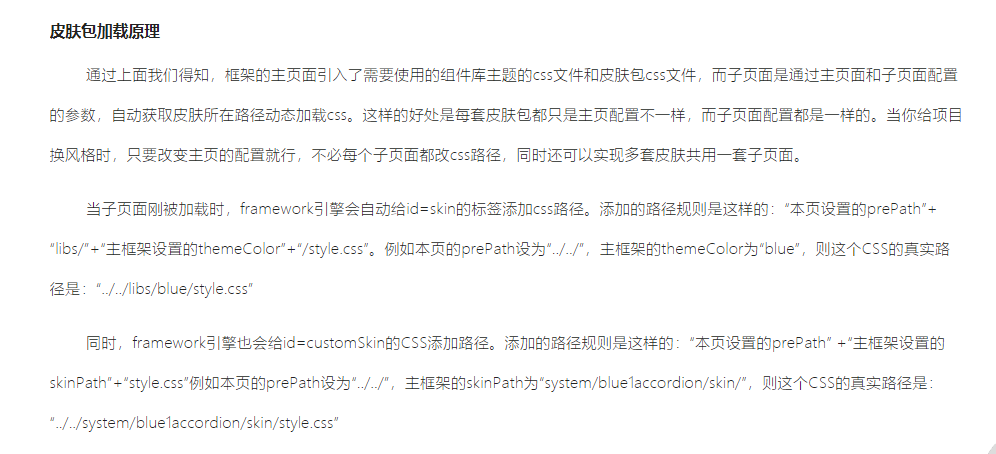
QiuckUI 使用的是XHTML严格型文档，必须加入该dtd声明。



QuickUI的换肤机制

主页面，配置即可





（允许用户换肤？）

https://blog.csdn.net/zzyly1/article/details/79505197

jQuery DataTables大数据异步二次加载渲染及initComplete事件bug

2018年03月10日 09:43:34 zzyhjm 阅读数 2570

我们在使用dataTables进行数据统计时，不可避免的会碰到对大数据的统计。当进行服务器端大数据读取时，毫无疑问的会占用大量加载时间，拖慢页面加载速度。为优化页面加载速度问题，我们便要在将请求中最耗时的部分在页面加载完成之后，进行二次加载，渲染入数据。

之前我采用的是重新发起dataTables渲染事件，来进行二次加载。如下：

<script type="javascript/text">

var datat = $('.dataTables-example').DataTable({

//初始化信息。。。 此处省略，请参看之前博文

});

datat.on('draw.dt',function() {

datat1.column(7, { //7是需要渲染的列

search: 'applied',

order: 'applied'

}).nodes().each(function(cell, i) {

$.get(url,{id:i},function(res){ //参数可以有其他获取方式

cell.innerHTML = res.data; //向单元格中写入数据

});

});

}).draw();

</script>

这种方法的缺点是会发起2次dataTables请求，影响加载速度。而利用dataTables自带的”DrawCallback”渲染事件参数。

<script type="javascript/text">

var datat = $('.dataTables-example').DataTable({

//其他初始化信息。。。

"drawCallback":function(s){

datat.column(7).nodes().each(function(cell,i){

$.get("\_\_URL\_\_/getClicknum",{id:i},function(res){

cell.innerHTML = res.data;

});

});

},

});

</script>

这样就可以避免重复请求，完成二次加载。

注意：不能使用”initComplete”加载完成事件，其在翻页时会失效。