**一、源起**

分析平台中，用来显示数据结果的结构（也可称为报告），此结构暂命名为：templet。

也可叫做分析报告结果模板。

**二、思路**

1-用json格式来描述模板，里面混杂html；

2-各种图的显示，在这里定义，但不在这里实现，定义包括，但不限于——

a)图的种类：饼图柱图等；

b)图的操作：钻取——这个本版本不处理

3-根据这个模板生成实际的显示内容。

4-这个模板是可以异步的，这里的异步指的是模板中用到的数据是异步的

5-模板要有结构，类似word的目录结构。

**三、流程图**

处理流程如下：



1-分析过程生成templet，这个文件属于TEMPLET-file范畴

2-analTempletJson总体框架一次性生成(或异步?)

**四、模板展现**

对templet的展现，可以有多种形式，目前看有两种：

a)html5，用于在界面展示——目前主要是PC

b)word，用于导出内容

将来可能还有

d)移动端展示——肯能与html5类似，或以此为基础

e)flash等

**注意：**实现展示时要考虑数据是否准备完成

**五、模板结构规定**

**1-数据格式规定**

1-带下划线的属性，是本标准的默认属性，其他属性是用户定义的，建议不带有下划线；

2-由\_HEAD属性、\_DATA属性和\_TEMPLET属性组成主体结构；

请参看样例。

**2-主体结构由解释**

**a)-“\_HEAD”：**

这个是头信息，包括如下内容：报告名称、模板编码、创建时间，描述。

***例子：***

|  |
| --- |
| "\_HEAD":{  "\_reportName":"test",  "\_code":"T.TEST::0002",  "\_cTime":"-480::1421744444796"  "\_desc":"测试模板"  } |

***解释：***

|  |  |  |  |
| --- | --- | --- | --- |
| **标签** | **分类** | **格式说明** | **补充** |
| \_id | **标准：必须** | 36位标准UUID  此编码是实体码，类似于类的实例 | 本期使用 |
| \_reportName | **标准：必须** | 报告名称 | 本期使用 |
| \_code | **标准：必须** | 报告，此编码分为大类编码：大写的以.隔开的名称，<::>后是小类型编号，采用数字，也可以是UUID  此编码是类型码，类似于类 | 本期使用 |
| \_cTime | **标准：必选** | 数据创建时间 | 本期使用 |
| \_desc | 标准：可选 | 数据信息的描述。 | 本期使用 |

**b)-“\_DLIST”：**

此模板需要的所有数据，是一个数组，数组中的元素为数据来源，都包括数据的编号、获取数据的url、若数据是jsonD格式，指明jsonD格式的编码。

***例子：***

|  |
| --- |
| "\_DLIST":[  {"\_id":"0", "\_ url":'getJsonD.do?datafile=/??/.jsond', \_ jsonDcode:'SP.TEAM-00007'},  {"\_id":"1"," \_url":'getJsonD.do?datafile=/??/anal\_quota\_716-446655440000.jsond', \_ jsonDcode:'SP.TEAM-00009'}  ] |

***数据来源标签解释：***

|  |  |  |  |
| --- | --- | --- | --- |
| **标签** | **分类** | **格式说明** | **补充** |
| \_id | **标准：必须** | 为数据编号，模板中的<d>标签会用到，只在本模板中使用的id，是整数，从0开始 | 本期使用 |
| \_url | **标准：必须** | 指明数据的来源，可以是一个服务过程的结果，也可以直接是一个json文件的地址 | 本期使用 |
| \_jsonDcode | 标准：可选 | jsonD格式的编码，若数据不是jsonD格式，也可以不设置此属性。  有此属性后， | 本期使用 |

