

Listing 1: Transcript of Usability Study

```
1  R: (Researcher)
2  P: (Participant)
3
4  ===Flyouts On===
5
6      First system, Flyouts ON.
7
8  R:  Start the system by pressing Apple K.
9  P:      Okay, is there an Apple on here?
10 R:  Yeah, I've mapped Ctrl to Apple.
11 P:      Okay, great.
12 P:      I'll do what I have to do to get started, and...
13          *Presses Apple K*
14          *Chibipoint interface appears*
15 P:      Okay.
```

0.1 ./README

2

```
1  =Rights:
2  ChibiPoint is made by Alex Birch.
3  Source code license is GNU GPL v3.Full ↗
   ↗ text available in file`LICENSE`.
4
5
6  =Installation:
7  Using Google Chrome,
8  1. Go to chrome://extensions/
9  2. Ensure `Developer mode` at the top is ↗
   ↗ checked.
10 3. `Load unpacked extension`
11 4. Choose this folder, `chibipoint`.
12 5. ChibiPoint should appear in yourlist ↗
   ↗ of Extensions, and be `Enabled`.
13 6. If you wish, you can customise the↗
   ↗ invocation shortcut using the`↗
   ↗ Keyboard Shortcuts` link at the↗
   ↗ bottom of chrome://extensions/
14
15
16 =Invocation:
17 Browse the web as usual.
18 Press Cmd+K (Mac) / Ctrl+K (Windows)to ↗
   ↗ invoke ChibiPoint interface.
19
20 You should see a grid. If you don t , ↗
```

```
21
22
23 =Usage:
24 Press the shortcuts you see on thegrid ↗
   ↗ to `drill down in thatdirection.
25 Labels fly out from suggestedclickables.↗
   ↗ These `flyouts` can beinvoked by↗
   ↗ pressing theircorresponding key.
26
27 By default, you drill with the numpad↗
   ↗ directions.
28 Flyout shortcuts are mapped to a 3x3grid↗
   ↗ on the left hand of a Qwerty↗
   ↗ keyboard.
29
30 Caps lock is not supported.
31
32
33 =Customizing keys:
34 By default we use numpad to `drill↗
   ↗ down . If you don t have a↗
   ↗ numpad, you should probably change↗
   ↗ the code a little.
35
36 Lines 80-154 concern keybindings.Sorry ↗
   ↗ the world isn t nicer.
37 There are a few defaults you mightwant ↗
   ↗ to try out, so fiddle aroundwith ↗
```

↗ that s bad. Probably an↗
↗ installation or keybinding problem.

```

38     ↪ numpad = true and Dvorak =false.
39 Just edit the code, and refresh anypage ↪
    ↪ that was using ChibiPoint; itshould↪
    ↪ update immediately.

```

0.2 ./LICENSE

```

1 The source code to ChibiPoint isreleased↪
    ↪ under the GNU GPL v3license, the ↪
    ↪ text of which
2 is available at:
3 http://www.gnu.org/licenses/gpl.html

```

0.3 ./manifest.json

```

1 {
2   "name": "ChibiPoint",
3   "description": "Point at webpagesusing↪
    ↪ the keyboard",
4   "version": "0.5",
5   "content_scripts": [
6     {
7       "matches": ["↪
    ↪ http://*/*"↪
    ↪ , "https:↪
    ↪ /*/*"],
8   "css": ["styles.css"],

```

```

    "js": ["src/lib/require.js","src/↪
    ↪ lib/require-cs.js","src/↪
    ↪ depend.js"],
    "run_at": "document_start"
  }
],
"permissions": [
  "tabs",
  "activeTab"
],
"background": {"scripts":["src/↪
    ↪ background.js"]},
"commands": {
  "toggle-feature-foo": {
    "suggested_key": {
      "default": "Ctrl+K",
      "mac": "Command+K"
    },
    "description": "Invoke ChibiPoint"
  }
},
"manifest_version": 2,
"web_accessible_resources": [
  "*.js"
]
}

```

0.4 ./styles.less

```

1 // out: styles.css

```

4

```

2  // compress: false, strictMath: false
3
4
5  @gridcol: #ccc;
6  @gridthick: 1px;
7
8  @growcoeff: 1.05;
9
10
11  body {
12      padding: 0;
13  }
14
15  .chibiPoint_gridContainer {
16      padding: 0;
17      //top: 8.3%;
18      top: 0;
19      left: calc(((100%*(@growcoeff - 1))/2) / ↗
                ↗ @growcoeff);
20
21      position: fixed;
22      width: calc(100%/@growcoeff);
23      height: calc(100%/@growcoeff);
24
25      pointer-events: none;
26
27      z-index: 2147483646;
28  }
29
30  .chibiPoint_hiddenGridContainer {

```

```

31  display: none;
32  }
33
34  .chibiPoint_grid {
35
36      position: relative;
37      left: calc(-(100%*(@growcoeff - 1))/2);
38
39      //top: -10%;
40      top: 0;
41      //float: left;
42
43      width: calc(100%*@growcoeff);
44      height: calc(100%*@growcoeff);
45
46      //overflow: auto;
47      display: table;
48      table-layout: fixed;
49
50      pointer-events: inherit;
51  }
52
53  .chibiPoint_gridBorderOn {
54      border: @gridthick solid @gridcol;
55      border-width: @gridthick @gridthick ↗
                ↗ @gridthick @gridthick;
56      //outline: 1px solid 0;
57  }
58
59  .chibiPoint_gridBorderOff {

```

5

```

60 }
61
62 .chibiPoint_row {
63     //border: 1px solid #ccc;
64     //border-width: 0 1px 1px 0;
65
66     display: table-row;
67     pointer-events: inherit;
68 }
69
70
71 @cellratio:calc(100%/3);
72
73 .chibiPoint_cell {
74     padding:0;
75
76     width:@cellratio;
77     height:@cellratio;
78     display: table-cell;
79
80
81     // text
82     line-height: @cellratio;
83     text-align:center;
84     vertical-align:middle;
85     color: #000;
86     color:rgba(0, 0, 0, 0.5);
87     font-family:Geneva, Arial,Helvetica, ↗
88         ↗ sans-serif;

```

```

89     pointer-events: inherit;
90
91     position:relative;
92     overflow:visible;
93     clear:both;
94 }
95
96 .chibiPoint_cellBorderOn {
97     /*border: @gridthick solid @gridcol;
98     border-width: @gridthick @gridthick ↗
99         ↗ @gridthick @gridthick;
100     //border-style:double;
101     outline: 1px solid 0;
102     outline-offset: 1px;*/
103
104     //this one!
105     box-shadow: 0 0 0 1px #fff, 0 0 02px ↗
106         ↗ #888;
107 }
108
109 .chibiPoint_cellBorderOff {
110 }
111
112 @backingprop: 0.8;
113 @radius: 15%;
114 @backinggrey: 230;
115
116 .chibiPoint_backing {
117     position:absolute;

```

9

117	top:calc(100%*(1-@backingprop)/2);	142	width:100%;
118	left:calc(100%*(1-@backingprop)/2);	143	height:100%;
119		144	
120	width:calc(100%*@backingprop);	145	position:fixed;
121	height:calc(100%*@backingprop);	146	padding:0;
122	background:rgba(@backinggrey, ↵	147	
	↵ @backinggrey, @backinggrey, 0.25) ↵	148	pointer-events: none;
	↵ ;	149	}
123		150	
124	z-index:-1;	151	.chibiPoint_shownCrosshairs {
125		152	.chibiPoint_fullContainer;
126	-moz-border-radius: @radius @radius;	153	
127	-webkit-border-radius: @radius@radius;	154	z-index:2147483645;
128	border-radius: @radius @radius;	155	}
129		156	
130	//background-blend-mode: difference, ↵	157	.chibiPoint_hiddenCrosshairs {
	↵ normal; // Chrome Canary	158	pointer-events: none;
131	//-webkit-background-blend-mode: ↵	159	
	↵ difference, normal; // WebKit ↵	160	display:none;
	↵ Nightly	161	}
132	}	162	
133		163	.chibiPoint_hairBorder {
134	@haircol:rgba(255, 0, 0, 1);	164	//border-width: @hairborderthick ↵
135	@hairthick:1px;		↵ @hairborderthick @hairborderthick ↵
136	@hairborderthick:1px;		↵ @hairborderthick;
137		165	//border-color:#00ffff;
138	.chibiPoint_fullContainer {	166	
139	top:0;	167	//outline-color;;
140	left:0;	168	//outline: 1px solid 0;
141		169	}

170		200	
171	.chibiPoint_hairlinecommon {	201	.chibiPoint_targeted {
172	padding:0;	202	background: yellow!important;
173		203	//padding: 3px 5px;
174	position:relative;	204	//margin: -3px -5px;
175		205	//line-height: 1.7;
176	pointer-events: inherit;	206	
177		207	border-radius: 3px;
178	background:@haircol;	208	/*border-color: red!important;
179		209	border: 3px solid black;
180	.chibiPoint_hairBorder;	210	box-sizing: border-box;*/
181	}	211	//border-radius: 3px;
182		212	box-shadow: 0 0 10px #ff00ff;
183	.chibiPoint_horizontalHair {	213	
184	.chibiPoint_hairlinecommon;	214	// turn off crosshairs:
185		215	//display: none;
186	left:inherit;	216	
187		217	background-blend-mode: multiply, ↗
188	width:inherit;		↗ normal; // Chrome Canary
189	height:@hairthick;	218	-webkit-background-blend-mode: ↗
190	}		↗ multiply, normal; // WebKit ↗
191			↗ Nightly
192	.chibiPoint_verticalHair {	219	
193	.chibiPoint_hairlinecommon;	220	//display:inline-block;
194		221	}
195	top:@hairthick;	222	
196		223	.chibiPoint_painted {
197	width:@hairthick;	224	background: cyan!important;
198	height:inherit;	225	//padding: 3px 5px;
199	}	226	//margin: -3px -5px;

227	//line-height: 1.7;	251	background: rgba(249,112,12,↵
228			↵ @flyoutAlpha)!important;
229	border-radius: 3px;	252	}
230	/*border-color: red!important;	253	.chibiPoint_paintedUnique4 {
231	border: 3px solid black;	254	background: rgba(186,167,19,↵
232	box-sizing: border-box;*/		↵ @flyoutAlpha)!important;
233	//border-radius: 3px;	255	}
234	box-shadow: 0 0 10px #ff0000;	256	.chibiPoint_paintedUnique5 {
235		257	background: rgba(172,0,237,↵
236	background-blend-mode: multiply,↵		↵ @flyoutAlpha)!important;
	↵ normal; // Chrome Canary	258	}
237	-webkit-background-blend-mode:↵	259	.chibiPoint_paintedUnique6 {
	↵ multiply, normal; // WebKit ↵	260	background: rgba(240,172,24,↵
	↵ Nightly		↵ @flyoutAlpha)!important;
238	//display:inline-block;	261	}
239	}	262	.chibiPoint_paintedUnique7 {
240		263	background: rgba(8,160,94,@flyoutAlpha↵
241	.chibiPoint_paintedUnique0 {		↵)!important;
242	background: rgba(0,173,0,@flyoutAlpha)↵	264	}
	↵ !important;	265	.chibiPoint_paintedUnique8 {
243	}	266	background: rgba(135,54,94,↵
244	.chibiPoint_paintedUnique1 {		↵ @flyoutAlpha)!important;
245	background: rgba(13,255,150,↵	267	}
	↵ @flyoutAlpha)!important;	268	
246	}	269	.chibiPoint_cluck {
247	.chibiPoint_paintedUnique2 {	270	background: red!important;
248	background: rgba(249,25,137,↵	271	//padding: 3px 5px;
	↵ @flyoutAlpha)!important;	272	//margin: -3px -5px;
249	}	273	//line-height: 1.7;
250	.chibiPoint_paintedUnique3 {	274	


```

275     /*border-radius: 3px;
276     border-color: red!important;
277     border: 3px solid black;*/
278
279     //display:inline-block;
280 }
281
282 .chibiPoint_flyoutContainer {
283     .chibiPoint_fullContainer;
284
285     z-index:2147483647;
286 }
287
288 .chibiPoint_shownfC {
289 }
290
291 .chibiPoint_hiddenfC {
292     display:none;
293 }
294
295 .chibiPoint_aFlyout {
296     position: absolute;
297     width:100%;
298     height:100%;
299 }
300
301 .chibiPoint_aFlyoutHidden {
302     display: none;
303 }
304
305 .chibiPoint_flyout {
306     //background: rgba(0,173,0, 0.30);
307     //outline: 1px solid 0!important;
308     border: 1px solid black;
309     position: absolute;
310     top:0;
311     left: 0;
312     width:20px;
313     text-align: center;
314     font-family:Geneva, Arial,Helvetica, ↵
315         ↵ sans-serif;
316
317     /*-webkit-text-stroke: 1px #000000;
318     -webkit-text-fill-color: #FFFFFF;*/
319     color: white;
320     text-shadow:
321         -1px -1px 0 #000,
322         1px -1px 0 #000,
323         -1px 1px 0 #000,
324         1px 1px 0 #000;
325 }
326 @flyoutAlpha: 1.0;
327
328 .chibiPoint_flyoutUnique0 {
329     background: rgba(0,173,0,@flyoutAlpha) ↵
330         ↵ ;
331 }
332 .chibiPoint_flyoutUnique1 {
333     background: rgba(13,255,150, ↵

```

```

333     ↪ @flyoutAlpha);
334 }
335 .chibiPoint_flyoutUnique2 {
336     background: rgba(249,25,137, ↪
337         ↪ @flyoutAlpha);
338 }
339 .chibiPoint_flyoutUnique3 {
340     background: rgba(249,112,12, ↪
341         ↪ @flyoutAlpha);
342 }
343 .chibiPoint_flyoutUnique4 {
344     background: rgba(186,167,19, ↪
345         ↪ @flyoutAlpha);
346 }
347 .chibiPoint_flyoutUnique5 {
348     background: rgba(172,0,237, ↪
349         ↪ @flyoutAlpha);
350 }
351 .chibiPoint_flyoutUnique6 {
352     background: rgba(240,172,24, ↪
353         ↪ @flyoutAlpha);
354 }
355     ↪ @flyoutAlpha);
356 }
357 .chibiPoint_lineUnique0 {
358     stroke: rgb(0,173,0,@flyoutLineAlpha);
359 }
360 .chibiPoint_lineUnique1 {
361     stroke: rgb(13,255,150, ↪
362         ↪ @flyoutLineAlpha);
363 }
364 .chibiPoint_lineUnique2 {
365     stroke: rgb(249,25,137, ↪
366         ↪ @flyoutLineAlpha);
367 }
368 .chibiPoint_lineUnique3 {
369     stroke: rgb(249,112,12, ↪
370         ↪ @flyoutLineAlpha);
371 }
372 .chibiPoint_lineUnique4 {
373     stroke: rgb(186,167,19, ↪
374         ↪ @flyoutLineAlpha);
375 }
376 .chibiPoint_lineUnique5 {
377     stroke: rgb(172,0,237,@flyoutLineAlpha ↪
378         ↪ );
379 }
380 .chibiPoint_lineUnique6 {
381     stroke: rgb(240,172,24, ↪
382         ↪ @flyoutLineAlpha);
383 }
384 .chibiPoint_lineUnique7 {
385     stroke: rgb(8,160,94,@flyoutLineAlpha) ↪

