算法库中英文切换设计方案

## **需求描述**

AT目前支持多语言功能，但是对于算法库的中英文支持不够，在安装完MACS651系统后，算法库的语言版本就定死了。AT的语言切换不能切换算法库的语言版本。

算法库中涉及的中英文切换的内容有：

1、POU别名

2、POU注释

3、变量说明

由于IEC编辑区的注释不在工程中显示，所以此处不做中英文切换。

需要提供一种跟现在的界面一样、操作一样的方式给二次开发人员，让其能在同一个库文件中保存多种语言版本。

## **方案设计**

针对以上需求分析，有以下若干种方案：

方案一：

***原来的场景：***

二次开发人员定义POU，在POU别名中输入某种语言的文字，在POU注释中输入某种语言的文字，在POU中定义变量，在变量说明中输入某种语言的文字。写组态逻辑。保存该库。然后做同样的操作保存为另一种语言的库。

***现在的场景：***

二次开发人员定义POU，在POU别名中输入语言一的文字@@语言二的文字@@语言三的文字@@……@@语言N的文字，在POU注释中输入语言一的文字@@语言二的文字@@语言三的文字@@……@@语言N的文字，在POU中定义变量，在变量说明中输入语言一的文字@@语言二的文字@@语言三的文字@@……@@语言N的文字。写组态逻辑。保存该库。

***后台处理分析：***

1、POU别名、POU注释、变量说明的字符限制去掉，序列化过程不变。

2、该库作为引用库，在库中变量序列化时，对读上来的POU别名、POU注释、变量说明进行处理，处理函数如下：

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*功能 :跟据语言ID，解析对应节的语言字符串

\*参数 :strParse,要解析的字符串，解析的结果也存储在该字符串中。

\*返回值 :无

\*创建者 :yewen

\*创建日期 :2012.04.28

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//目前的语言顺序 中文@@英文

void ParseStringWithSpecialLang(CString &strParse,UINT iLang);

3、POU别名、POU注释、变量说明的显示代码不变。

4、对于当前打开的工程或打开的库中的变量，所以操作不变。

***优缺点总结：***

1、代码改动很小，只有两处：POU别名、POU注释、变量说明的字符长度限制去掉；引用库中的变量序列化时对POU别名、POU注释、变量说明进行解析处理。

2、对于最终用户，显示上跟现在没有任何区别，最终用户也无法看到库中定义的各种语言的文字。只有在编辑库时才能看到各种语言的文字。

3、二次开发人员开发时比较方便，能在定义一个东西时立即加上各种语言的注释。

4、界面显示不够友好，第一次接触的人可能不知道这个规则。

5、语言顺序难以记忆。

6、语言过多时，文本框显示不全所有的文字。不过有替代方法：在记事本中编辑好内容，粘贴到文本框中。

方案二：

鉴于文本框的限制导致用户看起来比较吃力。方案二重点在于优化用户体验。

***若当前打开的是库：***

在POU的定义过程中，提供如图1所示界面给用户。在POU别名后有一个选则语言的下拉框，在下拉框后是输入的文本，可以点保存将当前语言和文本记录下来。下拉框切换语言后可以写新的语言的文本。POU注释同理。



图 1新建POU界面

在变量的定义过程中，改变原来Grid控件的显示外观，实现一种新风格的Grid控件的Cell，这种新控件外观如一个按钮，单击弹出如图2所示的对话框。上面的下拉框中可以选择语言，下面的下拉框中可以显示对应语言版本的变量说明。



图 2变量说明弹出对话框

当该变量增加了变量说明，则在该Cell的按钮上显示“…”。

***若当前打开的是工程：***

显示风格与原来一致。

***后台处理：***

1、增加POU别名、POU注释、变量说明分别对应的语言Map。这三个Map是语言和对应的文本的映射。

2、库修改POU别名、POU注释、变量说明后的值存储到对应的Map中。在该库作为引用库时，序列化其中内容时，读取软件语言，从Map中取出对应语言版本的文本，然后赋给POU别名、POU注释、变量说明所对应的变量。

3、变量显示Grid时判断是工程还是库，若是工程采用原来的显示和处理方式，若是库则将变量说明部分显示为新风格的Grid，然后将Map中的数据填充到图2的窗口中。

4、新建POU时由于POU别名和POU注释只在库中显示，所以此处统一改为新风格即可。

***优缺点：***

1、需要增加新风格的GridCell，难度较大。

2、需要修改新建POU界面的显示。需要针对工程和库分别显示不同的变量说明栏。

3、需要修改序列化。

4、界面体验较好。

方案三：

针对前两种方案都存在一定的缺陷。提出了第三种方案。

***前台显示：***

跟现在一样。

***后台处理：***

1、增加三个Map存储不同语言的POU别名、POU注释、变量说明。

2、打开库时，根据语言版本取Map中的值填充到POU别名、POU注释、变量说明中，用于显示。

3、保存时，根据语言版本将POU别名、POU注释、变量说明的值写回对应的Map。

***优缺点：***

1、改动小

2、二次开发人员不需要记住语言顺序，只需写当前语言的注释。

3、二次开发人员写其他语言注释，需要用对应语言的AT再次打开该库编写注释。不过相对于之前中文库和英文库需要全部重做已是很大的进步了。