buttonRow

10u

(width-2*10u)/count   (width-2*10u)/count

| swim | cruise | **row** | fish | dive |

50u   40u

440u

swim   cruise   row   fish   dive

swim   cruise   **row**   fish   dive
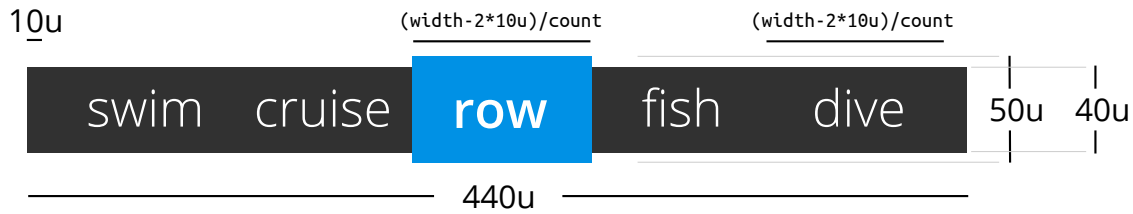
the user can drag the active rectangle and drop it over the desired value. The drag is constrained to the X axis, and the active rectangle snaps back to the nearest full option (does **not** stay midway between options)

swim   cruise   **row**   fish   dive

If the user clicks to an option other than the current, the rectangle animates to the new value with OutCubic easing. The new value becomes bold when the animation is finished.

proposal for the adventurous hacker

It will be more visually pleasing, and the row will fit more items if we make spaces between options equal, rather than option widths equal.

swim	cruise	row	**fish**	dive

vs

swim	cruise	row	**fish**	dive

bristlemouth	**dab**

vs

bristlemouth	**dab**	cod	eel

battleship	**tanker**	dinghy	barge

d

```
for (var item in buttonRow) tw+=item.width
d = tanker.width+(width-tw-2*10u)/count
```

when the highlight is between active items, it's width d is calculated as follows

```
for (var item in buttonRow) tw+=item.width
d = tanker.width+(width-tw-2*10u)/count
```

battleship	**tanker**	dinghy	barge

d_new   d   d_old

```
for (var item in buttonRow) tw+=item.width
d = tanker.width+(width-tw-2*10u)/count
```

when the highlight is between active items, it's width d is calculated as the weighted average of d_new and d_old where the weight is the proportion of the distances of the center of the highlight from the old and new center.

```
factor = distance(center, center_old)/distance(center_old,center_new)
d = (d_new*factor+d_old*(1-factor))
```