```

```

379     ↪ ;
380 }
381 .chibiPoint_lineUnique8 {
382     stroke: rgb(135,54,94,@flyoutLineAlpha ↪
383     ↪ );
384 }
385 .chibiPoint_lineDiv {
386 }
387 .chibiPoint_lineSvg {
388 }
390 @flyoutLineAlpha:1;
391 @flyoutLineWidth:2;
392 .chibiPoint_line {
393     //stroke:rgb(0,173,0);
394     stroke-width:@flyoutLineWidth;
395 }
396
397 .chibiPoint_evaluator {
398     position:fixed;
399
400     pointer-events: none;
401
402     font-family:Geneva, Arial,Helvetica, ↪
403     ↪ sans-serif;
404
405     color:red;

```

```

406
407     width:100%;
408     height:100%;
409
410     display:none;
411
412 }
413
414 .chibiPoint_timer {
415     .chibiPoint_evaluator;
416
417     left:10;
418     top:10;
419 }
420
421 .chibiPoint_keyCount {
422     .chibiPoint_evaluator;
423
424     left:10;
425     top:30;
426 }

```

0.5 ./styles.css

```

1  /* Generated by less 1.5.1 */
2  body {
3      padding: 0;
4  }
5  .chibiPoint_gridContainer {
6      padding: 0;

```

```

7   top: 0;
8   left: calc(2.380952380952383%);
9   position: fixed;
10  width: calc(95.23809523809524%);
11  height: calc(95.23809523809524%);
12  pointer-events: none;
13  z-index: 2147483646;
14  }
15  .chibiPoint_hiddenGridContainer {
16    display: none;
17  }
18  .chibiPoint_grid {
19    position: relative;
20    left: calc(-2.5000000000000002%);
21    top: 0;
22    width: calc(105%);
23    height: calc(105%);
24    display: table;
25    table-layout: fixed;
26    pointer-events: inherit;
27  }
28  .chibiPoint_gridBorderOn {
29    border: 1px solid #cccccc;
30    border-width: 1px 1px 1px 1px;
31  }
32  .chibiPoint_row {
33    display: table-row;
34    pointer-events: inherit;
35  }
36  .chibiPoint_cell {

```

```

37  padding: 0;
38  width: calc(33.333333333333336%);
39  height: calc(33.333333333333336%);
40  display: table-cell;
41  line-height: calc(33.333333333333336%);
42  text-align: center;
43  vertical-align: middle;
44  color: #000;
45  color: rgba(0, 0, 0, 0.5);
46  font-family: Geneva, Arial, Helvetica, ↵
    ↵ sans-serif;
47  pointer-events: inherit;
48  position: relative;
49  overflow: visible;
50  clear: both;
51  }
52  .chibiPoint_cellBorderOn {
53    /*border: @gridthick solid @gridcol;
54    border-width: @gridthick @gridthick ↵
    ↵ @gridthick @gridthick;
55    //border-style:double;
56    outline: 1px solid 0;
57    outline-offset: 1px;*/
58    box-shadow: 0 0 0 1px #fff, 0 0 0 2px ↵
    ↵ #888;
59  }
60  .chibiPoint_backing {
61    position: absolute;
62    top: calc(9.999999999999998%);
63    left: calc(9.999999999999998%);

```

```

64 width: calc(80%);
65 height: calc(80%);
66 background: rgba(230, 230, 230, 0.25);
67 z-index: -1;
68 -moz-border-radius: 15% 15%;
69 -webkit-border-radius: 15% 15%;
70 border-radius: 15% 15%;
71 }
72 .chibiPoint_fullContainer {
73 top: 0;
74 left: 0;
75 width: 100%;
76 height: 100%;
77 position: fixed;
78 padding: 0;
79 pointer-events: none;
80 }
81 .chibiPoint_shownCrosshairs {
82 top: 0;
83 left: 0;
84 width: 100%;
85 height: 100%;
86 position: fixed;
87 padding: 0;
88 pointer-events: none;
89 z-index: 2147483645;
90 }
91 .chibiPoint_hiddenCrosshairs {
92 pointer-events: none;
93 display: none;
94 }
95 .chibiPoint_hairlinecommon {
96 padding: 0;
97 position: relative;
98 pointer-events: inherit;
99 background: #ff0000;
100 }
101 .chibiPoint_horizontalHair {
102 padding: 0;
103 position: relative;
104 pointer-events: inherit;
105 background: #ff0000;
106 left: inherit;
107 width: inherit;
108 height: 1px;
109 }
110 .chibiPoint_verticalHair {
111 padding: 0;
112 position: relative;
113 pointer-events: inherit;
114 background: #ff0000;
115 top: -1px;
116 width: 1px;
117 height: inherit;
118 }
119 .chibiPoint_targeted {
120 background: yellow!important;
121 border-radius: 3px;
122 /*border-color: red!important;
123 border: 3px solid black;

```

```

124     box-sizing: border-box;*/
125     box-shadow: 0 0 10px #ff00ff;
126     background-blend-mode: multiply, ↵
        ↵ normal;
127     -webkit-background-blend-mode: multiply ↵
        ↵ , normal;
128 }
129 .chibiPoint_painted {
130     background: cyan!important;
131     border-radius: 3px;
132     /*border-color: red!important;
133     border: 3px solid black;
134     box-sizing: border-box;*/
135     box-shadow: 0 0 10px #ff0000;
136     background-blend-mode: multiply, ↵
        ↵ normal;
137     -webkit-background-blend-mode: multiply ↵
        ↵ , normal;
138 }
139 .chibiPoint_paintedUnique0 {
140     background: #00ad00 !important;
141 }
142 .chibiPoint_paintedUnique1 {
143     background: #0dff96 !important;
144 }
145 .chibiPoint_paintedUnique2 {
146     background: #f91989 !important;
147 }
148 .chibiPoint_paintedUnique3 {
149     background: #f9700c !important;
150 }
151 .chibiPoint_paintedUnique4 {
152     background: #baa713 !important;
153 }
154 .chibiPoint_paintedUnique5 {
155     background: #ac00ed !important;
156 }
157 .chibiPoint_paintedUnique6 {
158     background: #f0ac18 !important;
159 }
160 .chibiPoint_paintedUnique7 {
161     background: #08a05e !important;
162 }
163 .chibiPoint_paintedUnique8 {
164     background: #87365e !important;
165 }
166 .chibiPoint_cluck {
167     background: red!important;
168     /*border-radius: 3px;
169     border-color: red!important;
170     border: 3px solid black;*/
171 }
172 .chibiPoint_flyoutContainer {
173     top: 0;
174     left: 0;
175     width: 100%;
176     height: 100%;
177     position: fixed;
178     padding: 0;
179     pointer-events: none;

```

180	z-index: 2147483647;	207	background: #00ad00;
181	}	208	}
182	.chibiPoint_hiddenfC {	209	.chibiPoint_flyoutUnique1 {
183	display: none;	210	background: #0dff96;
184	}	211	}
185	.chibiPoint_aFlyout {	212	.chibiPoint_flyoutUnique2 {
186	position: absolute;	213	background: #f91989;
187	width: 100%;	214	}
188	height: 100%;	215	.chibiPoint_flyoutUnique3 {
189	}	216	background: #f9700c;
190	.chibiPoint_aFlyoutHidden {	217	}
191	display: none;	218	.chibiPoint_flyoutUnique4 {
192	}	219	background: #baa713;
193	.chibiPoint_flyout {	220	}
194	border: 1px solid black;	221	.chibiPoint_flyoutUnique5 {
195	position: absolute;	222	background: #ac00ed;
196	top: 0;	223	}
197	left: 0;	224	.chibiPoint_flyoutUnique6 {
198	width: 20px;	225	background: #f0ac18;
199	text-align: center;	226	}
200	font-family: Geneva, Arial, Helvetica, ↗ ↗ sans-serif;	227	.chibiPoint_flyoutUnique7 {
201	/*-webkit-text-stroke: 1px #000000;	228	background: #08a05e;
202	-webkit-text-fill-color: #FFFFFF;*/	229	}
203	color: white;	230	.chibiPoint_flyoutUnique8 {
204	text-shadow: -1px -1px 0 #000000, 1px ↗ ↗ -1px 0 #000000, -1px 1px 0 ↗ ↗ #000000, 1px 1px 0 #000000;	231	background: #87365e;
205	}	232	}
206	.chibiPoint_flyoutUnique0 {	233	.chibiPoint_lineUnique0 {
		234	stroke: #00ad00;
		235	}
		236	.chibiPoint_lineUnique1 {

237	stroke: #0dff96;	267	↪ sans-serif;
238	}	268	color: red;
239	.chibiPoint_lineUnique2 {	269	width: 100%;
240	stroke: #f91989;	270	height: 100%;
241	}	271	display: none;
242	.chibiPoint_lineUnique3 {	272	}
243	stroke: #f9700c;	273	.chibiPoint_timer {
244	}	274	position: fixed;
245	.chibiPoint_lineUnique4 {	275	pointer-events: none;
246	stroke: #baa713;	276	font-family: Geneva, Arial, Helvetica, ↪
247	}	277	↪ sans-serif;
248	.chibiPoint_lineUnique5 {	278	color: red;
249	stroke: #ac00ed;	279	width: 100%;
250	}	280	height: 100%;
251	.chibiPoint_lineUnique6 {	281	display: none;
252	stroke: #f0ac18;	282	left: 10;
253	}	283	top: 10;
254	.chibiPoint_lineUnique7 {	284	}
255	stroke: #08a05e;	285	.chibiPoint_keyCount {
256	}	286	position: fixed;
257	.chibiPoint_lineUnique8 {	287	pointer-events: none;
258	stroke: #87365e;	288	font-family: Geneva, Arial, Helvetica, ↪
259	}	289	↪ sans-serif;
260	.chibiPoint_line {	290	color: red;
261	stroke-width: 2;	291	width: 100%;
262	}	292	height: 100%;
263	.chibiPoint_evaluator {	293	display: none;
264	position: fixed;		left: 10;
265	pointer-events: none;		top: 30;
266	font-family: Geneva, Arial, Helvetica, ↪		}

0.6 ./src/background.js

```
1 console.log('alive');
2 chrome.commands.onCommand.addListener(↵
  ↵ function(command) {
3   console.log('Command:', command);
4   chrome.tabs.query({active: true,↵
     ↵ currentWindow: true}, function(↵
       ↵ tabs){
5     chrome.tabs.sendMessage(tabs[0].id,{↵
       ↵ greeting: "hello"},function(↵
         ↵ response) {
6       console.log(response.farewell);
7     });
8   });
9   });
```

0.7 ./src/depend.js

```
1 /// Cheekily change DOM prototype before↵
  ↵ page loads
2 // http://stackoverflow.com/questions↵
  ↵ /8915461/↵
  ↵ how-to-find-out-what-event-↵
  ↵ listener-types-attached-to-specific-↵
  ↵ html-element-in-c
3 // from spektom, thanks to kdzwinel
4
5 var injectedJS = "\
```

```
6 (function(original) { \
7   Element.prototype.addEventListener= ↵
     ↵ function(type, listener,↵
       ↵ useCapture) { \
8     var attr =this.getAttribute('↵
       ↵ _handlerTypes'); \
9     var types = attr ?attr.split(',') : ↵
       ↵ []; \
10    var found = false; \
11    for (var i = 0; i < types.length;++i↵
       ↵ ) { \
12      if (types[i] == type) { \
13        found = true; \
14        break; \
15      } \
16    } \
17    if (!found) { \
18      types.push(type); \
19    } \
20    this.setAttribute('_handlerTypes',↵
       ↵ types.join(',')); \
21    return original.apply(this,arguments↵
       ↵ ); \
22  } \
23 })(Element.prototype.addEventListener); ↵
  ↵ \
24 \
25 (function(original) { \
26   Element.prototype.removeEventListener= ↵
     ↵ function(type, listener,useCapture)↵
```

```

27     { \
28       var attr =this.getAttribute('↵
29       ↵ _handlerTypes'); \
30       var types = attr ?attr.split(',') : ↵
31       ↵ []; \
32       var removed = false; \
33       for (var i = 0; i < types.length;++i↵
34       ↵ ) { \
35         if (types[i] == type) { \
36           types.splice(i, 1); \
37           removed = true; \
38           break; \
39         } \
40         if (removed) { \
41           this.setAttribute('_handlerTypes',↵
42           ↵ types.join(',')'); \
43         } \
44         return original.apply(this, arguments↵
45         ↵ ); \
46       } \
47     })(Element.prototype.removeEventListener↵
48     ↵ ); \
49 ";
50
51 var script =document.createElement("↵
52 ↵ script");
53 script.type = "text/javascript";
54 script.appendChild(document.createTex↵
55 ↵ tNode(injectedJS));

```

```

48 document.documentElement.appendChild(↵
49 ↵ script);
50
51 ///////
52 url = "/src/";
53 if (chrome.extension != undefined) {
54   url =chrome.extension.getURL("/src/");
55 }
56
57 require({ baseUrl:url },["lib/↵
58 ↵ jquery-2.1.0.min","lib/within", "↵
59 ↵ lib/Blob","lib/FileSaver", "↵
60 ↵ evaluator", "test","trace", "lookup↵
61 ↵ ", "testonly","setup", "Grid", "↵
62 ↵ Crosshairs","Flyout"], function ($,↵
63 ↵ within, blob,saver, evaluator, ↵
64 ↵ test, trace,lookup, testonly, setup↵
65 ↵ , gridclass,crosshairclass, flyout)↵
66 ↵ {
67   // default namespace
68   window.birchlabs = {};
69
70   //This function is called when↵
71   ↵ scripts/helper/util.js is ↵
72   ↵ loaded.
73   //If util.js calls define(), then↵
74   ↵ this function is not fired ↵
75   ↵ until
76   //util's dependencies have loaded, ↵

```

```

64     ↪ and the util argument will hold
65     //the module value for "helper/util".
66     if (typeof(chrome) == "object" && ↪
67         ↪ chrome.extension) {
68         //chrome.extension.sendRequest({' ↪
69         ↪ get_options': true},function( ↪
70         ↪ response) {
71             setup.main();
72         //});
73     } else {
74         setup.main();
75     }
76 });

```

19

0.8 ./src/setup.js

```

1  // almost all of this code is borrowed ↪
2  ↪ from tokland's typeahead-find ↪
3  ↪ project.
4  // https://code.google.com/p/chrome- ↪
5  ↪ type-ahead/
6  // GNU GPL v3
7  // alert("sup");
8  // Called when the user clicks on the ↪
9  ↪ browser action.
10 // chrome.browserAction.onClicked.addL ↪
11 ↪ istener(function(tab) {

```

```

9  //chrome.tabs.onUpdated.addListener(f ↪
10 ↪ unction(tabId, changeInfo, tab) {
11     // No tabs or host permissions needed!
12     //console.log('Turning ' + tab.url+ ' ↪
13     ↪ red!');
14     //chrome.tabs.executeScript({
15     //code: 'document.body.style.back ↪
16     ↪ groundColor="red"'
17     //});
18 //});
19 define(["test"], function (test) {
20     window.birchlabs = window.birchlabs || {};
21     var birchlabs = window.birchlabs;
22     // to begin with, not in test mode
23     birchlabs.testmode = false;
24
25     if (chrome.runtime) {
26
27         chrome.runtime.onMessage.addListener(
28             function(request, sender, sendResponse) ↪
29             ↪ {
30                 /*console.log(sender.tab ?
31                     "from a content script:" ↪
32                     ↪ + sender.tab.url :
33                     "from the extension");*/
34                 if (request.greeting == "hello") {
35                     //sendResponse({farewell: "goodbye" ↪