**c)-“\_TEMPLET”：**

模板部分由树形结构组成，叶结点是seg(段)类型。

***例子：***

|  |
| --- |
| \_TEMPLET:[  {  id:"seg1",  name:"上传数据CCC",  title:'<div style="font-height:bold;font-size:24px;">上传数据￥￥￥￥</div>',  subSeg:[  {  id:"seg1\_1",  name:"结构分析",  content:'<div style="font-height:bold; font-size:18px;">1、结构分析</div><br/>  <div style="font-height:bold; ">“<d did="1" showType="value" value="excelMdmArray[0].sheetInfo.name"/>”页签(sheet<d did="1" showType="value" value="excelMdmArray[0].sheetInfo.index"/>)</div>为新增结构，元数据结构分析结果如下：<br/>  <d did="1" showType="table" value="excelMdmArray[0]"/><br/>  <div style="font-height:bold; ">“<d did="1" showType="value" value="excelMdmArray[1].sheetInfo.name"/>”页签(sheet<d did="1" showType="value" value="excelMdmArray[1].sheetInfo.index"/>)</div>为原有结构，元数据用原有结构。<br/>'  },{  id:"seg1\_2\_1",  name:"单项指标分析",  content:'<div style="font-height:bold; font-size:18px;">2、单项指标分析</div><br/>  <div style="font-height:bold; ">“<d did="2" showType="value" value="mdName"/>”[<d did="2" showType="value" value="quotas[0].titleName"/>]</div>指标分析：<br/>  “<d did="2" showType="value" value="quotas[0].titleName"/>”中  大多为{<d did="2" showType="value" value="quotas[0].tableData.tableBody[0].category"/>}占<d did="2" showType="value" value="quotas[0].tableData.tableBody[0].percent"/>%，  {<d did="2" showType="value" value="quotas[0].tableData.tableBody[1].category"/>}占<d did="2" showType="value" value="quotas[0].tableData.tableBody[1].percent"/>%，具体分析数据如下：<br/>  <d did="2" showType="table" value="quotas[0]" titleDcrt="quotas[0].titles"/>  <d did="2" showType="pie" value="quotas[0]" label="category", did="num" decorateView="{lableShow:[category, percent]}"/>  <br/>  <div style="font-height:bold; ">“<d did="2" showType="value" value="mdName"/>”[<d did="2" showType="value" value="quotas[1].titleName"/>]</div>指标分析：<br/>  新加测试showType=“line”：<br/><d did="2" showType="line" value="quotas[1]" label="category", did="num" decorateView="{lableShow:[category, percent]}" />  新加测试showType=“bar”：<d did="2" showType="bar" value="quotas[1]" label="category", did="num" decorateView="{lableShow:[category, percent]}" />  测试showType=“first”降序：<d did="2" showType="first( !(3 |num ) )" value="quotas[1]" decorateView="{#category#}占#percent#%"/>  测试showType=“first”升序：<d did="2" showType="first( 3 |num )" value="quotas[1]" decorateView="{#category#}占#percent#%"/>'  }  ]  }  ] |

**注意，Content中的内容html中的内容都用单引号。**

***seg(段)内元素标签解释：***

|  |  |  |  |
| --- | --- | --- | --- |
| **标签** | **分类** | **格式说明** | **补充** |
| id | **定义：必须** | 段id，注意此id仅作为区分标记，不带有任何树结构信息 | 本期使用 |
| name | **定义：可选** | 段名称，可用此生成树的名称 | 本期使用 |
| title | **定义：可选** | 段显示名称，可用此生成树的名称，也用此生成实际模板显示时的段标题。  **此属性内容是html代码** | 本期使用 |
| 以上两个属性必须要有一个被指定：  ·若只指定了name，则实际显示中，段标题也按照name设置；  ·若只指定了title，则实际显示中，树结点名称也按照title设置； | | | |
| content | 定义：可选 | 段内容，其中包括d标签，一般叶结点此标签都要设置内容  **此属性内容是html代码** | 本期使用 |
| subSeg | 定义：可选 | 下级段数组 | 本期使用 |

**3-<d>标签的详细解释**

用于显示数据内容的标签，与\_DATA中的数据配合，目前包括：

表格、图形(饼图、柱图、雷达图、折线图)、文字内容(简单取值，排名取值)

