

The `cellprops` package

CSS-like cell and table properties*

Julien “_FrnchFrpg_” RIVAUD†

Released 2018/06/14

1 `cellprops` documentation

This package reworks the internals of `tabular`, `array`, and similar constructs, and adds a `\cellprops` command accepting CSS-like selectors and properties. It implements the `border-collapse: separate` CSS model.

It depends on `mdwtab`, `xcolor` and of course `expl3` and `xparse`.

`cellprops` default settings mimic the LaTeX layout, that is left and right padding equal to `\tabcolsep` or `\arraycolsep`, zero top and bottom padding, but minimum height and depth corresponding to the table strut box.

I recommend to add globally:

```
cellprops{ td { padding: 1ex; min-height: 0pt; min-depth: 0pt; } }
```

so that you get better-looking tables by default.

1.1 Examples

To produce:

This is text	A_2	A_3	A_4
B1	<i>This is maths</i>	B_3	
C1	C_2	X	Y
D1	D_2	DX	v
E	F	$\int_a^b f(t)dt$	v

you can use:

```
\[
\cellprops{
  td {
    padding: 1ex;
    min-height: 0pt;
```

*This file describes v1.3, last revised 2018/06/14.

†E-mail: frnchfrpg@free.fr

```

        min-depth: 0pt;
        border-style: none solid solid none;
        background-color: black!5!white;
        text-align: center;
    }
    tr:nth-child(even) td {
        background-color: black!15!white;
    }
    td:nth-child(even) {
        background-color: yellow!20!white
    }
    tr:nth-child(even) td:nth-child(even) {
        background-color: yellow!50!white;
    }
    tr:first-child td {
        border-top-style: solid;
    }
    td:first-child {
        border-left-style: solid;
        math-mode: text;
        text-align: left;
    }
}
\begin{array}{nnnn}
    This is text & A_2 & A_3 & A_4 \\
    B1 & This is maths & B_3 & \\
    C1 & C_2 & X & Y \\
    D1 & D_2 & DX & v \\
    E & F & \int_a^b f(t) dt & v \\
\end{array}
\]
```

You can also use the `longtable` environment:

aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb

aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb
aaaaa	baaaa	caaaaa	dbbbb

This table has been produced by:

```
cellprops{
  td { border: thin solid black; }
  tr:nth-child(4n) td:first-child,
  tr:nth-child(4n+1) td:nth-child(2),
  tr:nth-child(4n+2) td:nth-child(3),
  tr:nth-child(4n+3) td:nth-child(4) {
    border: thick solid red;
  }
}

begin{longtable}{nnnn}
  aaaaa & baaaa & caaaaa & dbbbb \\
  ...
  aaaaa & baaaa & caaaaa & dbbbb \\
end{longtable}
```

1.2 Usage guide

```
<usage>:      '\cellprops{' [ <selectors> '{' <properties> '}' ]* '}'
<selectors>:  <selector> [, <selectors> ]
<selector>:   [<environment> ' ' ] <element1>
<element1>:   'table' | 'tr' [<pseudo-class>] [ ' ' <element2> ] <element2>
<element2>:   'td' [<pseudo-class>]
<pseudo-class>: 'nth-child(' <nth> ')'
<nth>:        <number> | 'odd' | 'even' | <number> 'n+' <number>
<properties>: [ <property> ';' ]*
```

```

<property>: 'padding: ' ( <dimension> ) {1,4} |
'padding-top: ' <dimension> |
'padding-right: ' <dimension> |
'padding-bottom: ' <dimension> |
'padding-left: ' <dimension> |
'min-height: ' <dimension> |
'min-depth: ' <dimension> |
'min-width: ' <dimension> |
'text-align: ' ( 'left' | 'right' | 'center' ) |
'math-mode: ' ( 'text' | 'math' | 'auto' ) |
'color: ' <color> |
'background-color: ' ( <color> | 'transparent' ) |
'border: ' [ <bd-width> ] [ <bd-style> ] [ <color> ] |
'border-top: ' [ <bd-width> ] [ <bd-style> ] [ <color> ] |
'border-right: ' [ <bd-width> ] [ <bd-style> ] [ <color> ] |
'border-bottom: ' [ <bd-width> ] [ <bd-style> ] [ <color> ] |
'border-left: ' [ <bd-width> ] [ <bd-style> ] [ <color> ] |
'border-width: ' ( <bd-width> ) {1,4} |
'border-top-width: ' <bd-width> ) |
'border-right-width: ' <bd-width> ) |
'border-bottom-width: ' <bd-width> ) |
'border-left-width: ' <bd-width> ) |
'border-style: ' ( <bd-style> ) {1,4} |
'border-top-style: ' <bd-style> ) |
'border-right-style: ' <bd-style> ) |
'border-bottom-style: ' <bd-style> ) |
'border-left-style: ' <bd-style> ) |
'border-color: ' ( <color> ) {1,4} |
'border-top-color: ' <color> ) |
'border-right-color: ' <color> ) |
'border-bottom-color: ' <color> ) |
'border-left-color: ' <color> )

<color>: 'inherit' | <xcolor-expression> |
'rgb(' <red-0-255> ', ' <green-0-255> ', ' <blue-0-255> ' ) ' |
'hsl(' <hue-0-360> ', ' <sat-0-1> ', ' <lum-0-1> ' ) '

```

Most of these properties are straight-forward. You should check a CSS documentation to get more information. A very good source is the Mozilla Developer Network.

Here are the supported column types:

- **n**: The most basic cell type, hbox, honoring all properties.
- **l**, **c**, and **r**: Same as **n** but with forced **text-align**.
- **M**<coltype>: Same as <coltype> but enforces **math-mode: math**. The net effect is that **Mc** will create a centered column whose contents are in non-display math mode.
- **T**<coltype>: Same as <coltype> but enforces **math-mode: text**.
- **p**{<width>}, **m**<width> and **b**<width>: parbox cell with the corresponding vertical alignment (**\vtop**, **\vcenter** or **\vbox**).

- `*{<count>}{<coltypes>}`: same as in `array` or `mdwtab`.
- `>{<prefix>}` and `<{<suffix>}`: same as in `array` or `mdwtab`.
- You can try to use constructs of `array` or `mdwtab`, but they might alter the function of `cellprops`. Most should be fine though.

The intended usage is to use `n`-type columns and set the properties with CSS, but \LaTeX -like columns in the preamble are often less verbose.