```

```

34         ↪ "});
35     test.shortcut();
36 }
37 });
38
39 /* Styles and addStyle borrowed from ↪
39     ↪ nice-alert.js project */
40
41 var styles = '\
42 #type-ahead-box {\
43     position: fixed;\
44     top: 0;\
45     right: 0;\
46     margin: 0;\
47     text-align: left;\
48     z-index: 2147483647;\
49     color: #000;\
50     border-bottom: 1px solid #ccc;\
51     border-bottom: 1px solid rgba ↪
51         ↪ (0,0,0,0.3);\
52     padding: 4px 8px;\
53     opacity: 0.9;\
54     float: right;\
55     clear: both;\
56     overflow: hidden;\
57     font-size: 18px;\
58     font-family: Arial, Verdana, Georgia, ↪
58         ↪ Serif;\
59     white-space: pre-wrap;\

```

```

60     min-width: 60px;\
61     outline: 0;\
62     -webkit-box-shadow: 0px 2px 8px rgba ↪
62         ↪ (0,0,0,0.2);\
63     -moz-box-shadow: 0px 2px 8px rgba ↪
63         ↪ (0,0,0,0.3);\
64 } \
65 \
66 #type-ahead-box small {\
67     letter-spacing: -0.12em;\
68     color: #444;\
69 }'
70
71 function addStyle(css) {
72     var head = document.getElementsByTagName ↪
72         ↪ gName('head')[0];
73     if (head) {
74         var style = document.createElement(" ↪
74         ↪ style");
75         style.type = "text/css";
76         style.appendChild(document.create ↪
76         ↪ TextNode(css));
77         head.appendChild(style);
78     }
79 }
80
81 function main() {
82
83     // Use setInterval to add events to ↪
83     ↪ document.body as soon as possible

```

```

84   interval_id = setInterval(function() {
85       if (document.body) {
86           clearInterval(interval_id);
87           test.init();
88       }
89   }, 100);
90 }
91
92 // doesn't seem to be needed?
93 // setAlternativeActiveDocument(document) ↵
94 ↵ ;
95 // options = default_options;
96
97 function is_shortcut(ev) {
98     var is_mac = ↵
99         ↵ navigator.appVersion.indexOf(" ↵
100             ↵ Mac") !== -1;
101     var is_windows = navigator.appVer ↵
102         ↵ sion.indexOf("Windows") !== -1;
103     var is_alternative_input = (is_mac && ↵
104         ↵ ev.altKey) || (is_windows && ↵
105         ↵ ev.ctrlKey && ev.altKey);
106     return !is_alternative_input && ( ↵
107         ↵ ev.altKey || ev.metaKey || ↵
108         ↵ ev.ctrlKey);

```

```

106 }
107
108 function stopEvent(ev) {
109     ev.preventDefault();
110     ev.stopPropagation();
111 }
112
113 return { main: main,
114         is_shortcut: is_shortcut};
115 });

```

0.9 ./src/main.js

```

1 // Warning to all: this is NOT the main ↵
2 ↵ file any more. Due to historical ↵
3 ↵ reasons,
4 // test.js became the main entrypoint. ↵
5 ↵ :)
6
7 // much of this code is borrowed from the ↵
8 ↵ typeahead-find project.
9 // https://code.google.com/p/chrome- ↵
10 ↵ type-ahead/
11 // GNU GPL v3
12
13 // alert("sup");
14
15 // Called when the user clicks on the ↵
16 ↵ browser action.

```

```

11 //chrome.browserAction.onClicked.addL↵
    ↪ istener(function(tab) {
12 //chrome.tabs.onUpdated.addListener(f↵
    ↪ unction(tabId, changeInfo, tab) {
13 // No tabs or host permissions needed!
14 //console.log('Turning ' + tab.url+ ' ↵
    ↪ red!');
15 //chrome.tabs.executeScript({
16 //code: 'document.body.style.back↵
    ↪ groundColor="red"
17 //});
18 //});
19
20 /* Styles and addStyle borrowed from↵
    ↪ nice-alert.js project */
21
22 var styles = '\
23 #type-ahead-box {\
24   position: fixed;\
25   top: 0;\
26   right: 0;\
27   margin: 0;\
28   text-align: left;\
29   z-index: 2147483647;\
30   color: #000;\
31   border-bottom: 1px solid #ccc;\
32   border-bottom: 1px solidrgb↵
    ↪ (0,0,0,0.3);\
33   padding: 4px 8px;\
34   opacity: 0.9;\

```

```

35 float: right;\
36 clear: both;\
37 overflow: hidden;\
38 font-size: 18px;\
39 font-family: Arial, Verdana, Georgia,↵
    ↪ Serif;\
40 white-space: pre-wrap;\
41 min-width: 60px;\
42 outline: 0;\
43 -webkit-box-shadow: 0px 2px 8pxrgb↵
    ↪ (0,0,0,0.2);\
44 -moz-box-shadow: 0px 2px 8pxrgb↵
    ↪ (0,0,0,0.3);\
45 }\
46 \
47 #type-ahead-box small {\
48   letter-spacing: -0.12em;\
49   color: #444;\
50 }'
51
52 function addStyle(css) {
53 var head = document.getElementsByTa↵
    ↪ gName('head')[0];
54 if (head) {
55   var style =document.createElement("↵
    ↪ style");
56   style.type = "text/css";
57   style.appendChild(document.create↵
    ↪ TextNode(css));
58   head.appendChild(style);

```

```

59     }
60 }
61
62 function main() {
63
64     // Use setInterval to add events to
65     // document.body as soon as possible
66     interval_id = setInterval(function() {
67         if (document.body) {
68             clearInterval(interval_id);
69             init();
70         }
71     }, 100);
72 }
73 function setAlternativeActiveDocument(doc
74     ) {
75     function dom_trackActiveElement(evt) {
76         if (evt && evt.target) {
77             doc._tafActiveElement = (evt.target
78                 == doc) ? null : evt.target;
79         }
80     }
81     function dom_trackActiveElementLost(evt
82         ) {
83         doc._tafActiveElement = null;
84     }
85     if (doc._tafActiveElement == undefined

```

```

85     doc._tafActiveElement = null;
86     doc.addEventListener("focus",
87         dom_trackActiveElement, true);
88     doc.addEventListener("blur",
89         dom_trackActiveElementLost,
90         true);
91 }
92 // doesn't seem to be needed?
93 // setAlternativeActiveDocument(document)
94 // ;
95 // options = default_options;
96 if (typeof(chrome) == "object" &&
97     chrome.extension) {
98     // chrome.extension.sendRequest({'ge
99         t_options': true}, function(
100             response){
101         main();
102     //});
103 } else {
104     main();
105 }
106 function is_shortcut(ev) {
107     var is_mac =
108         navigator.appVersion.indexOf("

```

```

106     ↪ Mac") !== -1;
107     var is_windows = navigator.appVer↪
108     ↪ sion.indexOf("Windows") !== -1;
109     var is_alternative_input =(is_mac &&↪
110     ↪ ev.altKey) || (is_windows&& ↪
111     ↪ ev.ctrlKey && ev.altKey);
112     return !is_alternative_input &&(↪
113     ↪ ev.altKey || ev.metaKey ||↪
114     ↪ ev.ctrlKey);
115 }
116 function stopEvent(ev) {
117     ev.preventDefault();
118     ev.stopPropagation();
119 }
120 function upMatch(element, matchFunction) ↪
121     ↪ {
122     while (element) {
123         var res = matchFunction(element);
124         if (res == null)
125             return null;
126         else if (res)
127             return element;
128         element = element.parentNode;
129     }
130     return element;
131 }

```

```

129 function getActiveElement(doc) {
130     return doc.activeElement ||↪
131     ↪ doc._tafActiveElement;
132 }
133 function isInputElementActive(doc) {
134     var element = getActiveElement(doc);
135     if (!element)
136         return;
137     var name = element.tagName.toLowerCase↪
138     ↪ ();
139     if (["input", "select", "textarea", "↪
140     ↪ object", "embed"].indexOf(name) ↪
141     ↪ >=0)
142         return true;
143     return (upMatch(element, function(el) {
144         if (!el.getAttribute ||↪
145         ↪ el.getAttribute('↪
146         ↪ contenteditable') == 'false')
147             return null;
148         return el.getAttribute('↪
149         ↪ contenteditable');
150     })))
151 }
152 function getRootNodes() {
153     var rootNodes = new Array();
154     var frames = document.getElementsByTagName↪
155     ↪ TagName('frame');
156     for (var i = 0; i < frames.length; i++)

```



```

151     rootNodes.push(frames[i]);
152     return rootNodes;
153 }
154
155 function init() {
156     var keycodes = {
157         "backspace": 8,
158         "tab": 9,
159         "enter": 13,
160         "spacebar": 32,
161         "escape": 27,
162         "n": 78,
163         "p": 80,
164         "g": 71,
165         "f3": 114,
166         "f4": 115,
167         "dot": 46,
168         "zero": 48,
169         "activate": 167
170     };
171
172     var numpadtype = true;
173     var numpadstart = 48;
174
175     var selectHistory = [];
176     selectHistory.push(".gridContainer");
177
178     function getLatestSelector() {
179         return selectHistory[selectHistory
            ↪ y.length-1];

```

```

180     }
181
182     function getPreviousSelector() {
183         if (selectHistory.length>1) {
184             selectHistory.pop(selectHistory.
            ↪ length-1);
185         }
186         return getLatestSelector();
187     }
188
189     //createGrid();
190     function setEvents(rootNode) {
191         var doc =
            ↪ rootNode.contentDocument
            ↪ || rootNode;
192
193         var body = rootNode.body;
194
195         if (!body || !body.addEventListener)
            ↪ {
196             return;
197         }
198
199         setAlternativeActiveD
            ↪ ocument(document);
200
201         // usually on keydown
202         //processSearch();
203         //draw();
204         var container = makeGridContainer(
            ↪ document.documentElement);

```

26

```

204         // need to make a grid           227
205         //selectHistory =[".",↵         228
206         ↵ gridContainer"];           229
207         //createGrid(container);       230
208
209 doc.addEventListener('keypress',function↵ 231
    ↵ (ev) {
210     if (isInputElementActive(doc)) {    232
211         return;
212     }
213     var code = ev.keyCode;              233
214     var ascii =String.fromCharCode(↵    234
        ↵ code);                          235
215
216     if (!is_shortcut(ev) && ascii&& [↵ 236
        ↵ keycodes.enter].indexOf(code)↵ 237
        ↵ ==-1) {
217         var tracer =doc.getElementById("↵ 238
            ↵ trace");                    239
218         if (tracer != null) {
219 doc.getElementById("trace").innerHTML= ↵ 240
            ↵ code;
220         }                                241
221                                         242
222         if (numpadtype) {                243
223             var num = code-numpadstart;  244
224             if (num<10 && num > 0) {
225                 if (tracer != null) {    245
226 doc.getElementById("trace").innerHTML+= ↵ 246

```

```

    ↵ ", " + (code-numpadstart);
        }
trace($(getLatestSelector()+"grid").get↵
    ↵ (0));

    $(getLatestSelector()+"cell↵
        ↵ ").text("");
    $(getLatestSelector()+"grid↵
        ↵ ").css({"border-color":↵
        ↵ "#C1E0FF",
        "border-width":"0px",
        "border-style":"solid"});
    $(getLatestSelector()+"cell↵
        ↵ ").css({"border-color":↵
        ↵ "#C1E0FF",
        "border-width":"0px",
        "border-style":"solid"});

    var rowIndexed1 =4-Math.ceil↵
        ↵ (num/3);
    var columnIndexed1 =((num-1)↵
        ↵ %3)+1;
    //cell.innerHTML = 7-(i*3-p);

    selectArray = [];
    selectArray.push(getLates↵
        ↵ tSelector());
    selectArray.push(".grid");
    selectArray.push(".row:↵

```

27

```

247     ↪ nth-of-type("+↵
     ↪ rowIndexed1+")"); 265
selectArray.push(".cell:↵ 266
     ↪ nth-of-type("+↵ 267
     ↪ columnIndexed1+")"); 268
248 selectHistory.push(selectArray.join("")) ↵ 269
    ↪ ;

249     createGrid($(getLatestSel↵ 270
     ↪ ector()).first());
251     console.log("not zero"); 271
252 } else if (code ==↵ 272
     ↪ keycodes.zero) { 273
253     console.log("zero"); 274
254     // remove a grid! 275
255     $(getLatestSelector()).empty(); 276
256     getPreviousSelector();
257
258     trace(getLatestSelector()); 277
259
260     var anticipatedHeight =$(↵ 278
     ↪ getLatestSelector()).↵ 279
     ↪ height();
261     var cellHeight =↵ 280
     ↪ anticipatedHeight/3;
262     $(getLatestSelector()+"↵ 281
     ↪ .row").each(function(↵ 282
     ↪ index) {
263         var i = index; 283
264         $(this).children().each(↵ 284

```

```

     ↪ function(index) {
        var p = index;
        if (cellHeight>12) {
            $(this).text(7-(i*3-p));

            var trans =↵
            ↪ document.createElement↵
            ↪ ('div');
            trans.className ="↵
            ↪ backing";

            $(this).append(trans);
        }
    });

    $(getLatestSelector()+"↵ 285
     ↪ ").css({"border-color":↵
     ↪ "#ccc",
     ↪ "border-width":"1px 0 01px"↵
     ↪ ,
     ↪ "border-style":"solid"});
    $(getLatestSelector()+"↵ 286
     ↪ ").css({"border-color":↵
     ↪ "#ccc",
     ↪ "border-width":"0 1px1px 0"↵
     ↪ ,
     ↪ "border-style":"solid"});
    }
    }
    if (code == keycodes.dot) {

```

```

285 // click
286 var $this = $(getLatestSelector ↵
    ↵ ()+".grid").first();
287 var offset = $this.offset(); 306
288 //console.log($this.get(0)); 307
289 //console.log($this.offset()); 308
290 var width = $this.outerWidth(); 309
291 var height = $this.outerHeight ↵ 310
    ↵ (); 311
292
293 var centerX = offset.left + ↵ 312
    ↵ width / 2;
294 var centerY = offset.top + ↵ 313
    ↵ height / 2; 314
295 console.log("x: "+centerX+"y: ↵ 315
    ↵ "+centerY); 316
297 317
298 var rootNodes = [window].concat ↵ 318
    ↵ (getRootNodes());
299 for (var i = 0; i < ↵ 319
    ↵ rootNodes.length; i++) { 320
300     var doc2 = rootNodes[i]. ↵ 321
        ↵ document || rootNodes[i] ↵ 322
        ↵ ].contentDocument; 323
301     if (!doc2 || !doc2.body) 324
302         continue; 325
303 326
304 327
305     var element = $( ↵

```

```

    ↵ doc2.elementFromPoint( ↵
    ↵ centerX, centerY)).get ↵
    ↵ (0);

    console.log(element);

    element.click();
}
} else if (code == ↵
    ↵ keycodes.activate) {
    if ($("#gridContainer").chi ↵
        ↵ ldren().length == 0) {
        // need to make a grid
        selectHistory = [" ↵
            ↵ gridContainer"];
        createGrid(container);
    } else {
        // toggle grid off
        selectHistory = [" ↵
            ↵ gridContainer"];
    }
    $(getLatestSelector()).empty();
}
}
stopEvent(ev);
}, false);
}

// guess this defers setting ↵
    ↵ events until root is loaded ↵

```

```

328         ↪ ?
        var rootNodes =[document].concat ↪
        ↪ (getRootNodes()); 347
329        for (var i = 0; i < ↪ 348
        ↪ rootNodes.length; i++) { 349
330            var rootNode = rootNodes[i];
331            if (rootNode.contentDocument) {
332                rootNode.addEventListener('load', ↪ 350
        ↪ function(ev) { 351
333                setEvents(ev.target.contentDocument);
334                }); 352
335            }
336            else if(!rootNode.contentDocument || ↪ 353
        ↪ rootNode.contentDocument.readyState ↪
        ↪ == 'complete') { 354
337                setEvents(rootNode.contentDocument ? ↪ 355
        ↪ rootNode.contentDocument : rootNode ↪ 356
        ↪ ); 357
338            }
339        } 358
340        //draw(); 359
341    }
342
343    function processSearch() { 360
344        var rootNodes =[window].concat( ↪ 361
        ↪ getRootNodes()); 362
345        for (var i = 0; i < ↪ 363
        ↪ rootNodes.length; i++) { 364
346            var doc =rootNodes[i]. ↪ 365
        ↪ document || ↪

