

ASiV 的国际化操作 TM650



贝加莱工业自动化
Perfection in Automation
www.br-automation.com



前提

培训模块: TM610 – AsiV 的基础

软件:

AS 2.5 或更高版本 / AR 2.80或更高版本

硬件:

4PP220.1043-75

目录

1 · 简介	3
1.1 目的	4
· 语言切换	5
2.1 语言切换要求	5
2.2 语言	5
2.3 字体	6
2.4 Unicode	9
2.5 可视组件字体	11
2.6 可视组件语言切换	12
2.7 在VC上一种新的语言的影响	20
3 · 语言切换实例	21
3.1 转换默认的文本	21
3.2 使用不同的字号	22
3.3 使用变量切换语言	23
4 · 切换物理单位	24
4.1 单位	24
4.2 比例	26
4.3 界限	27
4.4 单位组	27
4.5 连接单位组与数据点	31
4.6 简单比例	32
5 · 单位切换实例	37
5.1 显示带小数位的数值	38
5.2 从摄氏度切换到华氏度	39
6 · 总结	41

1、简介

在这个培训模块中，我们将了解到国际化图文显示应用意味着什么。这样可以很容易将机器销售到全世界而不需花费很多努力。



图. 1 国际化的可视组件

这个培训模块不仅涉及了项目的国际化方面，还通过分配物理单位简单设置了数据点和机器——以及过程——相关的限制也会讨论。

简介

1.1 目的

可以创建多语言的可视化应用。

此外，应该认识到语言和单位是不同的世界。

这里也包括在可视化应用中切换语言和单位带来的影响。

欢迎进入贝加莱世界 *)

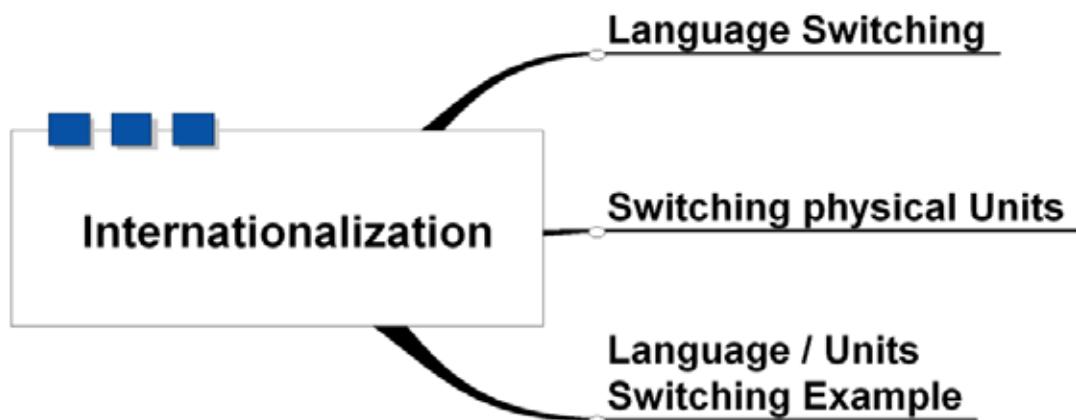


Fig. 2 目标

) Welcome to the world of B&R

2、语言切换

可视化应用的一个基本功能是具有显示不同文本的能力。

然而，只用一种语言显示文本是不够的。在语言切换中还有很多复杂的要求。

2.1 语言切换要求

当执行语言切换时，需要满足以下一些粗略标准。

- 将文本切换到需要的语言
- 切换单位（无须依赖语言）
- 将键盘切换为各个国家的布局
- 切换字体来显示不同的字符代码
- 显示Unicode字体
- 输入字符的下一个过程
- 切换位图

