

# The l3coffins-scaling package

## Scaling coffins\*

The L<sup>A</sup>T<sub>E</sub>X3 Project<sup>†</sup>

Released 2011/08/28

The material in this module provides low-level support for scaling and resizing coffins. The core coffin implementation is found in `l3coffins` in the `l3kernel` bundle. In contrast to that stable code, this code is highly experimental.

<code>\coffin_resize:Nnn</code>
<code>\coffin_resize:cnn</code>

`\coffin_resize:Nnn <coffin> {<width>} {<total-height>}`

Resized the `<coffin>` to `<width>` and `<total-height>`, both of which should be given as dimension expressions. These may include the terms `\TotalHeight`, `\Height`, `\Depth` and `\Width`, which will evaluate to the appropriate dimensions of the `<coffin>`.

<code>\coffin_scale:Nnn</code>
<code>\coffin_scale:cnn</code>

`\coffin_scale:Nnn <coffin> {<x-scale>} {<y-scale>}`

Scales the `<coffin>` by a factors `<x-scale>` and `<y-scale>` in the horizontal and vertical directions, respectively. The two scale factors should be given as real numbers.

## 1 l3coffins-scaling Implementation

```
1 <*package>
2 \ProvidesExplPackage
3   {\ExplFileName}{\ExplFileDate}{\ExplFileVersion}{\ExplFileDescription}
```

### 1.1 Resizing coffins

`\l_coffin_scale_x_fp` Storage for the scaling factors in  $x$  and  $y$ , respectively.  
`\l_coffin_scale_y_fp`

```
4 \fp_new:N \l_coffin_scale_x_fp
5 \fp_new:N \l_coffin_scale_y_fp
```

---

\*This file describes v2678, last revised 2011/08/28.

<sup>†</sup>E-mail: [latex-team@latex-project.org](mailto:latex-team@latex-project.org)

`\l_coffin_scaled_total_height_dim`  
`\l_coffin_scaled_width_dim`

When scaling, the values given have to be turned into absolute values.

```
6 \dim_new:N \l_coffin_scaled_total_height_dim
7 \dim_new:N \l_coffin_scaled_width_dim
```

`\coffin_resize:Nnn`  
`\coffin_resize:cnm`

Resizing a coffin begins by setting up the user-friendly names for the dimensions of the coffin box. The new sizes are then turned into scale factor. This is the same operation as takes place for the underlying box, but that operation is grouped and so the same calculation is done here.

```
8 \cs_new_protected_nopar:Npn \coffin_resize:Nnn #1#2#3
9 {
10   \coffin_set_user_dimensions:N #1
11   \fp_set_from_dim:Nn \l_coffin_scale_x_fp {#2}
12   \fp_set_from_dim:Nn \l_coffin_tmp_fp { \Width }
13   \fp_div:Nn \l_coffin_scale_x_fp { \l_coffin_tmp_fp }
14   \fp_set_from_dim:Nn \l_coffin_scale_y_fp {#3}
15   \fp_set_from_dim:Nn \l_coffin_tmp_fp { \TotalHeight }
16   \fp_div:Nn \l_coffin_scale_y_fp { \l_coffin_tmp_fp }
17   \hbox_set:Nn #1 { \resizebox * {#2} {#3} { \box_use:N #1 } }
18   \coffin_resize_common:Nnn #1 {#2} {#3}
19 }
20 \cs_generate_variant:Nn \coffin_resize:Nnn { c }
```

(End definition for `\coffin_resize:Nnn` and `\coffin_resize:cnm`. These functions are documented on page 1.)

`\coffin_resize_common:Nnn`

The poles and corners of the coffin are scaled to the appropriate places before actually resizing the underlying box.

```
21 \cs_new_protected_nopar:Npn \coffin_resize_common:Nnn #1#2#3
22 {
23   \prop_map_inline:cn { l_coffin_corners_ \int_value:w #1 _prop }
24   { \coffin_scale_corner:Nnnn #1 {##1} ##2 }
25   \prop_map_inline:cn { l_coffin_poles_ \int_value:w #1 _prop }
26   { \coffin_scale_pole:Nnnnnn #1 {##1} ##2 }
```

Negative  $x$ -scaling values will place the poles in the wrong location: this is corrected here.

```
27   \fp_compare:NNNT \l_coffin_scale_x_fp < \c_zero_fp
28   {
29     \prop_map_inline:cn { l_coffin_corners_ \int_value:w #1 _prop }
30     { \coffin_x_shift_corner:Nnnn #1 {##1} ##2 }
31     \prop_map_inline:cn { l_coffin_poles_ \int_value:w #1 _prop }
32     { \coffin_x_shift_pole:Nnnnnn #1 {##1} ##2 }
33   }
34   \coffin_end_user_dimensions:
35 }
```

(End definition for `\coffin_resize_common:Nnn`. This function is documented on page ??.)