Details for some properties:

- `math-mode`: `auto` means that the cell will be in math mode in environments `array`, `matrix`, ..., and in text mode in environments like `tabular`, ...
- `background-color` is only painted on the cell, and `transparent` actually means `inherit` except that if all values encountered are `inherit/transparent` no background is painted at all. That means that (currently) you cannot paint a row in some color and rely on transparency to have it bleed through a cell background.
- There are no columns in the CSS object model so you have to use `td:nth-child()` to select a column. Currently, cells spanning several columns actually increase the child count by the number of column they span, so that `nth-child` can still be used to select columns. This is not consistent with the CSS specification.
- Any `:nth-child(A n +B)` or `:nth-child(A n)` or `:nth-child(B)` is supported, with arbitrary A and B . The performance will slowly degrade the more different A are active (but if in a \TeX group, they will become inactive again when leaving the group). A *big difference* with the CSS specification is that currently n is not enforced non-negative. In particular `:nth-child(2 n +8)` *will* match for the second element. This also prevents tricks like `:nth-child(- n +3)`. I am investigating how to handle those without slowing down the general case.

1.3 Compatibility

This package has been tested compatible with `diagbox`, `spreadtab`, `colcell`. Compatibility with `longtable` has been specifically taken care of, provided `cellprops` is loaded afterwards. Table packages that only introduce new column types should be loaded after `mdwtab`, so either you load `mdwtab` manually and load your package in between `mdwtab` and `cellprops`, or you load your package after `cellprops` (provided it doesn't overwrite the machinery).

1.4 TODO

Add a test suite with compatibility tests. Improve the documentation, and test more \LaTeX table constructs and preamble column types. Enforce $n \geq 0$ in `:nth-child` selectors to match the CSS specification.

2 cellprops implementation

```

1 <*package>
2 <@@=cellprops>
3 \ProvidesExplPackage

```

```

4   {\ExplFileName}{\ExplFileDate}{\ExplFileVersion}{\ExplFileDescription}
5
6   \RequirePackage{xparse}
7   \RequirePackage{xcolor}

```

2.1 Loading and fixing mdwtab

There is a bug in the command `\colpop` of `mdwtab`: instead of just popping one name in the stack of column sets currently used, it empties it completely because one `\expandafter` is missing. This is proof that not many package authors really use this API as recommended by Mark WOODING... We thus load `mdwtab` and fix `\colpop`.

```

8   \RequirePackage{mdwtab}
9   \cs_set_nopar:Npn \tab@pop #1 { \tl_set:Nx #1 { \tl_tail:N #1 } }
10
11

```

2.2 Parsing CSS properties

Properties are parsed once at setting time, by expandable parsers that leave definitions in the input stream. All these resulting definitions are saved in a token list that will be expanded when we need the values. The goal is to have multiple token lists for multiple contexts, yet not to do the full parsing dance once per cell.

We first define a generic setter which just uses `\l__cellprops_property_value_<name>_tl` to store the value of the property. We define getters, one that leaves the value in the stream, and one saving the value in a token list.

```

12  \cs_new:Nn \__cellprops_generic_setter:nnn {
13      \exp_not:N \tl_set:Nn
14      \exp_not:c { l__cellprops_property_value_#2_tl }
15      {#1 {#3}}
16  }
17
18  \cs_set_nopar:Nn \__cellprops_get_property:n {
19      \tl_use:c { l__cellprops_property_value_#1_tl }
20  }
21
22  \cs_new_protected_nopar:Nn \__cellprops_get_property:nN {
23      \tl_if_exist:cTF { l__cellprops_property_value_#1_tl } {
24          \tl_set_eq:Nc #2 { l__cellprops_property_value_#1_tl }
25      }{
26          \tl_clear:N #2
27      }
28  }

```

(End definition for `\l__cellprops_property_value_<name>_tl` and others.)

The control sequence `__cellprops_property_type_<name>:nn` holds the setter for the property *<name>*. It can be set by the following helper:

```

29  \cs_new_protected:Nn \__cellprops_define_properties:nn {
30      \clist_map_inline:nn {#2} {
31          \cs_set:cpn { __cellprops_property_type_##1:nn } {#1}
32      }
33  }

```

(End definition for `__cellprops_property_type_<name>:nn` and `__cellprops_define_properties:nn`.)

`__cellprops_use_setter:nn` Sometimes we need to use a setter right away rather than save its action somewhere. The following helper does that with an x-expansion.

```

34 \cs_new:Nn \__cellprops_delegate_setter:nn {
35   \use:c {\__cellprops_property_type_#1:nn} {#1} {#2}
36 }
37 \cs_new_protected:Nn \__cellprops_use_setter:nn {
38   \use:x {
39     \__cellprops_delegate_setter:nn {#1} {#2}
40   }
41 }

```

(End definition for `__cellprops_use_setter:nn`.)

```

42 \cs_new_protected:Nn \__cellprops_parse_properties:Nn {
43   \tl_clear:N #1
44   \seq_set_split:Nnn \l_tmpa_seq {;} {#2}
45   \seq_map_inline:Nn \l_tmpa_seq {
46     \tl_if_empty:nF {##1} {
47       \exp_args:NNV \seq_set_split:Nnn \l_tmpb_seq \c_colon_str {##1}
48       \int_compare:nNnTF {\seq_count:N \l_tmpb_seq} = \c_two {
49         \seq_get_left:NN \l_tmpb_seq \l_tmpa_tl
50         \exp_args:NNV \str_set:Nn \l_tmpa_str \l_tmpa_tl
51         \seq_get_right:NN \l_tmpb_seq \l_tmpa_tl
52         \cs_if_exist:cTF { \__cellprops_property_type_\l_tmpa_str :nn } {
53           \tl_put_right:Nx #1 {
54             \exp_args:NVV \__cellprops_delegate_setter:nn
55               \l_tmpa_str \l_tmpa_tl
56           }
57         }{
58           % TODO: ERROR-no property with that name
59         }
60       }{
61         % TODO: ERROR-too many : or none at all
62       }
63     }
64   }
65 }
66
67 \cs_new:Nn \__cellprops_fourval_setter:nnnnnn {
68   \__cellprops_fourval_setter_aux:w
69     {#1}{#2}{#3}{#4}#6~{\q_no_value}~{\q_no_value}~{\q_no_value}~{\q_stop}
70 }
71 \cs_new:Npn \__cellprops_fourval_setter_aux:w #1#2#3#4#5~#6~#7~#8~#9\q_stop {
72   \__cellprops_delegate_setter:nn {#1} {#5}
73   \quark_if_no_value:nTF {#6} {
74     \__cellprops_delegate_setter:nn {#2} {#5}
75     \__cellprops_delegate_setter:nn {#4} {#5}
76   }{
77     \__cellprops_delegate_setter:nn {#2} {#6}
78     \quark_if_no_value:nTF {#8} {
79       \__cellprops_delegate_setter:nn {#4} {#6}
80     }{
81       \__cellprops_delegate_setter:nn {#4} {#8}