***例子：***

|  |
| --- |
| <d did=‘1’ showType=‘value’ value=‘excelMdmArray[0].sheetInfo.name’/>  <d did=‘1’ showType=‘table’ value=‘excelMdmArray[0]’/>  <d did=‘2’ showType=‘pie’ value=‘quotas[0]’ label=‘category’, data=‘num’ decorateView=‘{lableShow:[category, percent]}’/>  <d did=‘2‘ showType=‘bar’ value=‘quotas[1]’ label=‘category’, data =‘num’ decorateView=‘{lableShow:[category, percent]}’ />  <d did=‘2’ showType=‘line’ value=‘quotas[1]’ label=‘category’, data =‘num’ decorateView=‘{lableShow:[category, percent]}’ />  <d did=‘2’ showType=‘!first(3|num)’ value=‘quotas[1]’ decorateView=‘{#category#}占#percent#%’/>  <d did=‘2’ showType=‘first(3|num )’ value=‘quotas[1]’ decorateView=‘{#category#}占#percent#%’/> |

***解释：***

|  |  |  |  |
| --- | --- | --- | --- |
| **标签** | **分类** | **格式说明** | **补充** |
| did | **标准：必须** | 与\_DATA标签中的“数据来源”中的\_id相对应 | 本期使用 |
| showType | **标准：必须** | 数据显示形式 | 本期使用 |
| value | **标准：必须** | 获取的数据 | 本期使用 |
| lable | 标准：可选 | 只有饼、柱、折线图有意义，X轴信息 | 本期使用 |
| data | 标准：可选 | 只有饼、柱、折线图有意义，Y轴信息，一般是数值 | 本期使用 |
| decorateView | 标准：可选 | 对显示的修饰说明 | 本期使用 |

**注意：lable/data，若图进行90度翻转，怎样定义？？**

**a.1)-简单取值(value)：**

***例子：***

|  |
| --- |
| <d did=‘1’ showType=‘value’ value=‘excelMdmArray[0].sheetInfo.name’/> |

***解释：***

|  |  |
| --- | --- |
| did=1 | \_DLIST数据列表中\_id=1的jsonD数据 |
| showType=’value’ | 取值类型是“简单取值” |
| value=’excelMdmArray[0].sheetInfo.name’ | 取得jsonD中\_DATA对象中  excelMdmArray[0].sheetInfo.name的值  (excelMdmArray数组第0个记录sheetinfo对象中name属性的只) |

**a.2)-排名取值(first)：**

***例子：***

|  |
| --- |
| <d did=‘2’ showType=‘!first(3|num)’ value=‘quotas[1]’ decorateView=‘{#category#}占#percent#%’/>  <d did=‘2’ showType=‘first(3|num )’ value=‘quotas[1]’ decorateView=‘{#category#}占#percent#%’/> |

***解释：***

|  |  |
| --- | --- |
| did=2 | \_DLIST数据列表中\_id=2的jsonD数据 |
| showType=‘first(3|num)’ | 对数据中的num列，从大到小排序，取最后3列  相当于  对数据中的num列，从小到大排序，取前3列 |
| showType=‘!first(3|num)’ | 对数据中的num列，从大到小排序，取前3列 |
| value=‘quotas[1]’ | 取得jsonD中\_DATA对象中quotas[1]的数据表  (quotas数组第1个记录，此记录是表格数据) |
| decorateView  =‘{#category#}占#percent#%’ | 取得的数据按照此进行显示。  ##之间是列名，其他字符按照原样显示，如排序后的数据为：   |  | | --- | | {category:"北京", num:"42", percent:"21"},  {category:"四川", num:"8", percent:"4"},  {category:"其他", num:"2", percent:"1"}, |   **则显示为：**   |  | | --- | | {北京}占21%、{四川}占8%、{其他}占2% |   **注意：**  **1默认情况下每个显示之间用“、”分开(中文顿号)**  **2若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！** |

**问题：**

1-排序取值默认确定是从大到小排序的，请参考“对jsonD格式中表格数据的规定”