`\coffin_scale:Nnn` For scaling, the opposite calculation is done to find the new dimensions for the coffin.  
`\coffin_scale:cnn` Only the total height is needed, as this is the shift required for corners and poles. The scaling is done the T<sub>E</sub>X way as this works properly with floating point values without needing to use the `fp` module.

```

36 \cs_new_protected_nopar:Npn \coffin_scale:Nnn #1#2#3
37 {
38   \coffin_set_user_dimensions:N #1
39   \fp_set:Nn \l_coffin_scale_x_fp {#2}
40   \fp_set:Nn \l_coffin_scale_y_fp {#3}
41   \fp_compare:NNTF \l_coffin_scale_y_fp > \c_zero_fp
42   { \l_coffin_scaled_total_height_dim #3 \TotalHeight }
43   { \l_coffin_scaled_total_height_dim -#3 \TotalHeight }
44   \fp_compare:NNTF \l_coffin_scale_x_fp > \c_zero_fp
45   { \l_coffin_scaled_width_dim -#2 \Width }
46   { \l_coffin_scaled_width_dim #2 \Width }
47   \hbox_set:Nn #1 { \scalebox {#2} [#3] { \box_use:N #1 } }
48   \coffin_resize_common:Nnn #1
49   { \l_coffin_scaled_width_dim } { \l_coffin_scaled_total_height_dim }
50 }
51 \cs_generate_variant:Nn \coffin_scale:Nnn { c }

```

*(End definition for `\coffin_scale:Nnn` and `\coffin_scale:cnn`. These functions are documented on page 1.)*

`\coffin_scale_vector:nnNN` This functions scales a vector from the origin using the pre-set scale factors in  $x$  and  $y$ . This is a much less complex operation than rotation, and as a result the code is a lot clearer.

```

52 \cs_new_protected_nopar:Npn \coffin_scale_vector:nnNN #1#2#3#4
53 {
54   \fp_set_from_dim:Nn \l_coffin_tmp_fp {#1}
55   \fp_mul:Nn \l_coffin_tmp_fp { \l_coffin_scale_x_fp }
56   \dim_set:Nn #3 { \fp_to_dim:N \l_coffin_tmp_fp }
57   \fp_set_from_dim:Nn \l_coffin_tmp_fp {#2}
58   \fp_mul:Nn \l_coffin_tmp_fp { \l_coffin_scale_y_fp }
59   \dim_set:Nn #4 { \fp_to_dim:N \l_coffin_tmp_fp }
60 }

```

*(End definition for `\coffin_scale_vector:nnNN`. This function is documented on page ??.)*

`\coffin_scale_corner:Nnnn` Scaling both corners and poles is a simple calculation using the preceding vector scaling.  
`\coffin_scale_pole:Nnnnnn`

```

61 \cs_new_protected_nopar:Npn \coffin_scale_corner:Nnnn #1#2#3#4
62 {
63   \coffin_scale_vector:nnNN {#3} {#4} \l_coffin_x_dim \l_coffin_y_dim
64   \prop_put:cnx { l_coffin_corners_ \int_value:w #1 _prop } {#2}
65   { { \dim_use:N \l_coffin_x_dim } { \dim_use:N \l_coffin_y_dim } }
66 }
67 \cs_new_protected_nopar:Npn \coffin_scale_pole:Nnnnnn #1#2#3#4#5#6

```

```

68 {
69   \coffin_scale_vector:nnNN {#3} {#4} \l_coffin_x_dim \l_coffin_y_dim
70   \coffin_set_pole:Nnx #1 {#2}
71   {
72     { \dim_use:N \l_coffin_x_dim } { \dim_use:N \l_coffin_y_dim }
73     {#5} {#6}
74   }
75 }

```

(End definition for `\coffin_scale_corner:Nnnn`. This function is documented on page ??.)