```

```

82     }
83   }
84   \quark_if_no_value:nTF {#7} {
85     \__cellprops_delegate_setter:nn {#3} {#5}
86   }{
87     \__cellprops_delegate_setter:nn {#3} {#7}
88   }
89 }
90
91 \cs_new_protected:Nn \__cellprops_define_fourval_properties:nnnnnn {
92   \__cellprops_define_properties:nn {#1} { #3, #4, #5, #6 }
93   \__cellprops_define_properties:nn {
94     \__cellprops_fourval_setter:nnnnnn {#3}{#4}{#5}{#6}
95   }{
96     #2
97   }
98 }
99
100 \tl_const:Nn \c__cellprops_inherit_color_tl { \q_nil }
101
102 \cs_new_nopar:Nn \__cellprops_color_setter:nn {
103   \str_if_eq:nnTF {#2} {inherit} {
104     \__cellprops_generic_setter:nnn \exp_not:n {#1} {\c__cellprops_inherit_color_tl}
105   }{
106     \str_case_x:nnF { \str_range:nnn {#2} {1} {4} } {
107       {rgb} {
108         \__cellprops_generic_setter:nnn \use:n {#1} {
109           \exp_not:n {\color[RGB]} {\str_range:nnn {#2} {5} {-2}}
110         }
111       }{
112         \__cellprops_generic_setter:nnn \use:n {#1} {
113           \exp_not:n {\color[Hsb]} {\str_range:nnn {#2} {5} {-2}}
114         }
115       }{
116         \__cellprops_generic_setter:nnn \exp_not:n {#1} {
117           \color{#2}
118         }
119     }
120   }
121 }
122 \cs_new_nopar:Nn \__cellprops_bgcolor_setter:nn {
123   \str_if_eq:nnTF {#2} {transparent} {
124     \__cellprops_color_setter:nn {#1} {inherit}
125   }{
126     \__cellprops_color_setter:nn {#1} {#2}
127   }
128 }
129
130 \cs_new_nopar:Nn \__cellprops_linewidth_setter:nn {
131   \str_case:nnF {#2} {
132     {thin} { \__cellprops_generic_setter:nnn \exp_not:n {#1} { \fboxrule } }
133     {medium} { \__cellprops_generic_setter:nnn \exp_not:n {#1} { 2\fboxrule } }
134     {thick} { \__cellprops_generic_setter:nnn \exp_not:n {#1} { 3\fboxrule } }
135   }{

```

```

136         \_cellprops_generic_setter:nnn \exp_not:n {#1} {#2}
137     }
138 }
139
140 \cs_new_nopar:Nn \_cellprops_border_setter:nn {
141     \_cellprops_border_setter_aux:nw
142     {#1}#2~{\q_no_value}~{\q_no_value}~\q_stop
143 }
144 \cs_new:Npn \_cellprops_border_setter_aux:nw #1#2~#3~#4~#5\q_stop {
145     \quark_if_no_value:nTF {#4} {
146         \_cellprops_border_setter_isstyle:nTF {#2} {
147             \_cellprops_delegate_setter:nn {#1-width} {thin}
148             \_cellprops_delegate_setter:nn {#1-style} {#2}
149             \quark_if_no_value:nTF {#3} {
150                 \_cellprops_delegate_setter:nn {#1-color} {inherit}
151             }{
152                 \_cellprops_delegate_setter:nn {#1-color} {#3}
153             }
154         }{
155             \quark_if_no_value:nTF {#3} {
156                 %% TODO: Error, one no-style value, ambiguous
157             }{
158                 \_cellprops_border_setter_isstyle:nTF {#3} {
159                     \_cellprops_delegate_setter:nn {#1-width} {#2}
160                     \_cellprops_delegate_setter:nn {#1-style} {#3}
161                     \_cellprops_delegate_setter:nn {#1-color} {inherit}
162                 }{
163                     \_cellprops_delegate_setter:nn {#1-width} {#2}
164                     \_cellprops_delegate_setter:nn {#1-style} {none}
165                     \_cellprops_delegate_setter:nn {#1-color} {#3}
166                 }
167             }
168         }
169     }{
170         \_cellprops_delegate_setter:nn {#1-width} {#2}
171         \_cellprops_delegate_setter:nn {#1-style} {#3}
172         \_cellprops_delegate_setter:nn {#1-color} {#4}
173     }
174 }
175 \cs_new:Npn \_cellprops_border_setter_isstyle:nTF #1 {
176     \str_case:nnTF {#1} {
177         {none}{ } {hidden}{ } {dotted}{ } {dashed}{ } {solid}{ }
178         {double}{ } {groove}{ } {ridge}{ } {inset}{ } {outset}{ }
179     }
180 }
181
182 \_cellprops_define_properties:nn {
183     \_cellprops_generic_setter:nnn \exp_not:n
184 }{
185     min-height,
186     min-depth,
187     min-width,
188 }
189

```

```

190 \__cellprops_define_fourval_properties:nnnnnn
191   { \__cellprops_generic_setter:nnn \exp_not:n }
192   {padding}
193   {padding-top}{padding-right}{padding-bottom}{padding-left}
194
195 \__cellprops_define_properties:nn {
196   \__cellprops_generic_setter:nnn \tl_to_str:n
197 }{
198   text-align,
199   math-mode,
200 }
201
202 \__cellprops_define_properties:nn {
203   \__cellprops_color_setter:nn
204 }{
205   color,
206 }
207
208 \__cellprops_define_properties:nn {
209   \__cellprops_bgcolor_setter:nn
210 }{
211   background-color,
212 }
213
214 \__cellprops_define_fourval_properties:nnnnnn
215   { \__cellprops_linewidth_setter:nn }
216   {border-width}
217   {border-top-width}{border-right-width}
218   {border-bottom-width}{border-left-width}
219
220 \__cellprops_define_fourval_properties:nnnnnn
221   { \__cellprops_generic_setter:nnn \tl_to_str:n }
222   {border-style}
223   {border-top-style}{border-right-style}
224   {border-bottom-style}{border-left-style}
225
226 \__cellprops_define_fourval_properties:nnnnnn
227   { \__cellprops_color_setter:nn }
228   {border-color}
229   {border-top-color}{border-right-color}
230   {border-bottom-color}{border-left-color}
231
232 \__cellprops_define_properties:nn {
233   \__cellprops_border_setter:nn
234 }{
235   border, border-top, border-right, border-bottom, border-left
236 }
237
238 \NewDocumentCommand \cellprops { m } {
239   \__cellprops_parse_css:n {#1}
240 }
241
242 \cs_new_protected:Nn \__cellprops_parse_css:n {
243   \__cellprops_parse_css:w #1 \q_mark {\q_nil} \q_stop