```

```

        ↪ rootNodes[i]. ↪
        ↪ contentDocument;
        if (!doc || !doc.body)
            continue;
        var frame =rootNodes[i]. ↪
        ↪ contentWindow || ↪
        ↪ rootNodes[i];
    }
    console.log("windowInner: "+ ↪
    ↪ window.innerHeight);
    console.log("docHeight: "+ ↪
    ↪ doc.height);
    console.log("frameHeight: " + ↪
    ↪ frame.innerHeight);
}

function draw() {
    var box = document.getElementById(' ↪
    ↪ type-ahead-box');
    if (!box) {
        box =document.createElement('TABOX') ↪
        ↪ ;
        box.id = 'type-ahead-box';
        document.documentElement.appendChild ↪
        ↪ (box);
        addStyle(styles);
    }
    box.style.display = 'block';
    box.style['background-color'] ="# ↪
    ↪ ff0000";

```

30

```

366 box.innerHTML = "<small>soup</small>";
367 }
368
369 function makeGridContainer(root) {
370     var parent = document.createElement('
    ↪ div');
371     parent.className = 'gridContainer';
372     $(root).append(parent);
373     return parent;
374 }
375
376 function createGrid(root) {
377     /*var rootNodes =[window].concat
    ↪ (getRootNodes());
378     for (var i = 0; i <
    ↪ rootNodes.length; i++) {
379         var doc =rootNodes[i].
    ↪ document ||
    ↪ rootNodes[i].
    ↪ contentDocument;
380         if (!doc || !doc.body)
381             continue;
382         var frame =rootNodes[i].
    ↪ contentWindow ||
    ↪ rootNodes[i];
383     }
384     console.log("windowInner: "+
    ↪ window.innerHeight);
385     console.log("docHeight: "+doc.height);
386     console.log("frameHeight: "+

```

```

    ↪ frame.innerHeight);*/
387
388 var anticipatedHeight =$(root).height
    ↪ ();
389 var cellHeight = anticipatedHeight/3;
390 // do not display text smaller than12
    ↪ px
391 // which on a low-density monitoris 12
    ↪ pt
392
393 console.log(anticipatedHeight);
394
395 var parent =document.createElement('
    ↪ div');
396 parent.className = 'grid';
397
398 for (var i = 0; i < 3; i++) {
399     var row =
    ↪ document.createElement
    ↪ ('div');
400     row.className = 'row';
401     for (var p = 0; p < 3; p++) {
402         var cell =
    ↪ document.createElement(
    ↪ 'div');
403         cell.className = "cell";
404         if (cellHeight>12) {
405             cell.innerHTML =7-(i*3-p);
406             cell.style.fontSize
    ↪ =Math.min(96,

```

```

407         ↪ cellHeight*0.5)+"px"
var trans = ↪
408         ↪ document.createElement ↪
         ↪ ('div');
trans.className = "backing" ↪
409         ↪ ;
cell.appendChild(trans);
410     }
411     row.appendChild(cell);
412 }
413 parent.appendChild(row);
414 }
415
416 //root.appendChild(parent);
417 $(root).append(parent);
418 }

```

0.10 ./src/lookup.js

```

1  // some of this code is borrowed from ↪
   ↪ tokland's typeahead-find project.
2  // https://code.google.com/p/chrome- ↪
   ↪ type-ahead/
3  // GNU GPL v3
4
5  define([], function() {
6
7  function setAlternativeActiveDocument(doc ↪
   ↪ ) {
8      function dom_trackActiveElement(evt) {

```

```

9      if (evt && evt.target) {
10         doc._tafActiveElement = (evt.target ↪
   ↪ == doc) ? null : evt.target;
11     }
12 }
13
14 function dom_trackActiveElementLost(evt ↪
   ↪ ) {
15     doc._tafActiveElement = null;
16 }
17
18 if (doc._tafActiveElement == undefined ↪
   ↪ && !doc.activeElement) {
19     doc._tafActiveElement = null;
20     doc.addEventListener("focus", ↪
   ↪ dom_trackActiveElement, true);
21     doc.addEventListener("blur", ↪
   ↪ dom_trackActiveElementLost, ↪
   ↪ true);
22 }
23 }
24
25 function upMatch(element, matchFunction) ↪
   ↪ {
26     while (element) {
27         var res = matchFunction(element);
28         if (res == null)
29             return null;
30         else if (res)
31             return element;

```

```

32     element = element.parentNode;
33 }
34 return element;
35 }
36
37
38 function getActiveElement(doc) {
39     return doc.activeElement || ↵
40         ↵ doc._tafActiveElement;
41 }
42
43 function isInputElementActive(doc) {
44     var element = getActiveElement(doc);
45     if (!element)
46         return;
47     var name = element.tagName.toLowerCase(↵
48         ↵ ());
49     if (["input", "select", "textarea", "↵
50         ↵ object", "embed"].indexOf(name) ↵
51         ↵ >=0)
52         return true;
53     return (upMatch(element, function(el) {
54         if (!el.getAttribute || ↵
55             ↵ el.getAttribute('↵
56             ↵ contenteditable') == 'false')
57         return null;
58         return el.getAttribute('↵
59             ↵ contenteditable');
60     })))
61 }

```

```

55
56 function getRootNodes() {
57     var rootNodes = new Array();
58     var frames = document.getElementsByTagName(↵
59         ↵ TagName('frame');
60     for (var i = 0; i < frames.length; i++)
61         rootNodes.push(frames[i]);
62     return rootNodes;
63 }
64
65 function processSearch() {
66     var rootNodes = [window].concat(↵
67         ↵ getRootNodes());
68     for (var i = 0; i < ↵
69         ↵ rootNodes.length; i++) {
70         var doc = rootNodes[i].↵
71             ↵ document || ↵
72             ↵ rootNodes[i].↵
73             ↵ contentDocument;
74         if (!doc || !doc.body)
75             continue;
76         var frame = rootNodes[i].↵
77             ↵ contentWindow || ↵
78             ↵ rootNodes[i];
79     }
80     // console.log("windowInner: "+↵
81     ↵ window.innerHeight);
82     // console.log("docHeight: "+↵
83     ↵ doc.height);
84     // console.log("frameHeight: "+↵

```



```

75     ↪ frame.innerHeight);
76 }
77 function is_shortcut(ev) {
78     var is_mac = ↪
79     ↪ navigator.appVersion.indexOf(" ↪
80     ↪ Mac") !== -1;
81     var is_windows = navigator.appVe ↪
82     ↪ rsion.indexOf("Windows") !== ↪
83     ↪ -1;
84     var is_alternative_input =(is_mac ↪
85     ↪ && ev.altKey) || (is_windows&& ↪
86     ↪ ev.ctrlKey && ev.altKey);
87     return !is_alternative_input &&( ↪
88     ↪ ev.altKey || ev.metaKey || ↪
89     ↪ ev.ctrlKey);
90 }
91 function stopEvent(ev) {
92     ev.preventDefault();
93     ev.stopPropagation();
94 }
95 return {setAlternativeActiveDocument: ↪
96     ↪ setAlternativeActiveDocument,
97     ↪ getRootNodes: getRootNodes,
98     ↪ isInputElementActive: ↪
99     ↪ isInputElementActive,
100     ↪ is_shortcut: is_shortcut,

```

```

94     stopEvent: stopEvent};
95 });

```

0.11 ./src/test.js

```

1  define(["lib/jquery-2.1.0.min", "lib/ ↪
2  ↪ within", "lib/Blob", "evaluator", " ↪
3  ↪ trace", "lookup", "testonly", "Grid" ↪
4  ↪ , "Crosshairs", "Flyout"], function ↪
5  ↪ (jq, within, blob, evaler, tracer, ↪
6  ↪ lookup, testonly, gridclass, ↪
7  ↪ crosshairclass, flyout) {
8  var trace = tracer.trace;
9  var supertrace = tracer.supertrace;
10
11  window.birchlabs = window.birchlabs ↪
12  ↪ || {};
13  var birchlabs = window.birchlabs;
14
15  window.evaluator = window.evaluator ↪
16  ↪ || {};
17  var evaluator = window.evaluator;
18
19  //console.log(birchlabs);
20
21  function makeContainers() {
22      var Flyout = birchlabs.Flyout;
23      var Grid = birchlabs.Grid;
24      var Crosshairs = ↪
25      ↪ birchlabs.Crosshairs;

```

```

17
18     if (birchlabs.flyoutsOn) {
19         Flyout.makecontainer(document.d
20             ↪ ocumentElement);
21         var flyouts = [];
22     }
23     Flyout.hide();
24
25     var grid = new Grid();
26     var container = makeGridContainer(
27         ↪ document.documentElement);
28
29     var crosshairs = new Crosshairs(
30         ↪ document.documentElement, grid);
31     crosshairs.hide();
32
33     birchlabs.grid = grid;
34     birchlabs.container = container;
35     birchlabs.crosshairs = crosshairs;
36     //birchlabs.theFlyout = flyout;
37     birchlabs.flyouts = flyouts;
38 }
39
40 function makeGridContainer(root) {
41     var parent = document.createElement('
42         ↪ div');
43     parent.className = '
44         ↪ chibiPoint_gridContainer
45         ↪ chibiPoint_hiddenGridContainer'
46         ↪ ;

```

```

40     $(root).append(parent);
41     return parent;
42 }
43
44 function init() {
45     birchlabs.evaluateTabMode = false;
46     birchlabs.disallowTabbing = false;
47     birchlabs.flyoutsOn = true;
48     if (birchlabs.evaluateTabMode) {
49         birchlabs.flyoutsOn = false;
50     }
51     makeContainers();
52     evaler.makeEvaluators();
53     var Flyout = birchlabs.Flyout;
54     var Grid = birchlabs.Grid;
55     var Crosshairs =
56         ↪ birchlabs.Crosshairs;
57
58     var flyouts = birchlabs.flyouts;
59     var grid = birchlabs.grid;
60     var crosshairs =
61         ↪ birchlabs.crosshairs;
62
63     birchlabs.targetedElement = null;
64
65     var keycodes = {
66         "backspace": 8,
67         "tab": 9,
68         "enter": 13,
69         "spacebar": 32,

```

```

68     "escape": 27,
69     "n": 78,
70     "p": 80,
71     "g": 71,
72     "f3": 114,
73     "f4": 115,
74     "dot": 46,
75     "zero": 48,
76     "activate": 167
77 };
78
79 // Do you have a numpad?
80 var numpad = true;
81
82 if (numpad) {
83     // Drilling: numpad
84     // 7 8 9
85     // 4 5 6
86     // 1 2 3
87     var numpadMappings = {"7": [55, 7], "↵" 110
88         ↵ "8": [56, 8], "9": [57, 9],
89         ↵ "4": [52, 4], "5"↵ 112
90         ↵ : [53, 5], "↵"
91         ↵ "6": [54, 6],
92         ↵ "1": [49, 1], "2"↵ 114
93         ↵ : [50, 2], "↵" 115
94         ↵ "3"↵ 116
95         ↵ : [51, 3]};
96     var zeroKey = keycodes.zero;
97     // Undo drilling:

```

```

92     // 0
93 }
94
95 // Are you one of.. those?
96 var dvorak = false;
97
98 if (dvorak) {
99     // Flyouts: right hand dvorak grid
100     // G C R
101     // H T N
102     // M W V
103     var flyoutShortcuts = [103, 99, 114,
104                             104, 116, 110,
105                             109, 119, 118];
106
107     if (!numpad) {
108         // Drilling: left hand dvorak↵
109         ↵ grid
110         // ' , .
111         // A O E
112         // ; Q J
113         var numpadMappings = {"7": [39, 7], ↵
114         ↵ "8": [44, 8], "9": [46, 9],
115         ↵ "4": [97, 4], "5": [111, 5], "6": [101, 6],
116         ↵ "1": [59, 1], "2": [113, 2], "3": [106, 3]};
117         // Undo drilling:
118         // K
119         var zeroKey = 107;
120     }
121 } else {

```

```

120 // Flyouts: left hand qwerty grid 147
121 // Q W E
122 // A S D
123 // Z X C 148
124 var flyoutShortcuts = [113,119,101, 149
125                        97,115,100, 150
126                        122,120,99]; 151
127
128 // alphabet flyouts 152
129 // nah, these won't help anyone 153
130 /*var flyoutShortcuts = [97,98,99, 154
131                        100,101,102, 155
132                        103,104,105];*/ 156
133
134 if (!numpad) { 157
135     // Flyouts: right hand qwerty ↵ 158
136     ↵ grid 159
137     // U I O 160
138     // J K L 161
139     // N M , 162
140     var flyoutShortcuts = ↵ 163
141     ↵ [117,105,111, 164
142         106,107,108, 165
143         109,44,46]; 166
144
145 // Drilling: left hand qwerty ↵ 167
146 ↵ grid 168
147 // Q W E
148 // A S D
149 // Z X C

```

```

var numpadMappings = {"7" ↵
    ↵ : [113,7], "8" : [119,8], "9" ↵
    ↵ : [101,9],
"4" : [97,4], "5" : [115,5], "6" : [100,6],
"1" : [122,1], "2" : [120,2], "3" : [99,3]};
// Undo drilling:
// V
var zeroKey = 118;
}
}

birchlabs.numpadMappings = ↵
    ↵ numpadMappings;

if (birchlabs.flyoutsOn) {
    for (var i=0; ↵
        ↵ i<flyoutShortcuts.length; i ↵
        ↵ ++ ) {
        flyouts.push(new Flyout( ↵
            ↵ flyoutShortcuts[i]));
    }
}

//createGrid();
function setEvents(rootNode) {
    var doc = ↵
        ↵ rootNode.contentDocument ↵
        ↵ || rootNode;
    var body = rootNode.body;

```

```

169         if (!body || !↵
170             ↵ body.addEventListener) {
171             return;
172         }
173         lookup.setAlternativeActive ↵
174             ↵ Document(document);
175
176         if (document.getElementById("↵
177             ↵ chibipointtestmode") != ↵
178             ↵ null) {
179             birchlabs.testmode = true;
180         }
181         //trace(birchlabs.testmode);
182         // usually on keydown
183         //processSearch();
184         //draw();
185         //var container =↵
186             ↵ birchlabs.container;
187
188         if (birchlabs.testmode) {
189             toggleGrid();
190         }
191         var $window = $(window),
192             resize_ok = true,
193             timer;
194
195         timer = setInterval(function() {
196             resize_ok = true;
197         }, 50);
198
199         liveTargeter = function() {
200             if (resize_ok === true) {
201                 resize_ok = false;
202                 //trace($window.width());
203                 //trace($(".verticalHair")↵
204                     ↵ .first().offset().↵
205                     ↵ left);
206                 //highlightTarget();
207                 crosshairs.highlight(getDocRoot());
208                 pointFlyouts();
209             }
210         };
211
212         $window.resize(liveTargeter);
213         $window.scroll(liveTargeter);
214
215         // some keys can only be caught ↵
216         ↵ by keydown
217         doc.addEventListener('keydown', function(↵
218             ↵ ev) {
219             var code = ev.keyCode;
220             var ascii = String.fromCharCode↵
221                 ↵ (code);
222
223             if (code == keycodes.enter) {