`\coffin_x_shift_corner:Nnnn` These functions correct for the  $x$  displacement that takes place with a negative horizontal  
`\coffin_x_shift_pole:Nnnnnn` scaling.

```

76 \cs_new_protected_nopar:Npn \coffin_x_shift_corner:Nnnn #1#2#3#4
77 {
78   \prop_put:cnx { l_coffin_corners_ \int_value:w #1 _prop } {#2}
79   {
80     { \dim_eval:n { #3 + \box_wd:N #1 } } {#4}
81   }
82 }
83 \cs_new_protected_nopar:Npn \coffin_x_shift_pole:Nnnnnn #1#2#3#4#5#6
84 {
85   \prop_put:cnx { l_coffin_poles_ \int_value:w #1 _prop } {#2}
86   {
87     { \dim_eval:n { #3 + \box_wd:N #1 } } {#4}
88     {#5} {#6}
89   }
90 }

```

(End definition for `\coffin_x_shift_pole:Nnnnnn`. This function is documented on page ??.)

91 `\endpackage`

## Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

<b>B</b>		<code>\coffin_resize:cnm</code> . . . . . <i>1</i> , <u>8</u>
<code>\box_use:N</code> . . . . . <i>17</i> , <i>47</i>	<code>\coffin_resize:Nnn</code> . . . . . <i>1</i> , <u>8</u> , <i>8</i> , <i>20</i>	
<code>\box_wd:N</code> . . . . . <i>80</i> , <i>87</i>	<code>\coffin_resize_common:Nnn</code> . <i>18</i> , <u>21</u> , <i>21</i> , <i>48</i>	
<b>C</b>		<code>\coffin_scale:cnm</code> . . . . . <i>1</i> , <u>36</u>
	<code>\coffin_scale:Nnn</code> . . . . . <i>1</i> , <u>36</u> , <i>36</i> , <i>51</i>	
<code>\c_zero_fp</code> . . . . . <i>27</i> , <i>41</i> , <i>44</i>	<code>\coffin_scale_corner:Nnnn</code> . . . . <i>24</i> , <u>61</u> , <i>61</i>	
<code>\coffin_end_user_dimensions:</code> . . . . . <i>34</i>	<code>\coffin_scale_pole:Nnnnnn</code> . . . . <i>26</i> , <u>61</u> , <i>67</i>	

\coffin_scale_vector:nnNN .	52, 52, 63, 69	I	
\coffin_set_pole:Nnx	70	\int_value:w	23, 25, 29, 31, 64, 78, 85
\coffin_set_user_dimensions:N	10, 38	L	
\coffin_x_shift_corner:Nnnn	30, 76, 76	\l_coffin_scale_x_fp	
\coffin_x_shift_pole:Nnnnnn	32, 76, 83	.....	4, 4, 11, 13, 27, 39, 44, 55
\cs_generate_variant:Nn	20, 51	\l_coffin_scale_y_fp	4, 5, 14, 16, 40, 41, 58
\cs_new_protected_nopar:Npn	8, 21, 36, 52, 61, 67, 76, 83	\l_coffin_scaled_total_height_dim	
		.....	6, 6, 42, 43, 49
D		\l_coffin_scaled_width_dim	6, 7, 45, 46, 49
\dim_eval:n	80, 87	\l_coffin_tmp_fp	12, 13, 15, 16, 54–59
\dim_new:N	6, 7	\l_coffin_x_dim	63, 65, 69, 72
\dim_set:Nn	56, 59	\l_coffin_y_dim	63, 65, 69, 72
\dim_use:N	65, 72	P	
E		\prop_map_inline:cn	23, 25, 29, 31
\ExplFileDate	3	\prop_put:cnx	64, 78, 85
\ExplFileDescription	3	\ProvidesExplPackage	2
\ExplFileName	3	R	
\ExplFileVersion	3	\resizebox	17
F		S	
\fp_compare:NNNT	27	\scalebox	47
\fp_compare:NNNTF	41, 44	T	
\fp_div:Nn	13, 16	\TotalHeight	15, 42, 43
\fp_mul:Nn	55, 58	W	
\fp_new:N	4, 5	\Width	12, 45, 46
\fp_set:Nn	39, 40		
\fp_set_from_dim:Nn	11, 12, 14, 15, 54, 57		
\fp_to_dim:N	56, 59		
H			
\hbox_set:Nn	17, 47		