```

```

244 }
245
246 \tl_new:N \l__cellprops_parse_tmp_tl
247 \NewDocumentCommand \__cellprops_parse_css:w { lmu{\q_stop} } {
248   \quark_if_nil:nF {#2} {
249     \__cellprops_parse_properties:Nn \l__cellprops_parse_tmp_tl {#2}
250     \clist_map_inline:nn {#1} {
251       \__cellprops_parse_css_addprops:nV {##1} \l__cellprops_parse_tmp_tl
252     }
253     \__cellprops_parse_css:w #3 \q_stop
254   }
255 }
256
257
258 \seq_new:N \l__cellprops_parse_selector_seq
259 \tl_new:N \l__cellprops_parse_desc_tl
260
261 \str_const:Nn \c__cellprops_parse_nthchild_str { :nth-child( }
262 \prop_new:N \c__cellprops_parse_replace_prop
263 \prop_put:Nnn \c__cellprops_parse_replace_prop { :first-child } { :nth-child(1) }
264
265 \cs_new_protected:Nn \__cellprops_parse_selector:Nn {
266   \str_set:Nx \l_tmpa_str {#2}

```

Replace some aliases with their meaning:

```

267   \prop_map_inline:Nn \c__cellprops_parse_replace_prop {
268     \use:x {
269       \exp_not:n { \tl_replace_all:Nnn \l_tmpa_str }
270       { \tl_to_str:n { ##1 } } { \tl_to_str:n { ##2 } }
271     }
272   }

```

Replace all spaces by `\q_stop` to defeat the space removal feature of `\seq_set_split:Nnn`.

```

273   \tl_replace_all:Nnn \l_tmpa_str {~} {\q_stop}
274   \exp_args:NNVV
275   \seq_set_split:Nnn \l_tmpa_seq \c__cellprops_parse_nthchild_str \l_tmpa_str
276   \seq_pop_left:NN \l_tmpa_seq \l_tmpa_tl

```

And replace them back.

```

277   \tl_replace_all:Nnn \l_tmpa_tl {\q_stop} {~}
278   \seq_clear:N \l__cellprops_parse_selector_seq
279   \seq_put_right:NV \l__cellprops_parse_selector_seq \l_tmpa_tl
280   \seq_map_inline:Nn \l_tmpa_seq {
281     \tl_set:Nn \l_tmpa_tl { ##1 }

```

And replace them back again.

```

282     \tl_replace_all:Nnn \l_tmpa_tl {\q_stop} {~}

```

Now replace the first closing parenthesis by `\q_stop\prg_do_nothing:` to use `\q_stop` as a delimiter for `\seq_set_split:Nnn` and thus split at most once. Note that here the space trimming feature is desired for the left part, but not just at the right of the parenthesis, so `\prg_do_nothing:` will act as a guard and will be removed afterwards.

```

283     \tl_replace_once:Nnn \l_tmpa_tl { ) } { \q_stop\prg_do_nothing: }
284     \seq_set_split:NnV \l_tmpa_seq { \q_stop } \l_tmpa_tl

```

```

285     \seq_pop_right:NN \l_tmpa_seq \l__cellprops_parse_desc_tl
286     \tl_replace_once:Nnn \l__cellprops_parse_desc_tl { \prg_do_nothing: } {}
287     \seq_get_left:NNT \l_tmpa_seq \l_tmpa_tl {
288         \exp_args:NNV \__cellprops_parse_nth:Nn \l_tmpa_tl \l_tmpa_tl
289         \tl_put_left:Nn \l__cellprops_parse_desc_tl { ) }
290         \tl_put_left:NV \l__cellprops_parse_desc_tl \l_tmpa_tl
291     }
292     \seq_put_right:NV \l__cellprops_parse_selector_seq \l__cellprops_parse_desc_tl
293 }
294 \tl_set:Nx #1 {
295     \exp_args:NNV \seq_use:Nn
296         \l__cellprops_parse_selector_seq \c__cellprops_parse_nthchild_str
297 }
298 }
299
300 \str_const:Nn \c__cellprops_parse_n_str {n}
301 \seq_new:N \l__cellprops_used_nth_factors_seq
302 \cs_new_protected:Nn \__cellprops_parse_nth:Nn {
303     \str_case:nnF {#2} {
304         {even} { \str_set:Nn \l_tmpa_str {2n} }
305         {odd} { \str_set:Nn \l_tmpa_str {2n+1} }
306     }{
307         \str_set:Nn \l_tmpa_str {#2}
308     }
309     \exp_args:NNVV
310     \seq_set_split:Nnn \l_tmpa_seq \c__cellprops_parse_n_str \l_tmpa_str
311     \seq_pop_right:NN \l_tmpa_seq \l_tmpa_tl
312     \int_set:Nn \l_tmpb_int { 0\l_tmpa_tl }
313     \seq_get_left:NNTF \l_tmpa_seq \l_tmpa_tl {
314         \int_set:Nn \l_tmpa_int { 0\l_tmpa_tl }
315     }{
316         \int_zero:N \l_tmpa_int
317     }
318     \int_compare:nNnTF \l_tmpa_int = { 0 } {
319         \tl_set:Nx #1 { \int_use:N \l_tmpb_int }
320     }{
321         \int_set:Nn \l_tmpb_int { \int_mod:nn {\l_tmpb_int} {\l_tmpa_int} }
322         \tl_set:Nx #1 {
323             \int_use:N \l_tmpa_int \exp_not:V \c__cellprops_parse_n_str
324             + \int_use:N \l_tmpb_int }
325         \seq_put_right:Nx
326             \l__cellprops_used_nth_factors_seq { \int_use:N \l_tmpa_int }
327     }
328 }
329
330 \cs_new_protected:Npn \__cellprops_parse_css_addprops:nV #1 #2 {
331     \__cellprops_parse_selector:Nn \l_tmpa_tl {#1}
332     \tl_set:Nx \l_tmpa_tl { l__cellprops_property_group\l_tmpa_tl _tl }
333     \tl_if_exist:cF { \l_tmpa_tl } { \tl_clear:c { \l_tmpa_tl } }
334     \tl_put_right:cV { \l_tmpa_tl } #2
335 }
336
337 \cs_set_protected:Nn \__cellprops_recall_properties:n {
338     \tl_if_exist:cT { l__cellprops_property_group_#1_tl } {

