**模板转换目前存在的问题**

1. PonovoVM相应功能的参数基本都接不住PowerTest模板传过来的参数，之前省略的比较多，改变的也比较多，所以我要逐一把每个模块封装出一个针对PowerTest的功能，这个功能里只做两件事：
   1. 注册和PowerTest模板中<TestItemBegin><TestItemEnd>区段中所有被使用到的ID一一对应的参数；
   2. 在AfterSetParamter中将这些参数转换为PonovoVm对应功能实际使用的参数；
2. 完成上述功能我有以下疑问：
   1. 所有传入参数的值都必须是数值，不能是ID或者公式，这一点刚才已经和你确认过了；
   2. 因为<TestItemBegin><TestItemEnd>区段中所有ID都有唯一性和固定性，但通用参数、开关量和定值量区段中则不然，所以不建议额外对其他区段中的变量进行映射，可能导致某些模板缺值，而某些模板多值。是否能严格规范只映射<TestItemBegin><TestItemEnd>区段中的ID，以便完全避免这些问题？
   3. 如果额外对Binary中的参数进行映射，那就会存在从KA、KB、KC、KD这四个ID到BINARYIN这一个ID的转换，但BINARYIN已经有值并映射过了，PonovoVM中同一个ID将有两个数据来源，可能有错。

但如果不映射，额外封装的PowerTest功能就会没什么实用性，仅仅作为对接tpl而存在，对用户来说没什么使用价值。我觉得比较矛盾。

* 1. 像平衡系数获得方式（直接输入或手动计算）这种参数也是一样，不映射吧用户就没办法用，映射吧起不到实际作用，因为用到这个参数的地方并不是我来算的，而是传进来就已经算好的。所以此类为模板转换而封装的功能能否即不在左侧列表中可见，又能包含在导出的TestMacro中被COM接口调用？

解决方案

修改映射配置文件如下：

<macro-map name="ShotCom.ShotComponent" id="ShotCom.ShotComponent" mapname="距离保护试验" mapid="Distance">

<para-map-expr name=”” id=”” expr="Ztype==0">

<para-map-cal mapname="阻抗定值" mapid="ZSetting" cal-scpt="sqrt(R\*R+X\*X)"/>

<para-map-cal mapname="阻抗角" mapid="ZPh" cal-scpt="atan(R/X)"/>

</para-expr>

<para-map name="R" id="R" mapname="" mapid="" use="1"/>

<para-map name="X" id="X" mapname="" mapid="" use="1"/>

<para-map name="Z" id="Z" mapname="阻抗定值" mapid="ZSetting" use="1"/>

<para-map name="ZPH" id="ZPH" mapname="阻抗角" mapid="ZAng" use="1"/>

程序改进说明：

1. CPNTplEquation

增加了m\_fValueCal、m\_bHasCal，用于记录变量的计算值

1. CPNTplGroup

增加CPNTplEquation\* FindValueByID(const CString &strVarID)，用于查找数据组对象的属性为“V”的等式，即数据值

1. CpttAtsTransTool

TransElement\_TestItem函数末尾增加调用：

//参数映射表达式处理

Trans\_MacroMap\_Exprs(pElement, pMacroTest, pMacroMap);

1. 参数映射表达式各函数说明

**Trans\_MacroMap\_Expr**(CPNTplElement \*pElement, CMacroTest \*pMacroTest, CParaMapExpr \*pParaMapExpr)：如果表达式pParaMapExpr的值为TRUE，则处理，否则不处理。表达式包括AtsPara的计算公式

**Trans\_MacroMap\_Expr\_Cal**(CPNTplElement \*pElement, CMacroTest \*pMacroTest, CParaMapCal \*pParaMapCal)：计算公式处理，计算公式的值，然后添加到测试功能参数中。

Trans\_MacroMap\_GetValueByVarID(const CString &strVarID)：获取PowerTest数据变量的值

Trans\_MacroMap\_FindValueByVarID(const CString &strVarID)：查找PowerTest变量

Trans\_MacroMap\_GetEquationVal(CPNTplEquation \*pEquation)：计算PowerTest等式的值

这里面需要注意的是：等式CPNTplEquation对象后面可能是表达式，表达式可能包含PowerTest变量，因此需要递归计算变量的数值