2.2 语言

术语语言具体是指一种语言，如英语或日语。口语分成语言族。ISO 639-1 和 639-2 语言代码可以识别任何一种国际化语言。

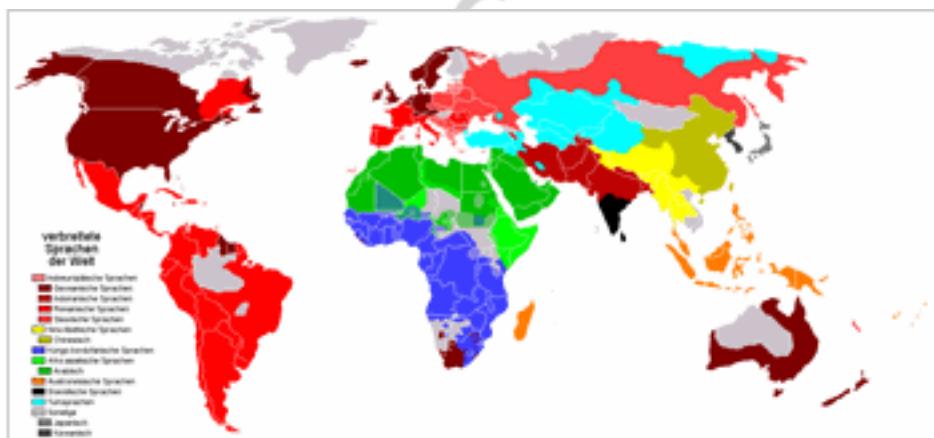


Fig. 3 世界的语言

2.3 字体

字体是印刷行业上的术语，是指字符设置的实际形式。

"ABC"

2.3.1 比较字体

一种字体中的每个字符的宽度一般是不同的，例如w比i占的空间要大。这些字体叫做比例字体。这组中常用的字体包括Times和Arial。

非比例字体最早用于打印机。为了简化打印机的结构，这些字符都采用相同的宽度。大概在这些字体中最为人熟知的是Courier。

这些字体也应用于早期传统的计算机终端中，这就是为什么非比例字体仍然在IT中使用的原因。

在衬线字体和非衬线字体之间也是有区别的（在法语中sans意思是"without"）。衬线字体是字母在笔画终端处的小特征。

衬线字体会形成一道水平线，更易于读者的眼睛定位。因此，衬线字体特别适合于书籍和文章的文本印刷。

今天的计算机监视器与印刷机相比有极低的分辨率。在监视器上使用衬线字体是阻碍了阅读而不是辅助阅读，因为这不像它们在实际文字中那么清楚。

因为不是所有的字体都能在每个系统上使用，所以将它们分成了组。当一组中的一种字体不能呈现，则可以使用这个组中的其它字体来代替。

- 比例且衬线字体：Times New Roman, Times (serif)
- 比例且非衬线字体：Helvetica, Arial (sans serif)
- 非比例字体：Courier New, Courier (monospace)

2.3.2 B&R 字体 CD (5SWFON.0000-00)

在选择字体时，必须要考虑许可条件。可视组件编辑器是基于Windows的编程环境，并可以用来创建可视化应用，因此可以选择Windows中的任何字体。

在运行时，只能使用共享的或免费的字体（即无需注册的字体），除非已经购买了相应字体的许可权。这是因为在运行并没有使用Windows系统。

B&R为Power Panel设备和5SWFON.0000-00 Font CD上的IPCs提供了两种不同的字体。

Thorndale WT (大约. 22 MB)

- 欧洲语体，斯拉夫和希腊字母
- 阿拉伯语
- 简体汉字
- 繁体汉字
- 韩语字母
- 日语字母

Andale WGL (大约. 620 kB)

- 欧洲语体，斯拉夫和希腊字母
- 阿拉伯语

在CD-ROM中包含了这两种字体，但是Thorndale字体有四种变化。这四种变化中的每一个都有一种最佳显示类型的亚洲字体，但是这四种字体都有完整的字符集。

Thorndale以衬线字体来显示西方字母。然而，CD中也包含Andale WG字体，它是以非衬线字体来显示西方字母的。

许可及供货形式

CD中提供了这些字体，顾客必须购买CD一次。字体许可证有自己的模块号并以粘贴注册号的形式提供(5SWFON.0001-00)。与许可证合同一致，这些字体只能安装在B&R器械上。许可证费用包含了在一个目标系统中使用这两种字体的权限。