```

```

339         \tl_use:c { l__cellprops_property_group_#1_tl }
340     }
341     \clist_map_inline:nn { \@currenvir } {
342         \tl_if_exist:cT { l__cellprops_property_group_##1~#1_tl } {
343             \tl_use:c { l__cellprops_property_group_##1~#1_tl }
344         }
345     }
346 }
347
348 \dim_new:N \l__cellprops_colsep_dim
349 \dim_new:N \l__cellprops_strut_ht_dim
350 \dim_new:N \l__cellprops_strut_dp_dim
351
352 \ExplSyntaxOff
353 \cellprops{
354     td {
355         padding: Opt \csname l__cellprops_colsep_dim\endcsname;
356         min-height: \csname l__cellprops_strut_ht_dim\endcsname;
357         min-depth: \csname l__cellprops_strut_dp_dim\endcsname;
358         min-width: Opt;
359         text-align: left;
360         math-mode: auto;
361         color: inherit;
362         background-color: transparent;
363         border: thin none inherit;
364     }
365     tr {
366         color: inherit;
367         background-color: transparent;
368     }
369     table {
370         padding: Opt; % No change at load time
371         color: inherit;
372         background-color: transparent;
373     }
374 }
375 \ExplSyntaxOn
376
377 \int_new:N \g__cellprops_row_int
378 \int_new:N \g__cellprops_col_int
379 \bool_new:N \g__cellprops_inrow_bool
380 \bool_gset_false:N \g__cellprops_inrow_bool
381
382 \box_new:N \l__cellprops_cell_box
383 \skip_new:N \l__cellprops_left_skip
384 \skip_new:N \l__cellprops_right_skip
385 \dim_new:N \g__cellprops_ht_dim
386 \dim_new:N \g__cellprops_dp_dim
387 \tl_new:N \g__cellprops_borders_tl
388
389 \tl_new:N \l__cellprops_restore_tl
390
391 \dim_new:N \l__cellprops_tablepadding_top_dim
392 \dim_new:N \l__cellprops_tablepadding_bottom_dim

```

```

393 \tl_new:N \l__cellprops_color_tl
394 \tl_new:N \l__cellprops_bgcolor_tl
395
396 % To count rows and columns
397 \cs_new_protected:Nn \__cellprops_array_init: {
398   \tl_set:Nx \l__cellprops_restore_tl {
399     \bool_if:NTF \g__cellprops_inrow_bool {
400       \exp_not:n {\bool_gset_true:N \g__cellprops_inrow_bool}
401     }{
402       \exp_not:n {\bool_gset_false:N \g__cellprops_inrow_bool}
403     }
404     \exp_not:n { \int_gset:Nn \g__cellprops_row_int }
405     { \int_use:N \g__cellprops_row_int }
406     \exp_not:n { \int_gset:Nn \g__cellprops_col_int }
407     { \int_use:N \g__cellprops_col_int }
408     \exp_not:n { \dim_gset:Nn \g__cellprops_ht_dim }
409     { \dim_use:N \g__cellprops_ht_dim }
410     \exp_not:n { \dim_gset:Nn \g__cellprops_dp_dim }
411     { \dim_use:N \g__cellprops_dp_dim }
412     \exp_not:n { \tl_gset:Nn \g__cellprops_borders_tl }
413     { \exp_not:V \g__cellprops_borders_tl }
414   }
415   \int_gzero:N \g__cellprops_row_int
416   \bool_gset_false:N \g__cellprops_inrow_bool
417   \tl_gclear:N \g__cellprops_borders_tl
418   \cs_set_eq:NN \__cellprops_orig_tab@readpreamble:n \tab@readpreamble
419   \cs_set_eq:NN \tab@readpreamble \__cellprops_readpreamble:n

```

Zero \col@sep but remember its value for the default padding.

```

420   \dim_set_eq:NN \l__cellprops_colsep_dim \col@sep
421   \dim_zero:N \col@sep

```

Also ignore *extrasep dimensions that are not part of cellprop interface and should be replaced by CSS equivalents.

```

422   \dim_zero:N \tab@extrasep
423   \group_begin:
424     \__cellprops_recall_properties:n {table}
425     \dim_gset:Nn \g_tmpa_dim { \__cellprops_get_property:n {padding-top} }
426     \dim_gset:Nn \g_tmpb_dim { \__cellprops_get_property:n {padding-bottom} }
427     \__cellprops_update_colors:
428     \tl_gset_eq:NN \g_tmpa_tl \l__cellprops_color_tl
429     \tl_gset_eq:NN \g_tmpb_tl \l__cellprops_bgcolor_tl
430   \group_end:
431   \dim_set_eq:NN \l__cellprops_tablepadding_top_dim \g_tmpa_dim
432   \dim_set_eq:NN \l__cellprops_tablepadding_bottom_dim \g_tmpb_dim
433   \tl_set_eq:NN \l__cellprops_color_tl \g_tmpa_tl
434   \tl_set_eq:NN \l__cellprops_bgcolor_tl \g_tmpb_tl
435   \__cellprops_recall_properties:n {tr}
436   \dim_set:Nn \l__cellprops_strut_ht_dim { \box_ht:N \@arstrutbox }
437   \dim_set:Nn \l__cellprops_strut_dp_dim { \box_dp:N \@arstrutbox }
438   \box_clear:N \@arstrutbox
439 }
440
441 \cs_set_nopar:Nn \__cellprops_array_startcontent: {
442   \hlx{s[\l__cellprops_tablepadding_top_dim]}

```

```

443 }
444
445 \cs_set_protected_nopar:Nn \__cellprops_readpreamble:n {
446   \cs_set_eq:NN \tab@readpreamble \__cellprops_orig_tab@readpreamble:n
447   \tl_put_left:Nn \tab@multicol {\__cellprops_startrow:}
448   \tl_put_left:Nn \tab@tabtext {\int_gincr:N \g__cellprops_col_int}
449   \tab@readpreamble{#1}
450   \exp_args:Nx \tab@preamble
451   { \exp_not:N\__cellprops_startrow: \the\tab@preamble \exp_not:N\__cellprops_endrow: }
452 }
453
454
455 \cs_set_eq:NN \__cellprops_orig_array:w \@array
456 \cs_set_protected_nopar:Npn \@array[#1]#2 {
457   \__cellprops_array_init:
458   \__cellprops_orig_array:w [#1]{#2}
459   \__cellprops_array_startcontent:
460 }
461
462 \cs_set_eq:NN \__cellprops_orig_LTmkpream:n \@mkpream
463 \cs_set_protected_nopar:Npn \@mkpream#1 {
464   \group_end:
465   \__cellprops_array_init:
466   \group_begin:
467   \__cellprops_orig_LTmkpream:n {#1}
468 }
469
470 \cs_set_eq:NN \__cellprops_orig_LTarray:w \LT@array
471 \cs_set_protected_nopar:Npn \LT@array [#1]#2 {
472   \__cellprops_orig_LTarray:w [#1]{#2}
473   \__cellprops_array_startcontent:
474 }
475