```

```

219 evaler.evaluatorIncrementKeys(code);
220     if(birchlabs.evaluateTabMode ↵ 239
        ↵ ) { 240
221     evaler.evaluatorWriteState ↵ 241
        ↵ ( ↵ 242
        ↵ document.activeElement ↵ 243
        ↵ ); 244
222 } else {
223     gridclick(grid); 245
224 246
225     lookup.stopEvent(ev); 247
226 }
227 } 248
228
229 if (code == keycodes.tab) { 249
230     if(birchlabs.evaluateTabMode ↵ 250
        ↵ ) { 251
231         // negative, for shift-tab ↵ 252
        ↵ ! 253
232         var code2 = ev.shiftKey? ↵ 254
        ↵ -code : code; 255
233 256
234 evaler.evaluatorIncrementKeys(code2); 257
235 evaler.checkFocusAfterWait(); 258
236 } else { 259
237     if( ↵ 260
        ↵ birchlabs.disallowTabbing ↵ 261
        ↵ ) {
238         alert("Please do not ↵ 262
            ↵ press tab during ↵

```

```

        ↵ this experiment!");
        lookup.stopEvent(ev);
document.activeElement.blur();
    }
    }
    /*if (code ==keycodes.spacebar ↵
        ↵ ) {
evaler.evaluatorIncrementKeys(code);
    }*/
    if(!birchlabs.evaluateTabMode) ↵
        ↵ {
        if (code ==keycodes.escape) ↵
            ↵ {
                if (!gridIsEmpty()) {
evaler.evaluatorIncrementKeys(code);
                    closeGrid();

                    lookup.stopEvent(ev);
                }
            }
        }
    });

    // no key bindings in tab mode
    if(!birchlabs.evaluateTabMode) {
doc.addEventListener('keypress',function ↵
    ↵ (ev) {
        if(lookup.isInputElementActive ↵
            ↵ (doc)) {

```

39

```

263         var tracer = ↵
                ↵ doc.getElementById("↵
                ↵ trace");
264         if (tracer != null) {
265     doc.getElementById("trace").innerHTML= ↵
        ↵ code;
266     }
267     return;
268 }
269 var code = ev.keyCode;
270 var ascii = String.fromCharCode ↵
        ↵ (code);
271
272     if (!lookup.is_shortcut(ev) && ↵
        ↵ ascii && [keycodes.enter]. ↵
        ↵ indexOf(code) == -1){
273         var tracer = ↵
                ↵ doc.getElementById("↵
                ↵ trace");
274         if (tracer != null) {
275     doc.getElementById("trace").innerHTML= ↵
        ↵ code;
276     }
277
278     /*if (numpadtype) {
279         var num = code-↵
                ↵ numpadstart;
280         if (num<10 && num > 0){
281             if (gridIsInUse()) {
282                 drill(num, grid, ↵

```

```

283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307

```

```

        ↵ crosshairs);
        lookup.stopEvent(ev);
    }
} else if (code == ↵
    ↵ keycodes.zero) {
    backup(grid, crosshairs);
    lookup.stopEvent(ev);
}
}*/
for (var j in numpadMappings) ↵
    ↵ {
    if (code == numpadMappings[↵
        ↵ j][0]) {
        if (gridIsInUse()) {
evaler.evaluatorIncrementKeys(code);
drill(numpadMappings[j][1], grid, ↵
    ↵ crosshairs);
        lookup.stopEvent(ev);
        break;
    }
    }
    if (code == zeroKey) {
evaler.evaluatorIncrementKeys(code);
        backup(grid, crosshairs);
        lookup.stopEvent(ev);
    }
}

```

```

308         }
309         if (code == keycodes.activate ↵ 332
            ↵ ) {
310     evaler.evaluatorIncrementKeys(code); 334
311         toggleGrid();
312
313         lookup.stopEvent(ev);
314     }
315     if (birchlabs.flyoutsOn) {
316         for (i=0; ↵ 337
            ↵ i<flyoutShortcuts.length ↵ 338
            ↵ i++) {
317             if (code == ↵ 339
                ↵ flyoutShortcuts[i]) ↵ 340
                ↵ {
318     evaler.evaluatorIncrementKeys(code); 342
319         evaler.evaluatorWri ↵ 343
            ↵ teState(flyouts[i] ↵
            ↵ ].getTarget());
320         flyouts[i].doClick();
321
322         closeGrid();
323
324         lookup.stopEvent(ev);
325         break;
326     }
327 }
328 }
329 }
330 }, false);

```

```

    }
}

// guess this defers setting events ↵
↵ until root is loaded?
var rootNodes = [document].concat( ↵
    ↵ lookup.getRootNodes());
for (var i = 0; i < ↵
    ↵ rootNodes.length; i++) {
    var rootNode = rootNodes[i];
    if (rootNode.contentDocument) {
rootNode.addEventListener('load', ↵
    ↵ function(ev) {
setEvents(ev.target.contentDocument);
    });
    }
    else if (!rootNode.contentDocument ↵
        ↵ || ↵
        ↵ rootNode.contentDocument.readyState ↵
        ↵ == 'complete') {
setEvents(rootNode.contentDocument ? ↵
    ↵ rootNode.contentDocument : rootNode ↵
    ↵ );
    }
    //draw();
}

function getDocRoot() {
    var rootNodes = [window].concat(1 ↵

```



```

352     ↳ lookup.getRootNodes()); 376
    for (var x = 0; x < rootNodes.length; 377
        ↳ x++) { 378
353     var doc2 = rootNodes[x].document || 379
        ↳ rootNodes[x].contentDocument; 380
354     if (!doc2 || !doc2.body) 381
355         continue; 382
356     return doc2; 383
357 } 384
358 } 385
359 386
360 function 387
    ↳ getAllClickablesInBoundsAndSort( 388
    ↳ root, rect, rects) { 389
361     var found = []; 390
362     var buckets = []; 391
363     for (var j=0; j<rects.length; j++) 392
        ↳ { 393
364         buckets.push([]); 394
365     } 395
366 396
367     var left = rect.left; 397
368     var top = rect.top; 398
369     var width = rect.width; 399
370     var height = rect.height; 400
371 401
372     var left2; 402
373     var top2; 403
374     var width2; 404
375     var height2; 405

```

```

    var cache = false;

    var r;
    var i;
    var res;
    var res2;
    //var q;
    //var ew;
    var itRect;
    // some of the obscure ones are
    ↳ guesses
    var careElements = {"A":true,
                        "INPUT":true,
"TEXTAREA":true,
                        "SELECT":true,
"BUTTON":true//,
                        // "output":true,
                        // "command":true,
                        // "kbd":true
                        };

    $(root).find("[_handlerTypes],A, 387
        ↳ INPUT, SELECT, TEXTAREA, 388
        ↳ BUTTON").each(function() { 389
    var found = false; 390
    var handlers = this.getAttribute( 391
        ↳ '_handlerTypes'); 392
    if (handlers != null) { 393
        var h = handlers.split(","); 394
    }

```

```

402         for (i=0; i<h.length; i++) { 419
403             if (h[i] == 'click' || h[i] ↵
404                 ↵ == 'mousedown') {
405                 found = true; 420
406                 break; 421
407             }
408         }
409         if (!found) {
410     if(careElements[this.nodeName]) found= ↵
411         ↵ true; 422
412     } 423
413     if (found) {
414         //if ($(this).css("visibility" ↵ 424
415         ↵ ").indexOf("hidden") != ↵ 425
416         ↵ -1 || !$(this).css(" ↵ 426
417         ↵ display").indexOf("none") ↵
418         ↵ !=-1) return; 427
419         var visible = !$(this).is(': ↵ 428
420         ↵ hidden') &&$(this).is(': ↵ 429
421         ↵ visible');
422         if (visible) { 430
423             //if(this == ↵ 431
424             ↵ document.documentElement ↵ 432
425             ↵ ) return this.ret.push( ↵ 433
426             ↵ this); 434
427         if(this == ↵
428             ↵ document.documentElement ↵
429             ↵ ) return;
430     }
431 }
432 }
433 }
434 }
435 }
436 }
437 }
438 }
439 }
440 }
441 }
442 }
443 }
444 }
445 }
446 }
447 }
448 }
449 }
450 }
451 }
452 }
453 }
454 }
455 }
456 }
457 }
458 }
459 }
460 }
461 }
462 }
463 }
464 }
465 }
466 }
467 }
468 }
469 }
470 }
471 }
472 }
473 }
474 }
475 }
476 }
477 }
478 }
479 }
480 }
481 }
482 }
483 }
484 }
485 }
486 }
487 }
488 }
489 }
490 }
491 }
492 }
493 }
494 }
495 }
496 }
497 }
498 }
499 }
500 }
501 }
502 }
503 }
504 }
505 }
506 }
507 }
508 }
509 }
510 }
511 }
512 }
513 }
514 }
515 }
516 }
517 }
518 }
519 }
520 }
521 }
522 }
523 }
524 }
525 }
526 }
527 }
528 }
529 }
530 }
531 }
532 }
533 }
534 }
535 }
536 }
537 }
538 }
539 }
540 }
541 }
542 }
543 }
544 }
545 }
546 }
547 }
548 }
549 }
550 }
551 }
552 }
553 }
554 }
555 }
556 }
557 }
558 }
559 }
560 }
561 }
562 }
563 }
564 }
565 }
566 }
567 }
568 }
569 }
570 }
571 }
572 }
573 }
574 }
575 }
576 }
577 }
578 }
579 }
580 }
581 }
582 }
583 }
584 }
585 }
586 }
587 }
588 }
589 }
590 }
591 }
592 }
593 }
594 }
595 }
596 }
597 }
598 }
599 }
600 }
601 }
602 }
603 }
604 }
605 }
606 }
607 }
608 }
609 }
610 }
611 }
612 }
613 }
614 }
615 }
616 }
617 }
618 }
619 }
620 }
621 }
622 }
623 }
624 }
625 }
626 }
627 }
628 }
629 }
630 }
631 }
632 }
633 }
634 }
635 }
636 }
637 }
638 }
639 }
640 }
641 }
642 }
643 }
644 }
645 }
646 }
647 }
648 }
649 }
650 }
651 }
652 }
653 }
654 }
655 }
656 }
657 }
658 }
659 }
660 }
661 }
662 }
663 }
664 }
665 }
666 }
667 }
668 }
669 }
670 }
671 }
672 }
673 }
674 }
675 }
676 }
677 }
678 }
679 }
680 }
681 }
682 }
683 }
684 }
685 }
686 }
687 }
688 }
689 }
690 }
691 }
692 }
693 }
694 }
695 }
696 }
697 }
698 }
699 }
700 }
701 }
702 }
703 }
704 }
705 }
706 }
707 }
708 }
709 }
710 }
711 }
712 }
713 }
714 }
715 }
716 }
717 }
718 }
719 }
720 }
721 }
722 }
723 }
724 }
725 }
726 }
727 }
728 }
729 }
730 }
731 }
732 }
733 }
734 }
735 }
736 }
737 }
738 }
739 }
740 }
741 }
742 }
743 }
744 }
745 }
746 }
747 }
748 }
749 }
750 }
751 }
752 }
753 }
754 }
755 }
756 }
757 }
758 }
759 }
760 }
761 }
762 }
763 }
764 }
765 }
766 }
767 }
768 }
769 }
770 }
771 }
772 }
773 }
774 }
775 }
776 }
777 }
778 }
779 }
780 }
781 }
782 }
783 }
784 }
785 }
786 }
787 }
788 }
789 }
790 }
791 }
792 }
793 }
794 }
795 }
796 }
797 }
798 }
799 }
800 }
801 }
802 }
803 }
804 }
805 }
806 }
807 }
808 }
809 }
810 }
811 }
812 }
813 }
814 }
815 }
816 }
817 }
818 }
819 }
820 }
821 }
822 }
823 }
824 }
825 }
826 }
827 }
828 }
829 }
830 }
831 }
832 }
833 }
834 }
835 }
836 }
837 }
838 }
839 }
840 }
841 }
842 }
843 }
844 }
845 }
846 }
847 }
848 }
849 }
850 }
851 }
852 }
853 }
854 }
855 }
856 }
857 }
858 }
859 }
860 }
861 }
862 }
863 }
864 }
865 }
866 }
867 }
868 }
869 }
870 }
871 }
872 }
873 }
874 }
875 }
876 }
877 }
878 }
879 }
880 }
881 }
882 }
883 }
884 }
885 }
886 }
887 }
888 }
889 }
890 }
891 }
892 }
893 }
894 }
895 }
896 }
897 }
898 }
899 }
900 }
901 }
902 }
903 }
904 }
905 }
906 }
907 }
908 }
909 }
910 }
911 }
912 }
913 }
914 }
915 }
916 }
917 }
918 }
919 }
920 }
921 }
922 }
923 }
924 }
925 }
926 }
927 }
928 }
929 }
930 }
931 }
932 }
933 }
934 }
935 }
936 }
937 }
938 }
939 }
940 }
941 }
942 }
943 }
944 }
945 }
946 }
947 }
948 }
949 }
950 }
951 }
952 }
953 }
954 }
955 }
956 }
957 }
958 }
959 }
960 }
961 }
962 }
963 }
964 }
965 }
966 }
967 }
968 }
969 }
970 }
971 }
972 }
973 }
974 }
975 }
976 }
977 }
978 }
979 }
980 }
981 }
982 }
983 }
984 }
985 }
986 }
987 }
988 }
989 }
990 }
991 }
992 }
993 }
994 }
995 }
996 }
997 }
998 }
999 }
1000 }

```

```

itRect = ↵
    ↵ this.getBoundingClientRect ↵
    ↵ ();

res = !( (itRect.top > top+ ↵
    ↵ height) || (itRect.top+ ↵
    ↵ itRect.height < top) || ↵
    ↵ (itRect.left > left+ ↵
    ↵ width ) || (itRect.left+ ↵
    ↵ itRect.width < left));

// it's certainly in the grid ↵
    ↵ somewhere
if(res) {

    // now find which buckets ↵
    ↵ to put it in
    for (i=0; i<rects.length; ↵
        ↵ i++) {
        r = rects[i];
        left2 = r.left;
        top2 = r.top;
        width2 = r.width;
        height2 = r.height;

        res2 = !((itRect.top ↵
            ↵ > top2+height2) ↵
            ↵ || (itRect.top + ↵
            ↵ itRect.height < ↵
            ↵ top2) || (↵

```

```

435         ↪ itRect.left > ↪
436         ↪ left2+width2 ) || ↪ 457
437         ↪ (itRect.left+↪ 458
438         ↪ itRect.width < ↪ 459
439         ↪ left2)); 460
440     } 461
441     } 462
442     } 463
443     }); 464
444     return buckets; 465
445 } 466
446 467
447 function highlightTarget() { 472
448     var crosshairs = ↪
449         ↪ birchlabs.crosshairs;
450     var flyouts = birchlabs.flyouts;
451     var grid = birchlabs.grid; 473
452     var doc2 = getDocRoot(); 474
453     475
454     var crosshaired = ↪ 476
455         ↪ crosshairs.highlight(doc2).↪ 477
456         ↪ get(0); 478
457     // some of the obscure ones are ↪

```

```

457     ↪ guesses
458     var careElements = {"A":true,
459         "INPUT":true,
460         "TEXTAREA":true,
461         "SELECT":true,
462         "BUTTON":true//,
463         //"output":true,
464         //"command":true,
465         //"kbd":true
466     };
467
468     var rect = grid.getLastGrid().get↪
469         ↪ (0).getBoundingClientRect();
470
471     var rects = [];
472     for (var i=0; i<3; i++) {
473         for (var j=0; j<3; j++) {
474             rects.push(grid.getLastRows().↪
475                 ↪ eq(i).children().eq(j).↪
476                 ↪ get(0).getB↪
477                 ↪ oundingClientRect());
478         }
479     }
480     if (birchlabs.flyoutsOn) {
481         var buckets =getAllClickables↪
482             ↪ InBoundsAndSort($(doc2.body↪
483             ↪ ), rect,rects);
484
485         // which element was selectedby ↪
486         ↪ each segment