2-排序的定义还不是特别好

3-若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！

4-分割符没有进行定义

**b)-列表取值(table)：**

***例子：***

|  |
| --- |
| <d did=‘1’ showType=‘table’ value=‘excelMdmArray[0]’/> |

***解释：***

|  |  |
| --- | --- |
| did=1 | \_DLIST数据列表中\_id=1的jsonD数据 |
| showType=‘table’ | 显示为table |
| showType=‘!first(3|num) | 对数据中的num列，从大到小排序，取前3列 |
| value=‘excelMdmArray[0]’ | 取得jsonD中\_DATA对象中excelMdmArray[0]的数据表  (excelMdmArray数组第0个记录，此记录是表格数据) |

**c.1)-饼图(pie)：**

***例子：***

|  |
| --- |
| <d did=‘2’ showType=‘pie’ value=‘quotas[0]’ label=‘category’, data=‘num’ decorateView=‘{lableShow:[category, percent]}’/> |

***解释：***

|  |  |
| --- | --- |
| did=2 | \_DLIST数据列表中\_id=2的jsonD数据 |
| showType=‘pie’ | 按饼图进行显示 |
| value=‘quotas[0]’ | 取得jsonD中\_DATA对象中quotas[0]的数据表  (quotas数组第0个记录，此记录是表格数据) |
| lable=‘category’, | 横坐标/分类维度/标签显示  对应quotas[1]的数据表中dataList中category列，一般这样的列是离散的，表现为字符串或时间点，从意义上看是分类和时间 |
| data=‘num’, | 纵坐标/或数值维度  对应quotas[0]的数据表中dataList中num列，一般这样的列是数值的 |
| decorateView  =‘{lableShow:[category, percent]}’ | 怎样显示饼图中的标签。  显示表格中的”category”列，并显示百分比  **注意：**  **1若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！** |

**问题：**

1-decorateView定义过于复杂

2-若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！

**c.2)-柱图(bar)：**

***例子：***

|  |
| --- |
| <d did=‘2‘ showType=‘bar’ value=‘quotas[1]’ label=‘category’, data =‘num’ decorateView=‘{lableShow:[category, percent]}’ /> |

***解释：***

|  |  |
| --- | --- |
| did=2 | \_DLIST数据列表中\_id=2的jsonD数据 |
| showType=‘bar’ | 按饼图进行显示 |
| value=‘quotas[1]’ | 取得jsonD中\_DATA对象中quotas[1]的数据表  (quotas数组第1个记录，此记录是表格数据) |
| lable=‘category’ | 横坐标/分类维度/标签显示  对应quotas[1]的数据表中dataList中category列，一般这样的列是离散的，表现为字符串或时间点，从意义上看是分类和时间 |
| data=‘num’ | 纵坐标/或数值维度  对应quotas[1]的数据表中dataList中num列，一般这样的列是数值的 |
| decorateView  =‘{lableShow:[category, percent]}’ | 怎样显示饼图中的标签。  显示表格中的”category”列，并显示百分比  **注意：**  **1若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！** |

**问题：**

1-decorateView定义过于复杂

2-若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！

**c.3)-折线图(line)：**

***例子：***

|  |
| --- |
| <d did=‘2’ showType=‘line’ value=‘quotas[1]’ label=‘category’, data =‘num’ decorateView=‘{lableShow:[category, percent]}’ /> |

***解释：***

|  |  |
| --- | --- |
| did=2 | \_DLIST数据列表中\_id=2的jsonD数据 |
| showType=‘line’ | 按饼图进行显示 |
| value=‘quotas[1]’ | 取得jsonD中\_DATA对象中quotas[1]的数据表  (quotas数组第1个记录，此记录是表格数据) |
| lable=‘category’ | 横坐标/分类维度/标签显示  对应quotas[1]的数据表中dataList中category列，一般这样的列是离散的，表现为字符串或时间点，从意义上看是分类和时间 |
| data=‘num’ | 纵坐标/或数值维度  对应quotas[1]的数据表中dataList中num列，一般这样的列是数值的 |
| decorateView  =‘{lableShow:[category, percent]}’ | 怎样显示饼图中的标签。  显示表格中的”category”列，并显示百分比  **注意：**  **1若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！** |

