修改说明：1.添加了对简单关联函数的要求（已用红色标出）

2.添加了对三种关联函数的json示例

初等关联函数—没有区间合并：

Primary\_RelationFuntion(x, a0, b0, a, b, c, d, x0)

1. 其中x是属性value值是在确定属性时输入，除了这个都需要用户在确定完关联函数中输入。

参数：a0:满意区间左边

b0满意区间右边

a ：可接受区间左边

b ：可接受区间右边

c ：过渡区间左边

d ：过渡区间右边

x0：最优值

Primary\_RelationFuntion(x, a, b, a, b, c, d, x0)----满意区间与可接受区间合并

参数同上

Primary\_RelationFuntion(x, a0, b0, a, b, a, b, x0)----过渡区间与可接受区间合并

参数同上

1. 当输入关联函数值时，
2. 如果只能确定其中两个区间时，则提醒用户可以将满意区间(a0,b0)和可接受区间(a,b)合并

提示--------满意区间是属性值一定能满足要求的区间，可接受区间是超过此区间就会坏的范围。

1. 将过渡区间(c,d)和可接受区间(a,b)合并

提示--------过渡区间是不能满足条件但经过转化有可能成为可接收区间进而满足条件的区间。一般如果没有此区间，可以直接合并(常用)！

合并区间界面需求：

1.提供用户在输入满意区间和过渡区间旁边都提供一个确定合并按钮，旁边给个问号提示图案，

-------------------提示内容

如果需要区间合并， 必须先填写可接受区间!

还有上面2当输入关联函数值时的内容

2.如果用户在没有填写可接受区间的时候按下了按钮，会弹出提示框提示输入可接受区间，如果填写了可接收区间并按下了满意区间(可过渡区间)按钮，会自动将满意区间输入框锁定（edit变为false），并且将可过渡区间的值赋值给满意区间(可过渡区间)，同时按钮消失。

3.界面检测需求

(1)区间左侧是否大于右侧

(2)如果检测用户满意区间与可接受区间相同，则检测最优值是否在满意区间之间，不是则提示。

(3)如果检测用户满意区间与可接受区间不相同，则检测最优值x0是否在可接受区间之间，不是则提示

(4)如果满意区间(有一边或两边)超出了可接受区间，则提示，如果可接受区间(有一边或两边)超出了可过渡区间，则提示。

(5)检测当区间合并时，如果满意区间(合并了可接受区间和过渡区间时)或过渡区间(合并了满意区间和可接受区间时)

等于可接受区间时，则提示用户修改或者请用户选择简单关联函数

简单关联函数：

Simple\_RelationFunction(x, a, b, M)

1. x是用户输入属性值时输入，其他皆为确定属性关联函数时输入，输入框（应该有限制的算法）只能输入实数，如果存在正无穷和负无穷，按输入框底下的按钮。在界面上get出来（a,b）区间时，需要变成字符串输出。
2. 界面检测需求
3. 区间左侧是否大于右侧
4. M为最优值，检测效果跟初等关联函数一样，必须在(a,b)可接受区间之间
5. 简单关联函数存在正无穷区间(inf)和负无穷区间(“-inf”)，故(a,b)区间两边都需要需要存储为字符串，简单关联函数函数中会检测区分。

3.提示用户输入正无穷区间(inf)和负无穷区间(“-inf”)时，直接按确定填写按钮(放在输入框下)，然后界面自动赋值。以免用户输入错误

离散关联函数：

1. 一个表格，表头为属性值，k值
2. 表格上面提示

---------------------属性值如“优秀”，“良好”，等口语化词

K值必须小于1,大于-1

当k值小于0时，视为此属性值不满足条件

1. 界面检测需求
2. K值必须大于-1,小于1
3. 不存在没有属性值就有k值的情况

三种关联函数json示例：[{"uuid":"996cbbc79ba54d4da46e1f4c3a6f4e80",

"characters":[{//进入属性characterbean1

"id":0,//初等关联函数示例

"uuid":"ebefb0afc4264c73bdab23c0c4b12e79",

"name":"yan",

"value":"12",

"weight":0.3,

"functionType":4,

"object":[20.0,50.0,35.0,85.0,35.0,85.0,35.0],

"note":"属性的初等关联函数合并了可接受区间和过渡区间",

"K":0},{//进入属性characterbean2

"id":0,//离散关联函数示例

"uuid":"7965e91edc294ea18672f43540edaaa7",

"name":"si",

"value":"good",

"weight":0.7,

"functionType":0,

"object":{"good":0.8,"bad":0.2},

"note":"属性的离散关联函数 good:0.8,bad:0.2",

"K":0}],

"extendTypeId":0,

userId":0,

"X":0.0,

"Y":0.0,

"KValue":0.0,

"success":false},//第一个基元结束

{"uuid":"971a9bd0d0e847f2a063a3450c7d8bc9",

"characters":[{//进入属性characterbean1

"id":0, //简单关联函数示例

"uuid":"5f9cdbbf5e9b4042a4f45509d7db3721",

"name":"wei",

"value":"8",

"weight":1.0,

"functionType":2,

"object":["-inf","10",5.0],

"note":"属性的简单关联函数 a:-inf(负无穷),b:10,M(最优值):5",

"K":0

}],

"extendTypeId":0,

"userId":0,

"X":0.0,

"Y":0.0,

"KValue":0.0,

"success":false}]//第二个基元结束

说明：我只用了ElementBean来转json，所以处理时应该也是用它如果不是，请赶紧跟我说。。。