```

```

479     var selected = [];
480     //console.log(crosshaired);
481
482     var bucketIndex = 0;
483     // one bucket for each segment
484     for (var i=0; i<3; i++) {
485         for (var j=0; j<3; j++) {
486             var fulfilled = false;
487             for (var b=0; b<buckets.length;
488                 ↪ b++) {
489                 // check everything in my
490                 ↪ allocated bucket first
491                 // but be prepared to cycle
492                 ↪ round to another bucket
493                 ↪ !
494                 var myBucket = buckets[(b
495                 ↪ ucketIndex+b)%
496                 ↪ buckets.length];
497                 for (var n = 0;
498                 ↪ n<myBucket.length; n++)
499                 ↪ {
500                 var inSelectedAlready =
501                 ↪ false;
502                 if (myBucket[n] !=
503                 ↪ crosshaired) {
504                     for (var s = 0;
505                     ↪ s<selected.length;
506                     ↪ s++) {
507                         // skip over
508                         if (selected[s] ==

```

```

509                 ↪ myBucket[n])
510                 {
511                     inSelectedAlready=
512                     ↪ true;
513                     break;
514                 }
515                 if(!inSelectedAlready) {
516                     // add to selected
517                     selected[bucketIndex] = myBucket[n];
518                     fulfilled = true;
519                     break;
520                 }
521             }
522             if (fulfilled) {
523                 break;
524             }
525             bucketIndex++;
526         }
527     }
528
529     for (var i=0; i<flyouts.length; i
530     ↪ ++){
531         var f = flyouts[i];
532         var targ = selected[i];
533         f.unsetTarget();
534         f.hide();
535         if (targ) {

```

```

524         f.setTarget(targ, ↵
           ↵ grid.getFirstGrid().get ↵
           ↵ (0));
525     }
526 }
527 //     var delayPoint = setInterval(↵
           ↵ function () {
528 //         clearInterval(delayPoint);
529         for (var i=0; i<flyouts.length; ↵
           ↵ i++) {
530             var f = flyouts[i];
531             var targ = selected[i];
532             if (targ) {
533                 f.show();
534             }
535         }
536         pointFlyouts();
537         // }, 50);
538         //trace(selected);
539     }
540 }
541
542 function unpointFlyouts() {
543     if (birchlabs.flyoutsOn) {
544         var flyouts = birchlabs.flyouts;
545
546         for (var i=0; i<flyouts.length; i ↵
           ↵ ++)) {
547             var f = flyouts[i];
548             f.unsetTarget();
549
550             f.hide();
551         }
552     }
553 }
554 function pointFlyouts() {
555     var grid = birchlabs.grid;
556
557     if (birchlabs.flyoutsOn) {
558         var flyouts = birchlabs.flyouts;
559
560         for (var i=0; i<flyouts.length; i ↵
           ↵ ++)) {
561             var f = flyouts[i];
562             f.point(grid.findPointerCoords());
563         }
564     }
565 }
566
567 function createGrid(root) {
568     var numpadMappings = ↵
           ↵ birchlabs.numpadMappings;
569
570     var anticipatedHeight = $(root). ↵
           ↵ height();
571     var cellHeight = anticipatedHeight/3;
572     // do not display text smaller than ↵
           ↵ 12px
573     // which on a low-density monitor is ↵
           ↵ 12pt

```

```

574
575     var parent = document.createElement 592
576         ↪ ('div'); 593
577     parent.className = 'chibiPoint_grid' 594
578         ↪ chibiPoint_gridBorderOn'; 595
579
580     var drawBorders = false; 596
581     // if it's not too small, give it 597
582         ↪ borders
583     var rect = $(root).get(0).getBo 598
584         ↪ undingClientRect();
585     if (rect.width>60 &&rect.height 599
586         ↪ >60) {
587         drawBorders = true;
588     }
589
590     for (var i = 0; i < 3; i++) {
591         var row = 600
592         ↪ document.createElement(' 601
593         ↪ div');
594         row.className = 'chibiPoint_row' 602
595         ↪ '
596         for (var p = 0; p < 3; p++) {
597             var cell = 601
598             ↪ document.createElement 602
599             ↪ ('div'); 603
600             if (drawBorders) { 604
601                 cell.className = " 605
602                 ↪ chibiPoint_cell 606
603                 ↪ chibiPoint_cellBorder 607

```

```

        ↪ "
    } else {
        cell.className = " 604
        ↪ chibiPoint_cell";
    }

    if (cellHeight>12) {
        cell.innerHTML = Stri 605
        ↪ ng.fromCharCode( 606
        ↪ numpadMappings[(7-( 607
        ↪ i*3-p)).toString() 608
        ↪ ][0]).toUpperCase() 609
        ↪ ;
        cell.style.fontSize 610
        ↪ =Math.min(96, 611
        ↪ cellHeight*0.5)+"px 612
        ↪ "
        var trans = 613
        ↪ document.createElement 614
        ↪ ('div');
        trans.className = " 615
        ↪ chibiPoint_backing" 616
        ↪ ;
        cell.appendChild(trans);
    }
    row.appendChild(cell);
    parent.appendChild(row);
}

```

```

608     //root.appendChild(parent);
609     $(root).append(parent);
610 }
611
612 function drill(num, grid, crosshairs) {
613     var Flyout = birchlabs.Flyout;
614
615     grid.getLastCells().text("");
616     grid.getLastGrid().addClass("chib
        ↳ iPoint_gridBorderOff");
617     grid.getLastCells().addClass("chi
        ↳ biPoint_cellBorderOff");
618     grid.getLastGrid().removeClass("c
        ↳ hibiPoint_gridBorderOn");
619     grid.getLastCells().removeClass("
        ↳ chibiPoint_cellBorderOn");
620
621     var rowIndexed1 = 4-Math.ceil(num/3);
622     var columnIndexed1 = ((num-1)%3)+1;
623     //cell.innerHTML = 7-(i*3-p);
624
625     selectArray = [];
626     selectArray.push(grid.getLatestSe
        ↳ lector());
627     selectArray.push(".chibiPoint_grid");
628     selectArray.push(".chibiPoint_row:
        ↳ nth-of-type("+rowIndexed1+")");
629     selectArray.push(".chibiPoint_cell:
        ↳ nth-of-type("+columnIndexed1+")");
630     grid.pushHistory(selectArray.join(""));

```

```

631     createGrid($(grid.getLatestSelector
        ↳ ()).first());
632
633     // move crosshairs
634     crosshairs.updatePosition(grid);
635
636     highlightTarget();
637     Flyout.show();
638
639 }
640
641 function backup(grid, crosshairs) {
642     var numpadMappings =
        ↳ birchlabs.numpadMappings;
643     var Flyout = birchlabs.Flyout;
644
645     if (gridIsInUse()) {
646
647         // remove a grid!
648         $(grid.getLatestSelector()).empty();
649         grid.getPreviousSelector();
650
651         //trace(grid.getLatestSelector());
652
653         var anticipatedHeight = $(
        ↳ grid.getLatestSelector()).
        ↳ height();
654         var cellHeight = anticipatedHeight
        ↳ /3;

```

```

656     grid.getLastRows().each(function(↵ 674
        ↵ index) {
657         var i = index;
658         $(this).children().each(fun↵
            ↵ ction(index) {
659             var p = index;
660             if (cellHeight>12) {
661                 $(this).text(String.fro↵ 677
                    ↵ mCharCode(↵
                    ↵ numpadMappings[(7-(i↵ 678
                    ↵ *3-p)).toString()↵ 679
                    ↵ ][0]).toUpperCase()); 680
662             }
663             var trans =↵ 682
                ↵ document.createElement↵
                ↵ ('div');
664             trans.className = "↵
                ↵ chibiPoint_backing";
665             $(this).append(trans);
666         }
667     });
668 });
669
670 grid.getLastGrid().removeClass("↵ 691
    ↵ chibiPoint_gridBorderOff");
671 grid.getLastCells().removeClass("↵ 692
    ↵ chibiPoint_cellBorderOff");
672 grid.getLastGrid().addClass("ch↵ 693
    ↵ ibiPoint_gridBorderOn");
673

```

```

// if it's not too small, giveit ↵
    ↵ borders
var rect = $(grid.getLatestSele↵
    ↵ ctor()).get(0).↵
    ↵ getBoundingClientRect();
if (rect.width>60 &&rect.height ↵
    ↵ >60) {
    grid.getLastCells().addClass("↵
        ↵ chibiPoint_cellBorderOn");
}

unpointFlyouts();

var exists =grid.getLastGrid().↵
    ↵ length>0;
if (exists) {
    // move crosshairs
    crosshairs.updatePosition(grid);
    Flyout.show();
    highlightTarget();
} else {
    closeGrid();
}
}

function clickOrFocus(element) {
    evaler.clickOrFocus(element);
    closeGrid();
}

```



```

698
699 function gridclick(grid) {
700     var coords = grid.findPointerCoords();
701     ↪ ;
702     var centerX = coords.centerX;
703     var centerY = coords.centerY;
704     var rootNodes = [window].concat(lookup.getRootNodes());
705     for (var i = 0; i < rootNodes.length; i++) {
706         var doc2 = rootNodes[i].document || rootNodes[i].contentDocument;
707         if (!doc2 || !doc2.body)
708             continue;
709
710         var element = $(doc2.elementFromPoint(centerX, centerY));
711
712         element.removeClass("chibiPoint_targeted");
713         element.addClass("chibiPoint_cluck");
714
715         var flash = setInterval(function() {
716             ↪ {
717                 if (element.hasClass("chibiPoint_cluck")) {
718                     ↪ oint_cluck");
719                     } else {
720                         element.addClass("chibiPoint_cluck");
721                     }
722                 }, 100);
723
724                 var delayUnclick = setInterval(function() {
725                     ↪ function () {
726                         clearInterval(flash);
727                         clearInterval(delayUnclick);
728                         element.removeClass("chibiPoint_cluck");
729                         ↪ oint_cluck");
730                         clickOrFocus(element.get(0));
731                         pointFlyouts();
732                     }, 500);
733                     //element.get(0).click();
734
735                 }
736             }
737         }
738     }
739
740     function shortcut() {
741         if (!birchlabs.evaluateTabMode) {
742             toggleGrid();
743             evaler.evaluatorIncrementKeys("Cmd_K");
744         }
745     }
746
747     function toggleGrid() {

```

```

745     if (!birchlabs.evaluateTabMode) {
746         var Flyout = birchlabs.Flyout;
747
748         var crosshairs = ↵
749             ↵ birchlabs.crosshairs;
750         var container = ↵
751             ↵ birchlabs.container;
752         var grid = birchlabs.grid;
753         document.activeElement.blur();
754         //document.documentElement.focus();
755         if (gridIsEmpty()) {
756             // need to make a grid
757             grid.initialize();
758             $(".chibiPoint_gridContainer").↵
759                 ↵ removeClass("↵
760                 ↵ chibiPoint_hiddenGridC↵
761                 ↵ ontainer");
762             createGrid(container);
763
764             //crosshairs =crosshairs||new ↵
765                 ↵ birchlabs.Crosshairs(↵
766                 ↵ document.documentElement, ↵
767                 ↵ grid);
768             //birchlabs.crosshairs =↵
769                 ↵ crosshairs;
770
771             // in case crosshairs already↵
772                 ↵ instantiated

```

```

765         crosshairs.updatePosition(grid);
766         Flyout.show();
767         highlightTarget();
768         crosshairs.show();
769     } else {
770         // toggle grid off
771         closeGrid();
772     }
773 }
774
775 function closeGrid() {
776     var Flyout = birchlabs.Flyout;
777
778     var crosshairs = ↵
779         ↵ birchlabs.crosshairs;
780     var grid = birchlabs.grid;
781
782     grid.initialize();
783     $(grid.getLatestSelector()).empty();
784
785     $(".chibiPoint_gridContainer").ad↵
786         ↵ dClass("↵
787         ↵ chibiPoint_hiddenGridContainer"↵
788         ↵ );
789
790     unpointFlyouts();
791     Flyout.hide();
792     crosshairs.hide();
793 }

```

51

```

791
792 function gridIsEmpty() {
793     return $(".chibiPoint_gridContainer" ↵
794         ↵ ).children().length == 0;
795 }
796
797 function gridIsInUse() {
798     //var grid = birchlabs.grid;
799     return $(".chibiPoint_grid").chil ↵
800         ↵ dren().length > 0;
801 }
802
803     return { init: init,
804             createGrid: createGrid,
805             shortcut: shortcut};
806 });

```

0.12 ./src/testonly.js

```

1  define(["trace", "test"],function(tracer ↵
2      ↵ , test) {
3      var trace = tracer.trace;
4
5      window.birchlabs =window.birchlabs ↵
6          ↵ ||{};
7      var birchlabs = window.birchlabs;
8
9      function newInterface(container) {
10         console.log(test);
11     }

```

```

10         // need to make a grid
11         test.createGrid(container);
12     }
13
14     return {newInterface: newInterface};
15 }

```

0.13 ./src/trace.js

```

1  define([], function() {
2
3      everLogged = false;
4
5      function trace(message) {
6          var logger =document.getElementById(" ↵
7              ↵ logger");
8          if (logger != null) {
9              if (!everLogged) {
10                 logger.value = "";
11                 everLogged = true;
12             }
13             proposedValue = message +
14                 "\n" + logger.value;
15
16             lines =proposedValue.split("\n", ↵
17                 ↵ 100);
18
19             culled = lines.join("\n");
20             logger.value = culled;
21         }
22     }

```

52

```

20 console.log(message);
21 }
22
23 function supertrace(o) {
24   for (var j in o) {
25     trace(j + ": " + o[j]);
26   }
27 }
28
29 function supertraceInline(o) {
30   var traces = [];
31   for (var j in o) {
32     traces.push(j + ": " + o[j]);
33   }
34   // why array.join() isn't working ↗
35   // ↗ haunts me
36   var str = "";
37   for (i=0; i<traces.length; i++) {
38     if (i!=0) {
39       str += ", \t";
40     }
41     str += traces[i];
42   }
43   trace(str);
44 }
45
46 return {trace: trace,
47         supertrace: supertrace,
48         supertraceInline: ↗
49         ↗ supertraceInline};

```

48 });

0.14 ./src/Grid.js

```

1  define(["lib/jquery-2.1.0.min", "lib/↗
   ↗ within"], function(jq, within) {
2
3  window.birchlabs = window.birchlabs||{};
4  var birchlabs = window.birchlabs;
5
6  (function(){
7    var Grid = function() {
8      this.initialize();
9    }
10   var p = Grid.prototype;
11
12   p.initialize = function() {
13     this.selectHistory = ["↗
14     ↗ chibiPoint_gridContainer"];
15   };
16
17   p.getLatestSelector = function() {
18     return this.selectHistory[this.se↗
19     ↗ lectHistory.length-1];
20   }
21
22   p.getPreviousSelector = function() {
23     if (this.selectHistory.length>1) {
24       this.selectHistory.pop(this.sel↗
25       ↗ ectHistory.length-1);

