

```

1  d3.TimeChart = function(){
2
3      var w = 800,
4          h = 600,
5          m = {t:0,r:0,b:0,l:0},
6          chartW = w - m.l - m.r,
7          chartH = h - m.t - m.b,
8          scaleX = d3.time.scale(),
9          scaleY = d3.scale.linear(),
10         timeRange = [new Date(), new Date()],
11         interval = d3.time.day,
12         layout = d3.layout.histogram(),
13         valueAccessor = function(d){ return d;};
14
15     var exports = function(_selection){
16         //recalculate width, height, scales, layout
17         chartW = w - m.l - m.r;
18         chartH = h - m.t - m.b;
19         scaleX = d3.time.scale().domain(timeRange).range([0,chartW]);
20         scaleY = d3.scale.linear().range([chartH,0]);
21
22         var bins = interval.range(timeRange[0],timeRange[1]);
23         bins.push(timeRange[1]);
24         layout
25             .range(timeRange)
26             .bins(bins)
27             .value(valueAccessor);
28
29         _selection.each(draw);
30     }
31
32     function draw(array){
33         var data = layout(array),
34             maxY = d3.max(data, function(d){return d.y})*1.1;
35         scaleY.domain([0, maxY]);
36
37         var svg = d3.select(this).selectAll('svg').data([array]);
38
39         //when DOM is drawn for the first time
40         var svgEnter = svg.enter().append('svg').attr('width',w).attr('height',h);
41         svgEnter.append('g').attr('transform','translate('+ m.l+' '+ m.t)').attr('class','
chart');
42         svgEnter.append('g').attr('transform','translate('+ m.l+' '+ m.t)').attr('class','
brush');
43
44         //bars
45         var bars = svg.select('.chart')
46             .selectAll('.bar')
47             .data( layout(array), function(d,i){return i});
48
49         bars.enter().append('rect').attr('class','bar');
50         bars.exit().remove();
51
52         bars.transition()
53             .attr('x',function(d){return scaleX(d.x)})
54             .attr('y',function(d){return scaleY(d.y)})
55             .attr('height',function(d){return chartH - scaleY(d.y)})
56             .attr('width',function(d){
57                 var time2 = d.x.getTime() + d.dx;
58                 return scaleX(time2) - scaleX(d.x);

```

```
59     });
60   }
61
62   exports.width = function(_x){
63     if(!arguments.length) return w;
64     w = _x;
65     return this;
66   }
67   exports.height = function(_x){
68     if(!arguments.length) return h;
69     h = _x;
70     return this;
71   }
72   exports.range = function(_r){
73     //type of _r --> [a, b] where a and b are Date objects
74     if(!arguments.length) return timeRange;
75     timeRange = _r;
76     return this;
77   }
78   exports.interval = function(_i){
79     //type of _i --> d3.time object
80     if(!arguments.length) return interval;
81     interval = _i;
82     return this;
83   }
84   exports.value = function(_v){
85     //type of _v --> value accessor function
86     if(!arguments.length) return valueAccessor;
87     valueAccessor = _v;
88     return this;
89   }
90   exports.margin = function(_m){
91     if(!arguments.length) return [m.t, m.r, m.b, m.l];
92     m.t = _m[0];m.r = _m[1];m.b = _m[2];m.l = _m[3];
93     return this;
94   }
95
96   return exports;
97 }
```