description

Prepare the heatmap

Usage

```
## S4 method for signature 'Heatmap'
prepare(object, row_order = NULL, split = object@matrix_param$split)
```

Arguments

object   a Heatmap-class object.

row_order   orders of rows, pass to make_row_cluster,Heatmap-method. Because if more
              than one heatmaps are drawn by columns, the order of some heatmap will be
              adjusted by one certain heatmap, this argument is used to pass a pre-defined row
              order.

split   how to split rows in the matrix, passing to make_row_cluster,Heatmap-method.

Details

The preparation of the heatmap includes following steps:

- making clustering on rows if specified (by calling make_row_cluster,Heatmap-method)
- making clustering on columns if specified (by calling make_column_cluster,Heatmap-method)
- making the layout of the heatmap (by calling make_layout,Heatmap-method)

This function is only for internal use.

Value

A Heatmap-class object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

set_component_height-Heatmap-method
Set height of each heatmap component

Description

Set height of each heatmap component

Usage

```
## S4 method for signature 'Heatmap'
set_component_height(object, k, v)
```

Arguments

object   a [Heatmap-class](#) object.

k   which components, see [Heatmap-class](#).

v   height of the component, a [unit](#) object.

Details

This function is only for internal use.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# no example for this internal method
NULL
```

show-ColorMapping-method
                        Print ColorMapping object

Description

Print ColorMapping object

Usage

```
## S4 method for signature 'ColorMapping'
show(object)
```

Arguments

object   a [ColorMapping-class](#) object.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

show-dispatch   *Method dispatch page for show*

Description

Method dispatch page for show.

Dispatch

show can be dispatched on following classes:

- [show,ColorMapping-method](#), [ColorMapping-class](#) class method
- [show,HeatmapAnnotation-method](#), [HeatmapAnnotation-class](#) class method
- [show,SingleAnnotation-method](#), [SingleAnnotation-class](#) class method
- [show,HeatmapList-method](#), [HeatmapList-class](#) class method
- [show,Heatmap-method](#), [Heatmap-class](#) class method

Examples

```
# no example
NULL
```

show-Heatmap-method   *Draw the single heatmap with default parameters*

Description

Draw the single heatmap with default parameters

Usage

```
## S4 method for signature 'Heatmap'
show(object)
```

Arguments

object   a Heatmap-class object.

Details

Actually it calls draw,Heatmap-method, but only with default parameters. If users want to customize the heatmap, they can pass parameters directly to draw,Heatmap-method.

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
mat = matrix(rnorm(80, 2), 8, 10)
mat = rbind(mat, matrix(rnorm(40, -2), 4, 10))
rownames(mat) = letters[1:12]
colnames(mat) = letters[1:10]

ht = Heatmap(mat)
ht
draw(ht, heatmap_legend_side = "left")
```

show-HeatmapAnnotation-method

Print the Heatmap Annotation object

Description

Print the Heatmap Annotation object

Usage

```
## S4 method for signature 'HeatmapAnnotation'
show(object)
```

Arguments

object   a [HeatmapAnnotation-class](#) object.

Value

No value is returned.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

show-HeatmapList-method

Draw a list of heatmaps with default parameters

Description

Draw a list of heatmaps with default parameters

Usage

```
## S4 method for signature 'HeatmapList'
show(object)
```

Arguments

object   a [HeatmapList-class](#) object.

Details

Actually it calls [draw,HeatmapList-method](#), but only with default parameters. If users want to customize the heatmap, they can pass parameters directly to [draw,HeatmapList-method](#).

Value

This function returns no value.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

---

show-SingleAnnotation-method
Print the SingleAnnotation object

Description

Print the SingleAnnotation object

Usage

```
## S4 method for signature 'SingleAnnotation'
show(object)
```

Arguments

object   a SingleAnnotation-class object.

Value

No value is returned.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

SingleAnnotation   *Constructor method for SingleAnnotation class*

Description

Constructor method for SingleAnnotation class

Usage

```
SingleAnnotation(name, value, col, fun, which = c("column", "row"),
    show_legend = TRUE, gp = gpar(col = NA))
```

Arguments

`name`	name for this annotation.
`value`	A vector of annotation.
`col`	colors corresponding to `value`. If the mapping is discrete mapping, the value of `col` should be a vector; If the mapping is continuous mapping, the value of `col` should be a color mapping function.
`fun`	a self-defined function to add annotation graphics. The argument of this function should only be a vector of index that corresponds to rows or columns.
`which`	is the annotation a row annotation or a column annotation?
`show_legend`	if it is a simple annotation, whether show legend when making the complete heatmap.
`gp`	graphic parameters for simple annotations.

Details

The most simple annotation is one row or one column grids in which different colors represent different classes of the data. Here the function use `ColorMapping-class` to process such simple annotation. `value` and `col` arguments controls values and colors of the simple annotation and a `ColorMapping-class` object will be constructed based on `value` and `col`.

`fun` is used to construct a more complex annotation. Users can add any type of annotation graphics by implementing a function. The only input argument of `fun` is a index of rows or columns which is already adjusted by the clustering. In the package, there are already several annotation graphic function generators: `anno_points`, `anno_histogram` and `anno_boxplot`.

One thing that users should be careful is the difference of coordinates when the annotation is a row annotation or a column annotation.

Value

A `SingleAnnotation-class` object.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

Examples

```
# discrete character
SingleAnnotation(name = "test", value = c("a", "a", "a", "b", "b", "b"))
SingleAnnotation(name = "test", value = c("a", "a", "a", "b", "b", "b"),
    which = "row")

# with defined colors
SingleAnnotation(value = c("a", "a", "a", "b", "b", "b"),
    col = c("a" = "red", "b" = "blue"))

# continuous numbers
require(circlize)
```

```
SingleAnnotation(value = 1:10)
SingleAnnotation(value = 1:10, col = colorRamp2(c(1, 10), c("blue", "red")))

# self-defined graphic function
SingleAnnotation(fun = anno_points(1:10))
```

SingleAnnotation-class

Class for a single annotation

Description

Class for a single annotation

Details

A complex heatmap always has more than one annotations on rows and columns. Here the `SingleAnnotation-class` defines the basic unit of annotations. The most simple annotation is one row or one column grids in which different colors represent different classes of the data. The annotation can also be more complex graphics, such as a boxplot that shows data distribution in corresponding row or column.

The `SingleAnnotation-class` is used for storing data for a single annotation and provides methods for drawing annotation graphics.

Methods

The `SingleAnnotation-class` provides following methods:

- `SingleAnnotation`: constructor method
- `draw,SingleAnnotation-method`: draw the single annotation.

Author(s)

Zuguang Gu <z.gu@dkfz.de>

See Also

The `SingleAnnotation-class` is always used internally. The public `HeatmapAnnotation-class` contains a list of `SingleAnnotation-class` objects and is used to add annotation graphics on heatmaps.

Examples

```
# for examples, please go to `SingleAnnotation` method page
NULL
```

Index