**问题：**

1-decorateView定义过于复杂

2-若列表中没有百分比，则解析时自动计算百分比，也就是说percent字段是有二义性的！！

**c.4)-雷达图(radar)：**

***例子：***

|  |
| --- |
|  |

***解释：***

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

请马海涛补上！！！

**标签和不同类型的必选/可选对比：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **标签**  **showType** | **JsonD数据指向**  **Did** | **显示类型**  **showType** | **数据**  **value** | **标签/横坐标**  **lable** | **数值/纵坐标**  **Data** | **显示修饰**  **decorateView** |
| **a.1)value** | √ | √ | √ | × | × | 可选  (显未定义) |
| **a.2)first** | √ | √ | √ | × | × | √ |
| **b)table** | √ | √ | √ | × | × | 可选  (显未定义) |
| **c.1)pie** | √ | √ | √ | √ | √ | 可选 |
| **c.2)bar** | √ | √ | √ | √ | √ | 可选 |
| **c.3)line** | √ | √ | √ | √ | √ | 可选 |
| **c.4)radar** |  |  |  |  |  |  |

**注意：！！！**

1-目前的定义，不是很统一，可能还需要重新定义各标签；

2-目前只能定义一种图，如果要实现图的叠加，则不能实现，这个在以后的定义中要实现；

3-是否有这样的图，不是数值而是点，看密度分布，如地图？？？

**六、对JsonD格式中表格数据的规定**

为配合templet中对二维表的展示，规定——有表数据的jsonD必须按如下方式定义：

**例子：**

|  |
| --- |
| tableDataDemo:{  columnName:"col\_4",  titleName:"籍贯",  **tableData:{**  sort:{sortCol:"num",shotType:"1"},  titles:[{category:"籍贯"},{num:"数量"},{percent:"百分比"}],  **dataList:**[  {category:"其他", num:"2", percent:"1"},  {category:"北京", num:"42", percent:"21"},  {category:"四川", num:"8", percent:"4"},  {category:"江苏", num:"6", percent:"3"},  {category:"河北", num:"36", percent:"18"},  {category:"天津", num:"27", percent:"13.5"},  {category:"上海", num:"25", percent:"12.5"},  {category:"河南", num:"28", percent:"14"},  {category:"辽宁", num:"12", percent:"6"},  {category:"浙江", num:"8", percent:"4"},  {category:"黑龙江", num:"6", percent:"3"}  ]  }  } |

**说明：**

有表数据的内容应该是json中的对象，其中要包括tableData属性，此属性也是一个json对象，这个对象中的dataList属性是真正的二维表数据。

·titles——说明列名称的对象

·sort——说明列表排序的对象

**解释：**

|  |  |  |  |
| --- | --- | --- | --- |
| **标签** | **分类** | **格式说明** | **补充** |
| tableData | **标准：必须** | 所有二维表数据都要有这个属性 | 本期使用 |
| ┝dataList  │ | **标准：必须** | 实际的二维表数据，是一个json数组  是tableData的子属性 | 本期使用 |
| ┝titles  │  │ | **标准：必须** | 对列数据进行说明，目前还只有名称，后续可考虑把数据类型加入  是tableData的子属性 | 本期使用 |
| └sort | 标准：可选 | 排序字段说明：  ·sortCol说明表按照那列进行排序  ·shotType说明排序的方式=1大到小;=2小到大  是tableData的子属性 | 本期使用 |

**应用举例：**

1、first标签：

<d did=‘2’ showType=‘!first(3|num)’ value=‘quotas[1]’ decorateView=‘{#category#}占#percent#%’/>

value=‘quotas[1]’：数据必须符合“表格数据”

!first(3|num)：倒排序，按照num列，此时显示数据时，就要看“表格数据”是否有sort属性，并根据属性确定是否需要前端排序(若列表是不排序列表)，和取数据的方式