```

The color inheritance is handled with `\l__cellprops_inherit_color_tl`, `\l__cellprops_color_tl` and `\l__cellprops_bgcolor_tl`. The role of `__cellprops_update_color:Nn` is to set the inherit fallback to the already existing value of #1 then set #1 to the CSS value, which can be the inherit variable.

```

476 \cs_new_protected_nopar:Nn \__cellprops_update_color:Nn {
477   \__cellprops_get_property:nN {#2} \l_tmpa_tl
478   \exp_args:NV \tl_if_eq:NNF \l_tmpa_tl \c__cellprops_inherit_color_tl {
479     \tl_set_eq:NN #1 \l_tmpa_tl
480   }
481 }
482
483 \cs_new_protected_nopar:Nn \__cellprops_update_colors: {
484   \__cellprops_update_color:Nn \l__cellprops_color_tl {color}
485   \__cellprops_update_color:Nn \l__cellprops_bgcolor_tl {background-color}
486 }
487
488 \cs_new_nopar:Nn \__cellprops_end_array:n {
489   \tl_if_empty:NF \g__cellprops_borders_tl { \ }
490   \crrc
491   \hbox{s[\l__cellprops_tablepadding_bottom_dim]}

```

```

492     #1
493     \tl_use:N \l__cellprops_restore_tl
494 }
495
496 \cs_set_eq:NN \__cellprops_orig_endarray: \endarray
497 \cs_set_nopar:Npn \endarray {
498     \__cellprops_end_array:n { \__cellprops_orig_endarray: }
499 }
500 \cs_set_eq:NN \endtabular \endarray
501 \cs_set_eq:cN {endtabular*} \endarray
502
503 \cs_set_eq:NN \__cellprops_orig_endLT: \endlongtable
504 \cs_set_nopar:Npn \endlongtable {
505     \__cellprops_end_array:n { \__cellprops_orig_endLT: }
506 }
507
508 \cs_new_protected_nopar:Nn \__cellprops_startrow: {
509     \bool_if:NF \g__cellprops_inrow_bool {
510         \bool_gset_true:N \g__cellprops_inrow_bool
511         \int_gincr:N \g__cellprops_row_int
512         \int_gset_eq:NN \g__cellprops_col_int \c_one
513         \dim_gzero:N \g__cellprops_ht_dim
514         \dim_gzero:N \g__cellprops_dp_dim
515     }
516 }
517
518 \cs_new_protected_nopar:Nn \__cellprops_endrow: {
519     \bool_if:NT \g__cellprops_inrow_bool {
520         \bool_gset_false:N \g__cellprops_inrow_bool
521     }
522 }
523
524 \cs_new_protected_nopar:Nn \__cellprops_cr:n {
525     \__cellprops_endrow:
526     \tl_if_empty:NF \g__cellprops_borders_tl {
527         \cr
528         \noalign{\nobreak}
529         \tl_use:N \g__cellprops_borders_tl
530         \tl_gclear:N \g__cellprops_borders_tl
531     }
532     \cr
533     \__cellprops_fix_valign_end:n {#1}
534     \use_none:n
535 }
536
537 \cs_set_protected_nopar:Npn \tab@tabcr #1#2 { \__cellprops_cr:n {#2} }
538 \cs_set_protected_nopar:Npn \@xargarraycr #1 { \__cellprops_cr:n {#1} }
539 \cs_set_protected_nopar:Npn \@yargarraycr #1 { \__cellprops_cr:n {#1} }
540 \tl_if_exist:NT \LT@echunk {
541     \tl_put_left:Nn \LT@echunk {
542         \tl_if_empty:NF \g__cellprops_borders_tl { \\ }
543     }
544 }
545

```

```

546 \cs_set_eq:NN \__cellprops_orig_multicolumn:w \multicolumn
547 \cs_set:Npn \multicolumn#1#2#3 {
548   \__cellprops_orig_multicolumn:w {#1}{#2}{
549     #3
550     \int_gadd:Nn \g__cellprops_col_int {#1}
551   }
552 }
553
554 \cs_new_nopar:Nn \__cellprops_fix_valign_end:n {
555   \noalign{
556     \dim_set:Nn \l_tmpa_dim {#1}
557     \skip_vertical:n {\l_tmpa_dim}
558     \exp_args:NV \tl_if_eq:nnTF \tab@hlstate {b} {
559       \dim_gadd:Nn \tab@endheight { \g__cellprops_dp_dim + \l_tmpa_dim }
560     }{
561       \int_compare:nNnT \g__cellprops_row_int = \c_one {
562         \dim_gadd:Nn \tab@endheight { \g__cellprops_ht_dim }
563       }
564     }
565   }
566 }
567
568 \colpush{tabular}
569
570 \coldef n{\tabcoltype{
571   \__cellprops_begincell:n{
572 }{
573   \__cellprops_endcell:
574 }}
575 \coldef l{\tabcoltype{
576   \__cellprops_begincell:n
577   {\__cellprops_use_setter:nn {text-align} {left}}
578 }{
579   \__cellprops_endcell:
580 }}
581 \coldef c{\tabcoltype{
582   \__cellprops_begincell:n
583   {\__cellprops_use_setter:nn {text-align} {center}}
584 }{
585   \__cellprops_endcell:
586 }}
587 \coldef r{\tabcoltype{
588   \__cellprops_begincell:n
589   {\__cellprops_use_setter:nn {text-align} {right}}
590 }{
591   \__cellprops_endcell:
592 }}
593 \coldef M#1{\__cellprops_MTcol:nn {math}{#1}}
594 \coldef T#1{\__cellprops_MTcol:nn {text}{#1}}
595 \cs_new_protected_nopar:Nn \__cellprops_MTcol:nn {
596   % TODO: error if align not l, c, or r
597   \exp_args:Nx \tabcoltype {
598     \exp_not:N \__cellprops_begincell:n {
599       \exp_not:n {\__cellprops_use_setter:nn {math-mode} {#1} }

```

```

600         \exp_not:n {\__cellprops_use_setter:nn {text-align}} {
601             \str_case:nn {#2} {
602                 {l} {left}
603                 {c} {center}
604                 {r} {right}
605             }
606         }
607     }
608 }{
609     \__cellprops_endcell:
610 }
611 }
612
613 \coldef p#1{\tabcoltype{
614     \__cellprops_begin_par_cell:nn \vtop {#1}
615 }}{
616     \__cellprops_end_par_cell:n {}
617 }}
618 \coldef m#1{\tabcoltype{
619     \__cellprops_begin_par_cell:nn {\c_math_toggle_token\vcenter} {#1}
620 }}{
621     \__cellprops_end_par_cell:n{\c_math_toggle_token}
622 }}
623 \coldef b#1{\tabcoltype{
624     \__cellprops_begin_par_cell:nn \vbox {#1}
625 }}{
626     \__cellprops_end_par_cell:n {}
627 }}
628
629
630 \colpop
631     Handle various :nth-child() forms.
632 \cs_new_protected_nopar:Nn \__cellprops_seq_nthchild:Nn {
633     \seq_clear:N #1
634     \seq_map_inline:Nn \l__cellprops_used_nth_factors_seq {
635         \seq_put_right:Nx #1 {
636             ##1 n + \int_eval:n{\int_mod:nn{#2}{##1}}
637         }
638     }
639     \seq_put_right:Nx #1 { \int_eval:n{#2} }
640 }
641 \cs_new_protected_nopar:Nn \__cellprops_begincell:n {
642     \__cellprops_begin_raw_cell:n {
643         \__cellprops_seq_nthchild:Nn \l_tmpa_seq { \g__cellprops_row_int }
644         \__cellprops_seq_nthchild:Nn \l_tmpb_seq { \g__cellprops_col_int }
645         \seq_map_inline:Nn \l_tmpa_seq {
646             \__cellprops_recall_properties:n {tr:nth-child(##1)}
647         }
648         \__cellprops_update_colors:
649         \__cellprops_recall_properties:n {td}
650         \__cellprops_recall_properties:n {tr~td}
651         \seq_map_inline:Nn \l_tmpb_seq {
652             \__cellprops_recall_properties:n {td:nth-child(##1)}

