

轮播组件

轮播组件的配置需要在高级一》表达式，下面去配置：

如下为一个表达式的范例：

```
expression = {
  "on" : {
    "init" : function(event){
      var temps = [];
      for(var i=0; i<10; i++){
        temps.push({
          label : "指标" + i,
          value : parseInt(Math.random() * 100),
          icon : "proudsmart ps-co2",
          status : parseInt(Math.random() * 4)
        })
      }
      event.target.render(temps);
    }
  },
  "config" : {
    slidesToShow : 6,
    slidesToScroll : 6
  },
  "render" : function(event){
    var wrap = $("<div></div>");
    var p = $("<p></p>");
    var p2 = $("<p></p>");
    var dom = $("<div></div>");
    var i = $("<i></i>");
    p.css("text-align", "center");
    p2.css("text-align", "center");
    p.text(event.item.value);
    p2.text(event.item.label);
    i.addClass(event.item.icon);
    i.css("font-size", "48px");
    dom.addClass("btn btn-primary");
    var status = event.item.status;
    dom.css("background-color", event.target.getAlertColor(status));
    dom.css("border", "0");
    dom.css("width", "74px");
    dom.css("height", "67px");
    dom.css("margin", "5px auto");
    dom.css("display", "block");
    dom.append(i);
    wrap.append(p);
    wrap.append(dom);
    wrap.append(p2);
    return wrap;
  }
}
```

其中

```
{
  "on" : {
    "init" : function(event){
    }
  }
}
```

为初始化的方法，在组件被加载时被执行。

```
var temps = [];
for(var i=0; i<10; i++){
  temps.push({
    label : "指标" + i,
    value : parseInt(Math.random() * 100),
    icon : "proudsmart ps-co2",
    status : parseInt(Math.random() * 4)
  })
}
```

```
event.target.render(temps);
```

这一段代码的作用是产生一个数组，数组中每一个对象代表一个重复单元的数据。

最下面的render方法返回一个js的dom对象。

```
"render" : function(event){
  var wrap = $("<div></div>");
  return wrap;
}
};
```

其中event对象包涵一个target对象，此对象对应之前temps数组中每一个单元的数据。