```

```

23     }
24     return this.getLatestSelector();
25 }
26
27 p.pushHistory = function(input) {
28     this.selectHistory.push(input);
29 }
30
31 p.exists = function() {
32     return this.selectHistory.length>1;
33 }
34
35 p.getLastRows = function() {
36     return$(this.getLatestSelector()+".↵
↵ chibiPoint_row");
37 };
38
39 p.getLastCells = function() {
40     return$(this.getLatestSelector()+".↵
↵ chibiPoint_cell");
41 };
42
43 p.getLastGrid = function() {
44     return$(this.getLatestSelector()+".↵
↵ chibiPoint_grid").first();
45 };
46
47 p.getFirstGrid = function() {
48     return $(this.selectHistory[0]+".↵
↵ chibiPoint_grid").first();

```

```

49 };
50
51 p.findPointerCoords = function() {
52     var element = this.getLastGrid();
53     // maybe there are no grids made
54     if (element.get(0)) {
55         // click
56         //var offset =gridElement.offset();
57         var offset = element.get(0).getB↵
↵ oundingClientRect();
58         //console.log($this.get(0));
59         //console.log($this.offset());
60         var width = element.outerWidth();
61         var height = element.outerHeight();
62
63         var centerX = offset.left +width / ↵
↵ 2;
64         var centerY = offset.top +height / ↵
↵ 2;
65
66         //console.log("x: "+centerX+" y:"+↵
↵ centerY);
67         return {centerX:centerX,↵
↵ centerY:centerY};
68
69     } else {
70         return null;
71     }
72 };
73

```

```

74     birchlabs.Grid = Grid;
75   })();
76
77   });

```

0.15 ./src/Crosshairs.js

```

1  define(["lib/jquery-2.1.0.min", "lib/
    ↪ within"], function(jq, within) {
2
3  window.birchlabs = window.birchlabs || {};
4  var birchlabs = window.birchlabs;
5
6  (function() {
7    var Crosshairs = function(root, grid) {
8      this.initialize(root, grid);
9    };
10
11    var p = Crosshairs.prototype;
12
13    p.setX = function(x) {
14      this.x = x;
15
16      // convert to percent
17      var elem = this.grid.getFirstGrid();
18      var rect = elem.get(0).
    ↪ getBoundingClientRect();
19      var pX = x*100/rect.width;
20      $(".chibiPoint_verticalHair").first
    ↪ ().css({left:pX+"%"});

```

```

21   };
22
23   p.setY = function(y) {
24     this.y = y;
25
26     // convert to percent
27     var elem = this.grid.getFirstGrid();
28     var rect = elem.get(0).
    ↪ getBoundingClientRect();
29     var pY = y*100/rect.height;
30
31     $(".chibiPoint_horizontalHair").f
    ↪ irst().css({top:pY+"%"});
32   };
33
34   p.show = function() {
35     this.hairs.className = "
    ↪ chibiPoint_crosshairs
    ↪ chibiPoint_shownCrosshairs";
36   };
37
38   p.hide = function() {
39     this.hairs.className = "
    ↪ chibiPoint_crosshairs
    ↪ chibiPoint_hiddenCrosshairs";
40     this.removeHighlights();
41   };
42
43   p.createCrosshairs = function(root) {
44     this.hairs = document.createElement('
    ↪ div');

```

```

44   this.hairs.className = "↵
      ↵ chibiPoint_crosshairs";
45
46   this.horizontalHair = ↵
      ↵ document.createElement('div');
47   this.horizontalHair.className = "↵
      ↵ chibiPoint_horizontalHair";
48
49   this.verticalHair = ↵
      ↵ document.createElement('div');
50   this.verticalHair.className = "↵
      ↵ chibiPoint_verticalHair";
51
52   $(this.hairs).append(this.horizon↵
      ↵ talHair);
53   $(this.hairs).append(this.vertica↵
      ↵ lHair);
54
55   $(root).append(this.hairs);
56 };
57
58 p.updatePosition = function() {
59   var coords = ↵
      ↵ this.grid.findPointerCoords();
60   if (coords) {
61     var centerX = coords.centerX;
62     var centerY = coords.centerY;
63
64     this.setX(centerX);
65     this.setY(centerY);

```

```

66   }
67 };
68
69 p.highlight = function(root) {
70   var coords = ↵
      ↵ this.grid.findPointerCoords();
71   if (coords) {
72     var centerX = coords.centerX;
73     var centerY = coords.centerY;
74
75     var element = $(↵
      ↵ root.elementFromPoint(centerX, ↵
      ↵ centerY));
76
77     // unhighlight previous element
78     this.removeHighlights();
79     this.targetedElement = element;
80
81     element.addClass("chibiPoint_tar↵
      ↵ geted");
82
83     // doesn't seem to work :(
84     /*element.trigger('mouseover');
85     element.trigger('hover');
86     element.trigger('mouseenter');
87     element.mouseover();
88     element.hover();
89     element.mouseenter();*/
90     //element.get(0).mouseover();
91     //element.get(0).hover();

```

```

92     return element;
93   } else {
94     return null;
95   }
96 };
97
98
99 p.removeHighlights = function() {
100   if (this.targetedElement) {
101     this.targetedElement.removeClass(
102       ↪ "chibiPoint_targeted");
103     this.targetedElement.removeClass(
104       ↪ "chibiPoint_cluck");
105   }
106 };
107
108 p.initialize = function(root, grid) {
109   this.targetedElement = null;
110   this.createCrosshairs(root);
111   this.grid = grid;
112   this.updatePosition();
113   this.show();
114 };
115 birchlabs.Crosshairs = Crosshairs;
116 })();
117 });

```

0.16 ./src/Flyout.js

```

1  define(["lib/jquery-2.1.0.min", "lib/
   ↪ within", "evaluator"], function(jq, ↪
   ↪ within, evaler) {
2
3  window.birchlabs = window.birchlabs || {};
4  var birchlabs = window.birchlabs;
5
6  (function() {
7    var Flyout = function(root, label) {
8      this.initialize(root, label);
9    };
10
11    Flyout.latestId = 0;
12
13    Flyout.getNextId = function() {
14      return Flyout.latestId++;
15    };
16
17    Flyout.getContainer = function(root) {
18      return this.container || this.makeco
19         ↪ ntainer(root);
20    };
21
22    Flyout.makecontainer = function(root) {
23      this.container = ↪
24         ↪ document.createElement('div');
25      this.container.className = "↪

```



```

24         ↪ flyoutContainer hiddenfC";
25     $(root).append(this.container);
26     return this.container;
27 };
28
29 var p = Flyout.prototype;
30
31 p.setX = function(x) {
32     var rect = Flyout.getContainer().↪
33         ↪ getBoundingClientRect();
34     x = Math.min(Math.max(2, x),↪
35         ↪ rect.width-22);
36     this.x = x;
37
38     // convert to percent
39     var pX = x*100/rect.width;
40     $(this.flyout).css({left:pX+"%"});
41     //$(this.flyout).css({left:x});
42     //$(this.flyout).css({left:x});
43 };
44
45 p.setY = function(y) {
46     var rect = Flyout.getContainer().↪
47         ↪ getBoundingClientRect();
48     y = Math.min(Math.max(2, y),↪
49         ↪ rect.height-22);
50     this.y = y;
51
52     // convert to percent

```

```

49     var pY = y*100/rect.height;
50     $(this.flyout).css({top:pY+"%"});
51     //$(this.flyout).css({left:x});
52 };
53
54 /*p.setY = function(y) {
55     this.y = y;
56
57     // convert to percent
58     var elem = this.grid.getFirstGrid();
59     var rect =elem.get(0).↪
60         ↪ getBoundingClientRect();
61     var pY = y*100/rect.height;
62
63     $(".horizontalHair").first().css({↪
64         ↪ top:pY+"%"});
65 };*/
66
67 Flyout.show = function() {
68     Flyout.getContainer().className = "↪
69         ↪ chibiPoint_flyoutContainer↪
70         ↪ chibiPoint_shownfC";
71     //this.hairs.className = "crosshairs ↪
72         ↪ shownCrosshairs";
73 };
74
75 Flyout.hide = function() {
76     Flyout.getContainer().className = "↪
77         ↪ chibiPoint_flyoutContainer↪
78         ↪ chibiPoint_hiddenfC";
79     //this.hairs.className = "crosshairs ↪

```

```

72     ↪ hiddenCrosshairs";
73 };
74 p.hide = function() {
75     $(this.aFlyout).addClass("chibiPoi↪
76     ↪ nt_aFlyoutHidden");
77     //$ (this.aFlyout).removeClass("bir↪
78     ↪ chAFlyoutShown");
79 };
80 p.show = function() {
81     $(this.aFlyout).removeClass("chibi↪
82     ↪ Point_aFlyoutHidden");
83 };
84 p.build = function(label) {
85     var container = Flyout.getContainer(↪
86     ↪ this.root);
87
88     this.aFlyout = document.createElement↪
89     ↪ ('div');
90     this.aFlyout.className = "↪
91     ↪ chibiPoint_aFlyout";
92
93     this.lineDiv = document.createElement↪
94     ↪ ('div');
95     this.lineDiv.className = "↪
96     ↪ chibiPoint_lineDiv";
97     this.svg = document.createElement('↪
98     ↪ svg');

```

```

92     this.svg.className = "↪
93     ↪ chibiPoint_lineSvg"
94     this.svgLine = document.createElement↪
95     ↪ ('line');
96     this.svgLine.className = "↪
97     ↪ chibiPoint_line↪
98     ↪ chibiPoint_lineUnique"+↪
99     ↪ this.unique;
100     this.svgLine.id = "↪
101     ↪ chibiPoint_flyoutLine"+↪
102     ↪ this.unique;
103     //svgLine.style="stroke:rgb(255,0,0);↪
104     ↪ stroke-width:2";
105     $(this.svgLine).attr({x1:"0",
106     ↪ y1:"0",
107     ↪ x2:"200",
108     ↪ y2:"200"});
109     //$ (svgLine).attr({"data-bind": "↪
110     ↪ attr: {x1:x1,y1:y1,x2:x2,y2:y2↪
111     ↪ }"});
112     $(this.svg).append(this.svgLine);
113     $(this.lineDiv).append(this.svg);
114     this.flyout = document.createElement(↪
115     ↪ 'div');
116     this.flyout.className = "↪
117     ↪ chibiPoint_flyout↪
118     ↪ chibiPoint_flyoutUnique"+↪

```

```

109      ↪ this.unique;
      this.flyout.innerHTML = String.fr ↪
      ↪ omCharCode(label).toUpperCase() ↪
      ↪ ;
110
111      $(this.aFlyout).append(this.flyout);
112      $(this.aFlyout).append(this.lineDiv);
113
114      $(container).append(this.aFlyout);
115
116      // refresh namespace
117      $(this.lineDiv).html($(this.lineDiv) ↪
      ↪ .html());
118  };
119
120  /*p.updatePosition = function() {
121      ↪ this.grid.findPointerCoords();
122      var centerX = coords.centerX;
123      var centerY = coords.centerY;
124
125      this.setX(centerX);
126      this.setY(centerY);
127  };*/
128
129  // root required only if it's thefirst ↪
130      ↪ and you haven't done a ↪
131      ↪ makeContainer yet
132  p.initialize = function(label, root) {
133      this.root = root;

```

```

      this.unique = Flyout.getNextId();
      this.build(label);
      this.setX(this.unique*20);
      this.setY(this.unique*20);

      //this.updatePosition();
      //this.show();
  };

  p.point = function(avoidCoords) {
  if (this.target) {
      var rect = ↪
      ↪ this.target.getBoundingClientRect ↪
      ↪ ();
      var left = rect.left;
      var top = rect.top;
      var width = rect.width;
      var height = rect.height;

      var me = $(Flyout.getContainer());

      // adapted from within.js
      var myRect = me.get(0). ↪
      ↪ getBoundingClientRect();
      var res = !( (myRect.top > top+height) ↪
      ↪ || (myRect.top+myRect.height < ↪
      ↪ top) || (myRect.left > left+width ↪
      ↪ ) || (myRect.left+myRect.width < ↪
      ↪ left));

```

```

155     if(res) {
156         this.show();
157         // disperse away from grid, ↵
158             ↵ assuming it exists
159         var xdist = 0;
160         var ydist = 0;
161         var dist = 0;
162         var angle = 0;
163         if (avoidCoords) {
164             xdist = avoidCoords.centerX- ↵
165                 ↵ rect.left;
166             ydist = avoidCoords.centerY- ↵
167                 ↵ rect.top;
168             dist = Math.sqrt(xdist*xdist+ydist ↵
169                 ↵ *ydist);
170             angle = Math.atan2(ydist, xdist);
171         }
172         var avoidDist = Math.min((1/(dist* ↵
173             ↵ dist*dist))*1000000000, 60);
174
175         this.setX((rect.left-20)-Math.sin( ↵
176             ↵ angle)*avoidDist);
177         this.setY((rect.top-20)+Math.cos( ↵
178             ↵ angle)*avoidDist);
179         var theX = left;
180         var theY = top;
181
182         $("#+this.svgLine.id).attr({x1: ↵
183             ↵ this.x,
184         y1:this.y,

```

```

177     x2:theX,
178     y2:theY});
179     this.paintTarget();
180     } else {
181         this.hide();
182         this.unpaintTarget();
183     }
184     } else {
185         this.hide();
186     }
187 };
188
189 p.clickOrFocus = function(element) {
190     evaler.clickOrFocus(element);
191 }
192
193 p.doClick = function() {
194     if (this.target) {
195
196         var element = $(this.target);
197         $(this.target).addClass("chibiPoin ↵
198             ↵ t_cluck");
199
200         var flash = setInterval(function() {
201             if (element.hasClass("chibiPoint ↵
202                 ↵ _cluck")) {
203                 element.removeClass("chibiPoin ↵
204                     ↵ t_cluck");
205             } else {

```

```

204 element.addClass("chibiPoint_cluck");
205     }
206     }, 100);
207
208     var toClick = this.clickOrFocus;
209     var thatTarget = this.target;
210     var delayUnclick = setInterval(↵
211         ↵ function () {
212             clearInterval(flash);
213             clearInterval(delayUnclick);
214             $(thatTarget).removeClass("chibiP
215                 ↵ iPoint_cluck");
216             toClick(thatTarget);
217         }, 500);
218     }
219     };
220
221     p.paintTarget = function() {
222         if (this.target) {
223             $(this.target).addClass("chibiPoin
224                 ↵ t_painted");
225             $(this.target).addClass("chibiPoin
226                 ↵ t_paintedUnique"+this.unique);
227         }
228     };
229
230     p.unpaintTarget = function() {
231         if (this.target) {
232             $(this.target).removeClass("chibiP
233                 ↵ oint_painted");
234         }
235     };

```

```

229     $(this.target).removeClass("chibiP
230         ↵ oint_paintedUnique"+this.unique)↵
231         ↵ ;
232     }
233 };
234
235 p.unsetTarget = function() {
236     this.unpaintTarget();
237     this.target = false;
238 };
239
240 p.getTarget = function() {
241     return this.target;
242 };
243
244 p.setTarget = function(element,↵
245     ↵ coordsyst) {
246     /*if (element == null) {
247         $("#"+this.svgLine.id).attr({x2:0,
248             y2:0});
249         return;
250     }*/
251     this.unsetTarget();
252     this.target = element;
253     //this.point();
254
255     /*var rect2 =↵
256         ↵ coordsyst.getBoundingClientRect()↵
257         ↵ ;
258     var pX = theX*100/rect2.width;

```

62

```

254   var pY = theY*100/rect2.height;*/
255
256   /*$(this.svgLine).attr({x2:theX,
257                           y2:theY});
258   $(this.svgLine).attr({id:"hey"});
259   //$(this.lineDiv).html($(this.lin
      ↪ eDiv).html());
260   $(this.lineDiv).html($(this.lineDiv).
      ↪ html());*/
261   };
262
263   birchlabs.Flyout = Flyout;
264   })();
265
266   });