```

```

653     }
654     \seq_map_inline:Nn \l_tmpa_seq {
655         \__cellprops_recall_properties:n {tr:nth-child(##1)~td}
656     }
657     \seq_map_inline:Nn \l_tmpa_seq {
658         \seq_map_inline:Nn \l_tmpb_seq {
659             \__cellprops_recall_properties:n {tr:nth-child(##1)~
660                 td:nth-child(###1)}
661         }
662     }
663     \__cellprops_update_colors:
664     #1
665     \hbox_set:Nw \l__cellprops_cell_box
666     \str_case_x:nnF {\__cellprops_get_property:n {math-mode}} {
667         { text } { \tab@btext }
668         { math } { \tab@bmaths }
669     }{% any other treated as |auto|
670         \tab@bgroup
671     }
672 }
673 }
674
675 \cs_new_protected_nopar:Nn \__cellprops_endcell: {
676     \str_case_x:nnF {\__cellprops_get_property:n {math-mode}} {
677         { text } { \tab@etext }
678         { math } { \tab@emaths }
679     }{% any other treated as |auto|
680         \tab@egroup
681     }
682     \hbox_set_end:
683     \__cellprops_end_raw_cell:
684 }
685
686 \cs_new_protected_nopar:Nn \__cellprops_begin_par_cell:nn {
687     \savenotes
688     \__cellprops_begin_raw_cell:n{
689         \hbox_set:Nw \l__cellprops_cell_box
690         #1
691         \bgroup
692         \hsize#2\relax
693         \@arrayparboxrestore
694         \global\@minipagetrue
695         \everypar{
696             \global\@minipagefalse
697             \everypar{}
698         }
699         \__cellprops_recall_properties:n {td~p}
700     }
701 }
702 \cs_new_protected_nopar:Nn \__cellprops_end_par_cell:n {
703     \ifhmode\@maybe@unskip\par\fi
704     \unskip
705     \egroup
706     #1

```

```

707 \hbox_set_end:
708 \_cellprops_end_raw_cell:
709 \spewnotes\hfil
710 }
711
712 \cs_new_protected_nopar:Nn \_cellprops_begin_raw_cell:n {
713   \group_begin:
714   % Additional init code
715   #1
716   % Install the cell color
717   \_cellprops_update_colors:
718   \tl_use:N \l\_cellprops_color_tl
719 }
720
721 \cs_new_protected_nopar:Nn \_cellprops_make_solid_hborder:nnn {
722   \group_begin:
723     \hbox_set_to_wd:Nnn \l_tmpa_box {1pt} {
724       \hss
725       \hbox:n {
726         #3 % install color
727         \vrule height~\dim_eval:n{#1+#2}
728           ~depth~\dim_eval:n{#2}
729           ~width~3pt
730       }
731       \hss
732     }
733     \box_set_ht:Nn \l_tmpa_box { \c_zero_dim }
734     \box_set_dp:Nn \l_tmpa_box { \c_zero_dim }
735     \kern 1pt
736     \box_use:N \l_tmpa_box
737     \xleaders
738       \box_use:N \l_tmpa_box
739       \skip_horizontal:n {-4pt+plus~1fil}
740     \box_use:N \l_tmpa_box
741     \kern 1pt
742     \skip_horizontal:n {0pt+plus~1fil}
743   \group_end:
744 }
745 \cs_new_protected_nopar:Nn \_cellprops_make_solid_vborder:nnn {
746   \group_begin:
747     \hbox_set_to_wd:Nnn \l_tmpa_box {0pt} {
748       \hbox:n {
749         #3 % install color
750         \vrule height~\dim_eval:n{#2}~width~\dim_eval:n{#1}
751       }
752       \hss
753     }
754     \box_set_ht:Nn \l_tmpa_box { \c_zero_dim }
755     \box_set_dp:Nn \l_tmpa_box { \c_zero_dim }
756     \box_use:N \l_tmpa_box
757   \group_end:
758 }
759 \clist_map_inline:nn {
760   dotted, dashed, solid, double,

```

```

761     groove, ridge, inset, outset
762 }{
763     \cs_set_eq:cN {__cellprops_make_#1_hborder:nnn} \__cellprops_make_solid_hborder:nnn
764     \cs_set_eq:cN {__cellprops_make_#1_vborder:nnn} \__cellprops_make_solid_vborder:nnn
765 }
766
767 \dim_new:N \l__cellprops_border_width_dim
768 \str_new:N \l__cellprops_border_style_str
769 \tl_new:N \l__cellprops_border_color_tl
770 \cs_new_protected_nopar:Nn \__cellprops_get_border_info:n {
771     \dim_set:Nn \l__cellprops_border_width_dim {\__cellprops_get_property:n {border-#1-width}}
772     \__cellprops_get_property:nN {border-#1-style} \l_tmpa_tl
773     \exp_args:NNV \str_set:Nn \l__cellprops_border_style_str \l_tmpa_tl
774     \tl_clear:N \l__cellprops_border_color_tl
775     \cs_if_exist:cTF {__cellprops_make_#1__cellprops_border_style_str_hborder:nnn} {
776         \__cellprops_update_color:Nn \l__cellprops_border_color_tl {border-#1-color}
777     }{
778         \dim_zero:N \l__cellprops_border_width_dim
779     }
780 }
781
782 \cs_new_protected_nopar:Npn \__cellprops_make_hborder:nnnn #1 {
783     \use:c { __cellprops_make_#1_hborder:nnn }
784 }
785 \cs_new_protected_nopar:Npn \__cellprops_make_vborder:nnnn #1 {
786     \use:c { __cellprops_make_#1_vborder:nnn }
787 }
788
789 \cs_new_protected_nopar:Nn \__cellprops_end_raw_cell: {
790     % Here \l__cellprops_cell_box must contain the contents of the cell
791     %
792     % Prepare the borders token list
793     \int_compare:nNnT \g__cellprops_col_int = 1 {
794         \tl_gclear:N \g__cellprops_borders_tl
795     }
796     \tl_gput_right:Nx \g__cellprops_borders_tl {
797         \tl_if_empty:NF \g__cellprops_borders_tl { \exp_not:n {&} }
798         \exp_not:n { \omit \kern \c_zero_dim }
799     }
800     % Handle padding-top, min-height and border-top
801     \__cellprops_get_border_info:n {top}
802     \box_set_ht:Nn \l__cellprops_cell_box {
803         \dim_max:nn
804         {\box_ht:N \l__cellprops_cell_box}
805         {\__cellprops_get_property:n {min-height}}
806         + (\__cellprops_get_property:n {padding-top})
807         + \l__cellprops_border_width_dim
808     }
809     \dim_compare:nNnT \l__cellprops_border_width_dim > \c_zero_dim {
810         \tl_gput_right:Nx \g__cellprops_borders_tl {
811             \exp_not:N \__cellprops_make_hborder:nnnn
812             { \exp_not:V \l__cellprops_border_style_str }
813             { \dim_use:N \l__cellprops_border_width_dim }
814             {