2、bar标签：

<d did=‘2‘ showType=‘bar’ value=‘quotas[1]’ label=‘category’, data =‘num’ decorateView=‘{lableShow:[category, percent]}’ />

value=‘quotas[1]’：数据必须符合“表格数据”

label=‘category’：“表格数据”中dataList数组中元素的category属性(“表格数据”中dataList的category列)作为标签(横坐标)

data=‘num’：“表格数据”中dataList数组中元素的num属性(“表格数据”中dataList的num列)作为标签(横坐标)

**七、样例**

|  |
| --- |
| {  \_HEAD:{  reportName:"“XXXSSS”分析报告"  },  \_DATA:[/\*data问题，一定要注重规范\*/  {\_id:1, \_url:'http://localhost:8080/sa/jsonD/getJsonD.do?uri=/demo/templetDemo/metedataInfo550E8400-E29B-11D4-A716-446655440000.jsond', \_jsonD\_code:'SP.TEAM-00007'},  {\_id:2, \_url:'http://localhost:8080/sa/jsonD/getJsonD.do?uri=/demo/templetDemo/anal\_quota\_716-446655440000.jsond'}  ],  \_TEMPLET:[  {  id:"seg1",  name:"上传数据CCC",  title:'<div style="font-height:bold;font-size:24px;">上传数据￥￥￥￥</div>',  subSeg:[  {  id:"seg1\_1",  name:"结构分析",  content:'<div style="font-height:bold; font-size:18px;">1、结构分析</div><br/>  <div style="font-height:bold; ">“<d did="1" showType="value" value="excelMdmArray[0].sheetInfo.name"/>”页签(sheet<d did="1" showType="value" value="excelMdmArray[0].sheetInfo.index"/>)</div>为新增结构，元数据结构分析结果如下：<br/>  <d did="1" showType="table" value="excelMdmArray[0]"/><br/>  <div style="font-height:bold; ">“<d did="1" showType="value" value="excelMdmArray[1].sheetInfo.name"/>”页签(sheet<d did="1" showType="value" value="excelMdmArray[1].sheetInfo.index"/>)</div>为原有结构，元数据用原有结构。<br/>'  },{  id:"seg1\_2\_1",  name:"单项指标分析",  content:'<div style="font-height:bold; font-size:18px;">2、单项指标分析</div><br/>  <div style="font-height:bold; ">“<d did="2" showType="value" value="mdName"/>”[<d did="2" showType="value" value="quotas[0].titleName"/>]</div>指标分析：<br/>  “<d did="2" showType="value" value="quotas[0].titleName"/>”中  大多为{<d did="2" showType="value" value="quotas[0].tableData.tableBody[0].category"/>}占<d did="2" showType="value" value="quotas[0].tableData.tableBody[0].percent"/>%，  {<d did="2" showType="value" value="quotas[0].tableData.tableBody[1].category"/>}占<d did="2" showType="value" value="quotas[0].tableData.tableBody[1].percent"/>%，具体分析数据如下：<br/>  <d did="2" showType="table" value="quotas[0]" titleDcrt="quotas[0].titles"/>  <d did="2" showType="pie" value="quotas[0]" label="category", did="num" decorateView="{lableShow:[category, percent]}"/>  <br/>  <div style="font-height:bold; ">“<d did="2" showType="value" value="mdName"/>”[<d did="2" showType="value" value="quotas[1].titleName"/>]</div>指标分析：<br/>  新加测试showType=“line”：<br/><d did="2" showType="line" value="quotas[1]" label="category", did="num" decorateView="{lableShow:[category, percent]}" />  新加测试showType=“bar”：<d did="2" showType="bar" value="quotas[1]" label="category", did="num" decorateView="{lableShow:[category, percent]}" />  测试showType=“first”降序：<d did="2" showType="first( !(3 |num ) )" value="quotas[1]" decorateView="{#category#}占#percent#%"/>  测试showType=“first”升序：<d did="2" showType="first( 3 |num )" value="quotas[1]" decorateView="{#category#}占#percent#%"/>'  }  ]  }  ]  } |