```

0.17 ./src/flyoutworker.js

```

1   onmessage = function (oEvent) {
2       console.log("Called back by theworker
      ↪ !\n");
3   };

```

0.18 ./src/evaluator.js

```

1   define(["lib/jquery-2.1.0.min", "lib/
      ↪ FileSaver", "trace"], function(jq,
      ↪ saver, tracer) {

```

```

2   window.evaluator = window.evaluator
      ↪ ||{};
3   var evaluator = window.evaluator;
4
5   window.birchlabs = window.birchlabs
      ↪ ||{};
6   var birchlabs = window.birchlabs;
7
8   function makeEvaluators() {
9       var keyCountElem = makeKeyCount(d
      ↪ ocument.documentElement);
10      var timerElem = makeTimer(
      ↪ document.documentElement);
11
12      var keys = "";
13      var times = "";
14      var timeDeltas = "";
15      var timesHuman = "";
16      var timeDeltasHuman = "";
17      var keyCount = 0;
18      var timer = new Date().getTime();
19      var totalTime = 0;
20      // this timer is running before any
      ↪ keypresses are used
21      var trueBegin = new Date().getTime();
22      var beginDelta = 0;
23      var wholeDuration = 0;
24
25      evaluator.keys = keys;
26      evaluator.times = times;

```

```

27 evaluator.timeDeltas = timeDeltas;
28 evaluator.timesHuman = timesHuman;
29 evaluator.timeDeltasHuman = ↵
    ↵ timeDeltasHuman;
30 evaluator.keyCount = keyCount;
31 evaluator.timer = timer;
32 evaluator.trueBegin = trueBegin;
33 evaluator.beginDelta = beginDelta;
34 evaluator.wholeDuration = ↵
    ↵ wholeDuration;
35 evaluator.totalTime = totalTime;
36
37 evaluator.keyCountElem = ↵
    ↵ keyCountElem;
38 evaluator.timerElem = timerElem;
39
40 birchlabs.downloadTelemetry = false;
41
42 drawEvaluators();
43 }
44
45 function clickOrFocus(element) {
46     var justFocus = {/"A":true,
47                     "INPUT":true,
48     "TEXTAREA":true,
49                     "SELECT":true,
50     "BUTTON":true//,
51     //"output":true,
52     //"command":true,
53                     //"kbd":true

```

```

54 };
55
56 if (justFocus[element.nodeName]) {
57     element.focus();
58     checkNewFocus(element, ↵
        ↵ window.location);
59 } else {
60     evaluatorWriteState(element);
61     element.click();
62     element.focus();
63 }
64 }
65
66 function checkFocusAfterWait() {
67     focusCheckTimer = setInterval(↵
        ↵ function () {
68         clearInterval(focusCheckTimer);
69         checkNewFocus(document.activeElement, ↵
            ↵ window.location);
70     }, 0);
71 }
72
73 function checkNewFocus(element, ↵
    ↵ location) {
74     tracer.trace(element);
75     tracer.trace(location.host);
76     endTest = false;
77
78     if (location.host == "www.↵
        ↵ halifax-online.co.uk") {

```

```

79     if (element.id == "↵ 101
        ↵ frmLogin:strCustomerLogin_pwd↵ 102
        ↵ ") {
80         endTest = true; 103
81     }
82 } else if (location.host == "↵ 104
        ↵ www.kickstarter.com") { 105
83     if (element.id == "↵ 106
        ↵ user_password_confirmation") ↵ 107
        ↵ { 108
84         endTest = true; 109
85     } 110
86 } 111
87 112
88 if (endTest) { 113
89     evaluatorWriteState(element); 114
90     alert("Success! Please move onto ↵ 115
        ↵ the next test."); 116
91 } 117
92 } 118
93 119
94 function makeTimer(root) { 120
95     var parent =document.createElement('↵ 121
        ↵ div'); 122
96     parent.className = 'chibiPoint_timer'↵ 123
        ↵ ; 124
97     $(root).append(parent);
98     return parent;
99 }
100

```

```

function makeKeyCount(root) {
    var parent =document.createElement('↵
        ↵ div');
    parent.className = '↵
        ↵ chibiPoint_keyCount';
    $(root).append(parent);
    return parent;
}

function evaluatorIncrementKeys(theKey)↵
    ↵ {

    var evaluator = window.evaluator;

    var nowTime = (new Date().getTime());
    if (evaluator.keyCount==0) {
        var beginDelta = nowTime-↵
            ↵ evaluator.trueBegin;
        evaluator.beginDelta = beginDelta;
        // time from the first keypress
        evaluator.timer = nowTime;
    }

    var thisDelta = nowTime-(evaluator↵
        ↵ r.timer+evaluator.totalTime);
    var wholeDelta = nowTime-↵
        ↵ evaluator.timer;

    var nowDate = new Date(nowTime);
    var nowDateString =(↵

```



```

    ↪ nowDate.toLocaleDateString() + ↪ 145
    ↪ " " + nowDate.toLocaleTimeString ↪
    ↪ ());
125
126     evaluator.keyCount++;
127     evaluator.keys += theKey+", ";
128     evaluator.times += nowTime+", ";
129     evaluator.timeDeltas += thisDelta+", " ↪ 150
    ↪ ;
130     evaluator.timesHuman += nowDateString ↪ 152
    ↪ +", ";
131     evaluator.timeDeltasHuman += ↪ 154
    ↪ formatTime(thisDelta)+", ";
132     evaluator.totalTime = wholeDelta;
133     evaluator.wholeDuration = wholeDelta+ ↪ 156
    ↪ evaluator.beginDelta;
134
135     //evaluator.keyCountElem.innerHTML = ↪ 158
    ↪ keyCount;
136     //evaluator.timerElem.innerHTML = ↪ 159
    ↪ timer;
137
138     drawEvaluators();
139 }
140
141 function drawEvaluators() {
142     var evaluator = window.evaluator;
143
144     evaluator.keyCountElem.innerHTML= ↪ 167
    ↪ evaluator.keys;
    168

```

```

    evaluator.timerElem.innerHTML = ↪
    ↪ formatTime(evaluator.totalTime) ↪
    ↪ ;
}

function formatTime(time) {
    var ret = time % 1000 + ' ms';
    time = Math.floor(time / 1000);
    if (time !== 0) {
        ret = time % 60 + "s "+ret;
        time = Math.floor(time / 60);
        if (time !== 0) {
            ret = time % 60 + "min"+ ↪
            ↪ ret;
            time = Math.floor(time/ ↪
            ↪ 60);
            if (time !== 0) {
                ret = time % 60 + "h "+ ↪
                ↪ ret;
            }
        }
    }
    return ret;
}

function evaluatorWriteState( ↪
    ↪ clickedElem) {
    //var b = blob.Blob;
    //console.log(b);

```

```

169
170     var endMilli = parseInt(evaluator.↵
        ↵ timer)+parseInt(↵
        ↵ evaluator.totalTime);
171
172     var startDate = new Date(↵
        ↵ evaluator.timer);
173     var endDate = new Date(endMilli);
174     var trueBeginDate = new Date(↵
        ↵ evaluator.trueBegin);
175
176     var output = "page: "+document.URL;
177     output += "\n clicked: "+clickedElem;
178     output += "\n startTime: "+↵
        ↵ evaluator.timer;
179     output += "\n endTime: "+endMilli;
180     output += "\n duration: "+↵
        ↵ evaluator.totalTime;
181     output += "\n durationHuman: "+↵
        ↵ formatTime(evaluator.totalTime)↵
        ↵ ;
182     // includes think time beforefirst ↵
        ↵ keypress
183     output += "\n\n trueBegin: "+↵
        ↵ evaluator.trueBegin;
184     output += "\n trueBeginDate: "+(↵
        ↵ trueBeginDate.toLocaleDateString↵
        ↵ ())+ " " +↵
        ↵ trueBeginDate.toLocaleTimeString↵
        ↵ ());

```

```

185     output += "\n thinkDuration: "+↵
        ↵ evaluator.beginDelta;
186     output += "\n thinkDurationHuman: "+↵
        ↵ formatTime(evaluator.beginDelta↵
        ↵ );
187     output += "\n wholeDuration: "+↵
        ↵ evaluator.wholeDuration;
188     output += "\n wholeDurationHuman: "+↵
        ↵ formatTime(↵
        ↵ evaluator.wholeDuration);
189     output += "\n\n startDate: "+(↵
        ↵ startDate.toLocaleDateString() ↵
        ↵ + " " + ↵
        ↵ startDate.toLocaleTimeString())↵
        ↵ ;
190     output += "\n endDate: "+(↵
        ↵ endDate.toLocaleDateString() + ↵
        ↵ " " + endDate.toLocaleTimeString↵
        ↵ ());
191     output += "\n\n times: "+↵
        ↵ evaluator.times;
192     output += "\n timesHuman: "+↵
        ↵ evaluator.timesHuman;
193     output += "\n\n timeDeltas: "+↵
        ↵ evaluator.timeDeltas;
194     output += "\n timeDeltasHuman: "+↵
        ↵ evaluator.timeDeltasHuman;
195     output += "\n\n keyCount: "+↵
        ↵ evaluator.keyCount;
196     output += "\n keys: "+evaluator.keys;

```

```

197     output += "\n\n CHIBIPOINT ON:"+(!↵
198         ↵ birchlabs.evaluateTabMode);
199     output += "\n FLYOUTS ON:"+↵
200         ↵ birchlabs.flyoutsOn;
201     var finalElem = "\n\nClicked.. \n";
202     for (var j in clickedElem) {
203         finalElem += " " + j + ": ↵
204         ↵ " + clickedElem[j] ↵
205         ↵ + "\n";
206     }
207     output += finalElem;
208     var bb = new Blob([output]);
209     //console.log(bb);
210     //bb.append((new XMLSerializer).s↵
211     ↵ erializeToString(document));
212     //var blob = bb.getBlob("application↵
213     ↵ /xhtml+xml;charset=" +↵
214     ↵ document.characterSet);
215     if (birchlabs.downloadTelemetry) {
216         saver.saveAs(bb, "document.txt");
217     }
218     return {makeEvaluators:makeEvaluators,
219         clickOrFocus:clickOrFocus,
220         evaluatorIncrementKeys:evaluatorIncrementKeys,
221         evaluatorWriteState:evaluatorWriteState,
222         WriteState,
223         checkNewFocus:checkNewFocus,
224         checkFocusAfterWait:checkFocusAfterWait,
225         sAfterWait};
226 }

```

0.19 ./src/lib/Blob.js

```

1  // err, actually we don't need this, do ↵
2  ↵ we? WebKit has its ownBlobBuilder. ↵
3  ↵ So we will not'require.js' it.
4
5  define([], function() {
6
7      /* Blob.js
8       * A Blob implementation.
9       * 2013-12-27
10
11      * By Eli Grey, http://eligrey.com
12      * By Devin Samarin, https://github.com/↵
13      ↵ eboyjr
14      * License: X11/MIT
15      * See LICENSE.md
16
17      */
18
19      /*global self, unescape */
20      /*jslint bitwise: true, regexp: true, ↵
21      ↵ confusion: true, es5: true, vars: ↵

```

```

17     ↪ true, white: true,
18     plusplus: true */
19     /*! @source http://purl.eligrey.com/g↪
20     ↪ ithub/Blob.js/blob/master/Blob.js ↪
21     ↪ */
22     /*
23     [PRINT VERSION]: This is an external↪
24     ↪ library, rather than our own work. ↪
25     ↪ Toreduce print
26     burden, we have redacted all but the↪
27     ↪ non-original file contents. However↪
28     ↪ these can be
29     slotted into place by grabbing those↪
30     ↪ contents from the above URL.
31     */
32     return {Blob: Blob};
33 }));

```

0.20 ./src/lib/FileSaver.js

```

1  define([], function() {
2
3  /*! FileSaver.js
4  *   A saveAs() FileSaver implementation.
5  *   2014-01-24
6  *
7  *   By Eli Grey, http://eligrey.com

```

```

8  *   License: X11/MIT
9  *   See LICENSE.md
10  */
11
12  /*global self */
13  /*jslint bitwise: true, indent: 4,↪
14  ↪ laxbreak: true, laxcomma: true,↪
15  ↪ smarttabs: true, plusplus: true */
16
17  /*! @source http://purl.eligrey.com/g↪
18  ↪ ithub/FileSaver.js/blob/master/↪
19  ↪ FileSaver.js */
20
21  /*
22  [PRINT VERSION]: This is an external↪
23  ↪ library, rather than our own work. ↪
24  ↪ Toreduce print
25  burden, we have redacted all but the↪
26  ↪ non-original file contents. However↪
27  ↪ these can be
28  slotted into place by grabbing those↪
29  ↪ contents from the above URL.
30  */
31
32  return {saveAs: saveAs};
33 }));

```

0.21 ./src/lib/jquery-2.1.0.min.js

```

1  /*! jQuery v2.1.0 | (c) 2005, 2014 jQuery ↵
   ↵ Foundation, Inc. | jquery.org/ ↵
   ↵ license */
2  /*! @source http://code.jquery.com/jq ↵
   ↵ uery-2.1.0.min.js */
3  /*
4  [PRINT VERSION]: This is an external ↵
   ↵ library, rather than our own work. ↵
   ↵ Toreduce print
5  burden, we have redacted all but the ↵
   ↵ non-original file contents. However ↵
   ↵ these can be
6  slotted into place by grabbing those ↵
   ↵ contents from the above URL.
7  */

```

0.22 ./src/lib/require-cs.js

```

1  // from nonowarn's example ContentScript ↵
   ↵ with RequireJS
2  // https://github.com/nonowarn/ ↵
   ↵ content-script-with-requirejs
3  // no license provided
4
5  /*! @source https://github.com/nonowarn/ ↵
   ↵ content-script-with-requirejs/blob/ ↵
   ↵ master/lib/require-cs.js */
6  /*
7  [PRINT VERSION]: This is an external ↵
   ↵ library, rather than our own work. ↵

```

```

   ↵ Toreduce print
8  burden, we have redacted all but the ↵
   ↵ non-original file contents. However ↵
   ↵ these can be
9  slotted into place by grabbing those ↵
   ↵ contents from the above URL.
10 */

```

0.23 ./src/lib/require.js

```

1  /*
2  RequireJS 2.1.10 Copyright (c)2010-2014 ↵
   ↵ , The Dojo Foundation AllRights ↵
   ↵ Reserved.
3  Available via the MIT or new BSD ↵
   ↵ license.
4  see: http://github.com/jrburke/requirejs ↵
   ↵ for details
5  */
6  /*! @source http://requirejs.org/docs/ ↵
   ↵ release/2.1.10/minified/require.js ↵
   ↵ */
7  /*
8  [PRINT VERSION]: This is an external ↵
   ↵ library, rather than our own work. ↵
   ↵ Toreduce print
9  burden, we have redacted all but the ↵
   ↵ non-original file contents. However ↵
   ↵ these can be

```

```

10  slotted into place by grabbing those ↵
11  ↵ contents from the above URL.
    */

```

0.24 ./src/lib/within.js

```

1  define(["lib/jquery-2.1.0.min"], function ↵
    ↵ (jq) {
2    // from jquery++, MIT license
3    /*! @source https://github.com/ ↵
        ↵ jqueryjs/jquerymx/blob/ ↵
        ↵ master/dom/within/within.js ↵
        ↵ */
4    /*

```

```

5    [PRINT VERSION]: This is an ↵
        ↵ external library, rather ↵
        ↵ than our ownwork. To reduce ↵
        ↵ print
6    burden, we have redacted allbut ↵
        ↵ the non-original file ↵
        ↵ contents. However these can ↵
        ↵ be
7    slotted into place bygrabbing ↵
        ↵ those contents from the ↵
        ↵ above URL.
8    */
9
10   });

```