```

```

815         \exp_not:n { \g__cellprops_dp_dim + \g__cellprops_ht_dim - }
816         \dim_use:N \l__cellprops_border_width_dim
817     }
818     { \exp_not:V \l__cellprops_border_color_tl }
819 }
820 }
821 % Handle padding-bottom, min-depth and border-bottom
822 \__cellprops_get_border_info:n {bottom}
823 \box_set_dp:Nn \l__cellprops_cell_box {
824     \dim_max:nn
825     { \box_dp:N \l__cellprops_cell_box }
826     { \__cellprops_get_property:n {min-depth} }
827     + ( \__cellprops_get_property:n {padding-bottom} )
828     + \l__cellprops_border_width_dim
829 }
830 \dim_compare:nNnT \l__cellprops_border_width_dim > \c_zero_dim {
831     \tl_gput_right:Nx \g__cellprops_borders_tl {
832         \exp_not:N \__cellprops_make_hborder:nnnn
833         { \exp_not:V \l__cellprops_border_style_str }
834         { \dim_use:N \l__cellprops_border_width_dim }
835         { \exp_not:n { Opt } }
836         { \exp_not:V \l__cellprops_border_color_tl }
837     }
838 }
839 % To fix vertical alignment later
840 \dim_gset:Nn \g__cellprops_ht_dim {
841     \dim_max:nn
842     { \g__cellprops_ht_dim }
843     { \box_ht:N \l__cellprops_cell_box }
844 }
845 \dim_gset:Nn \g__cellprops_dp_dim {
846     \dim_max:nn
847     { \g__cellprops_dp_dim }
848     { \box_dp:N \l__cellprops_cell_box }
849 }
850 % Handle hpadding and halign
851 \skip_set:Nn \l_tmpa_skip {
852     \dim_max:nn
853     { Opt }
854     { ( \__cellprops_get_property:n {min-width} )
855       - \box_wd:N \l__cellprops_cell_box }
856 }
857 \skip_add:Nn \l_tmpa_skip {
858     Opt plus 1fil
859 }
860 % padding-left and border-left
861 \__cellprops_get_border_info:n {left}
862 \skip_set:Nn \l__cellprops_left_skip
863     { \__cellprops_get_property:n {padding-left} + \l__cellprops_border_width_dim }
864 \dim_compare:nNnT \l__cellprops_border_width_dim > \c_zero_dim {
865     \tl_gput_right:Nx \g__cellprops_borders_tl {
866         \exp_not:N \__cellprops_make_vborder:nnnn
867         { \exp_not:V \l__cellprops_border_style_str }
868         { \dim_use:N \l__cellprops_border_width_dim }

```

```

869         { \exp_not:n { \g__cellprops_dp_dim + \g__cellprops_ht_dim } }
870         { \exp_not:V \l__cellprops_border_color_tl }
871     }
872 }
873 \tl_gput_right:Nx \g__cellprops_borders_tl {
874     \exp_not:n {
875         \skip_horizontal:n {0pt~plus~1fil}
876         \kern \c_zero_dim
877     }
878 }
879 \__cellprops_get_border_info:n {right}
880 \skip_set:Nn \l__cellprops_right_skip
881     { \__cellprops_get_property:n {padding-right} + \l__cellprops_border_width_dim }
882 \dim_compare:nNnT \l__cellprops_border_width_dim > \c_zero_dim {
883     \tl_gput_right:Nx \g__cellprops_borders_tl {
884         \exp_not:N \skip_horizontal:n
885             { - \dim_use:N \l__cellprops_border_width_dim }
886         \exp_not:N \__cellprops_make_vborder:nnnn
887             { \exp_not:V \l__cellprops_border_style_str }
888             { \dim_use:N \l__cellprops_border_width_dim }
889             { \exp_not:n { \g__cellprops_dp_dim + \g__cellprops_ht_dim } }
890             { \exp_not:V \l__cellprops_border_color_tl }
891         \exp_not:N \skip_horizontal:n
892             { \dim_use:N \l__cellprops_border_width_dim }
893         \exp_not:n { \kern \c_zero_dim }
894     }
895 }
896 \str_case:x:nnF { \__cellprops_get_property:n {text-align} } {
897     { right } {
898         \skip_add:Nn \l__cellprops_left_skip { \l_tmpa_skip }
899     }
900     { center } {
901         \skip_add:Nn \l__cellprops_left_skip { \l_tmpa_skip / 2 }
902         \skip_add:Nn \l__cellprops_right_skip { \l_tmpa_skip / 2 }
903     }
904 } {% any other treated as |left|
905     \skip_add:Nn \l__cellprops_right_skip { \l_tmpa_skip }
906 }
907 \kern \c_zero_dim
908 \tl_if_empty:NF \l__cellprops_bgcolor_tl {
909     \group_begin:
910     % Paint a background with leaders
911     \tl_use:N \l__cellprops_bgcolor_tl % install the color
912     \skip_set:Nn \l_tmpa_skip {
913         \l__cellprops_left_skip
914         + \box_wd:N \l__cellprops_cell_box
915         + \l__cellprops_right_skip
916     }
917     \leaders
918         \vrule
919         \skip_horizontal:N \l_tmpa_skip
920     \skip_horizontal:n {-\l_tmpa_skip}
921     \group_end:
922 }

```

```

923     \skip_horizontal:N \l__cellprops_left_skip
924     \box_use_clear:N \l__cellprops_cell_box
925     \skip_horizontal:N \l__cellprops_right_skip
926     \kern\c_zero_dim
927     \group_end:
928 }
929 \end{package}

```