



链滴

绘画板 05——设置图形边框颜色与背景色 (喷枪)

作者: [crick77](#)

原文链接: <https://ld246.com/article/1474601036362>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

github地址: <https://github.com/wangyuheng/painter>

DEMO地址: <http://painter.crick.wang/>

边框与背景色

首先, 使用了开源的 [spectrum.js](#) 作为颜色选择器,

之前以SVG对象保存全局方法与对象, 现在通过新定义一个GlobalStatus, 存在全局属性, 并通过按状态判断状态

```
var GlobalStatus = {
  defaultFontColor:"#0073c6",
  defaultFillColor:"#ecec",
  defaultLineSize:"2",
  isPicked:function(){
    return $("#tool_pick").hasClass("active");
  },
  isPreFilled:function(){
    return $("#tool_fill").hasClass("active");
  },
  getFontColor:function(){
    return $("#font_color").attr("data-color");
  },
  getFillColor:function(){
    return $("#fill_color").attr("data-color");
  },
  getLineSize:function(){
    return this.defaultLineSize;
  },
  getFillOpacity:function(){
    var fillOpacity = "0.0";
    if (this.getFillColor() != this.defaultFillColor) {
      fillOpacity = "1.0";
    }
    return fillOpacity;
  }
}
```

在首页给默认线条颜色及背景颜色赋值, 并设置状态data-color保存颜色值

```
$("#font_color span").css("background", GlobalStatus.defaultFontColor);
$("#font_color").attr("data-color", GlobalStatus.defaultFontColor);
$("#fill_color span").css("background", GlobalStatus.defaultFillColor);
$("#fill_color").attr("data-color", GlobalStatus.defaultFillColor);
```

颜色选择器发生变化时, 同时变更颜色与data-color属性值

```
$("#colorPalettes").spectrum({
  flat: true,
  showPaletteOnly: true,
```

```

color: '#0073c6',
move: function(tinycolor) {
  console.log("color move");
  console.log(tinycolor.toHexString())
  if ($("#fill_color").hasClass("active")) {
    $("#fill_color").attr("data-color", tinycolor.toHexString());
    $("#fill_color span").eq(0).css("background", tinycolor.toHexString());
  } else if ($("#font_color").hasClass("active")) {
    $("#font_color").attr("data-color", tinycolor.toHexString());
    $("#font_color span").eq(0).css("background", tinycolor.toHexString());
  }
},
palette: spectrumColors
});

```

这里有一个取巧的地方，为了让背景颜色透明显示，规定ecec为透明颜色。选择此颜色时，将透明opacity设置为全透明。

这样做有一个问题，如果用户自定义此颜色，可以将自定义的颜色修改为eeced，不会在视觉上有较影响。

在绘制元素时，通过GlobalStatus设置元素的边框、颜色、及透明度。已rect为例

```

function mousedown(event) {
  console.log('rect mousedown');
  drawing = true;
  startPoint = svgDoc.transformPoint(event);
  element = parent.rect(0, 0).fill(GlobalStatus.getFillColor()).style("fill-opacity", GlobalStatus.g
tFillOpacity()).stroke({
  width: GlobalStatus.getLineSize(),
  color: GlobalStatus.getFontColor()
});
}

```

此时，绘制的颜色就会根据设置的颜色及背景色发生变化。

喷枪

在GlobalStatus中增加了isPreFilled()方法，判断喷枪是否被选中。在Element的扩展方法中，监听click事件。

如果为选中喷枪的点击，则修改元素的对应属性值即可。

```

_ele.on("click", function() {
  if (GlobalStatus.isPreFilled()) {
    if ($("#fill_color").hasClass("active")) {
      _ele.fill(GlobalStatus.getFillColor());
      _ele.style("fill-opacity", GlobalStatus.getFillOpacity());
    } else {
      _ele.style("stroke", GlobalStatus.getFontColor());
    }
  }
});

```

```
    }  
  } else if (GlobalStatus.isPicked()) {  
    if (!_ele.attr("picked")) {  
      _ele.attr("picked", true);  
      console.log(_ele.handleBorder);  
      _ele.handleBorder = _ele.handleBorder || new HandleBorder(svgDoc);  
      _ele.handleBorder.showShade(_ele);  
      pickedElementList.push(_ele);  
    } else {  
      _ele.attr("picked", null);  
      _ele.handleBorder && _ele.handleBorder.hideShade(_ele);  
      pickedElementList.remove(_ele);  
    }  
  }  
});
```

这里有一个用户体验的问题，喷枪是同时修改边框色和背景色，还是分别修改？

现在的实现方法为分别修改。用户需要点击线条与背景色，然后再用喷枪工具分别修改元素颜色。