2.3.3 TrueType字体和位图字体之间的不同

TrueType 定义了一种保存字体的方式（有关字母形象的信息）。TrueType 字体是基于向量的，即它们的大小可以增加或减少而不丢失质量。

相对而言，位图字体是以象素图形来保存的。在字体文件中没有正确保存的字号必须用插补的方法生成，这样就会丢失质量。

注意：

在可视组件中，所有的TrueType 字体都可以无限制地使用。位图字体可以使用其“实际大小”。

2.4 Unicode

在所有已知的文本和字符系统中，Unicode是一种为每个图形字符或元素定义数字代码的国际标准。这将解决在不同国家使用编码不兼容的问题。传统的计算机字符系统包含由128(7-bit)个字符或256(8-bit)个字符(ASCII = 7-bit)组成的字符集。如果排除了控制字符，字符系统就分别减少到96和230个字符和特殊字符。这些字符编码在同一个文本中只允许同时显示几种语言。

这严重地阻碍了国际化的数据交换。根据工业的，国家的和ISO标准，Unicode提供了一种清晰的方式来显示在已有字符集中的所有字符。

在一开始，很多人认为65,536个字符(16-bit)已足够了，但很快证明实际并不是这样。现在这一标准已设计为1,114,112(2²⁰ + 2¹⁶)字符，它覆盖的编码范围从U 000000到U 10FFFF。到现在为止，Unicode 4.0已分配了96,382个不同的字符编码，这相当于输入有效范围的9%。

2.4.1 历史回顾

最早的一种数字编码是Morse码，但随着电传、Baudot码和Murray码的引入，Morse码很快受到电报线的排挤。从Murray码开始，距离ASCII码只有一步之遥。在计算机早期年代，认为ASCII码用于文字和字母的标准编码，它代表了用于信息交换的美国标准编码，而且是最早用英语显示的字符。此后ASCII码被扩展了，因此也可以显示其它语言的特殊字符了。不幸的是，8-bit码由于剩余的空间太小，因此不能同时处理所有的特殊字符，这就是它不得不扩大几倍的原因。用8位码容纳汉字也是不可能的。这一编码最终导致了Unicode，一种支持所有文化和地区的所有字符的编码。今天，Unicode是国际范围内交换信息的不可缺少的标准，并且在世界范围的网络中给人留下了深刻印象。

2.4.2 可视组件和Unicode

Unicode字符在Visual Components 2.4.1的版本中可以显示和工作。

"Unicode 字符可以在
所有 SG4 目标上显示
(Power Panel 100 / 200, IPC, APC)."

使用Unicode字体需要注意:

- 在目标系统中, Unicode字体需要比ANSI字体更多的内存空间, 因为它包括更多的字符。
- 用于翻译的工具也必须具有Unicode字符。

在Windows操作系统中, 这通常不会有什不同, 因为Windows操作系统通常都会有足够的硬件驱动空间。

可视组件中使用Unicode的规则:

- 静态或动态文本, 列表框和报警系统(TM640)都可以使用Unicode字符。
- 文字和数字的输入/输出区域可以数据类型为USINT, STRING (ANSI字符), 或UINT (Unicode字符) 的数组来显示字符。
- Unicode字符只能在运行中的某个状态输入。IME (输入法编辑器), 可以在Windows的VC编辑器中使用, 但在运行中不能使用。
- 可视组件可以用Unicode字符在内部单独处理文本。

2.5 可视组件字体

对每种语言来说，可视组件字体是管理一种独立的实际字体，它的属性为：

- 字体
- 字号(像素)
- 加粗 和/或斜体显示

2.5.1 可视组件默认字体

与可视组件一起安装的字体是Bitstream Vera Sans字体。



图. 4 默认字体: Bitstream Vera Sans

2.5.2 在运行中管理字体

当编译完VC项目时，它的字体也会编译到一个.BR模块中。在一种字体中使用多少种字号是无关紧要的，以为因为我们毕竟在使用TrueType字体。

然而，如果使用了加粗或斜体，那么就要创建一个独立的.BR模块。这是因为在Windows中，也使用独立的字体文件来显示这种类型。

语言切换

2.6 可视组件的语言切换

这部分将涉及以下主题：

- 语言设置
- 语言切换规则
- 添加一种新的语言
- 使用变量来切换语言
- 为不同语言输入文本
- 导出/导入语言文本
- 切换键盘布局
- 根据语言切换位图

2.6.1 语言设置

语言设置是用于创建可视化应用中的语言。但通常在创建可视化应用时，就已创建了几种语言。

在VC编辑器的全局属性下定义语言设置。

当要创建可视化应用时，必须确保所有语言的文本完全适合于所提供的区域。

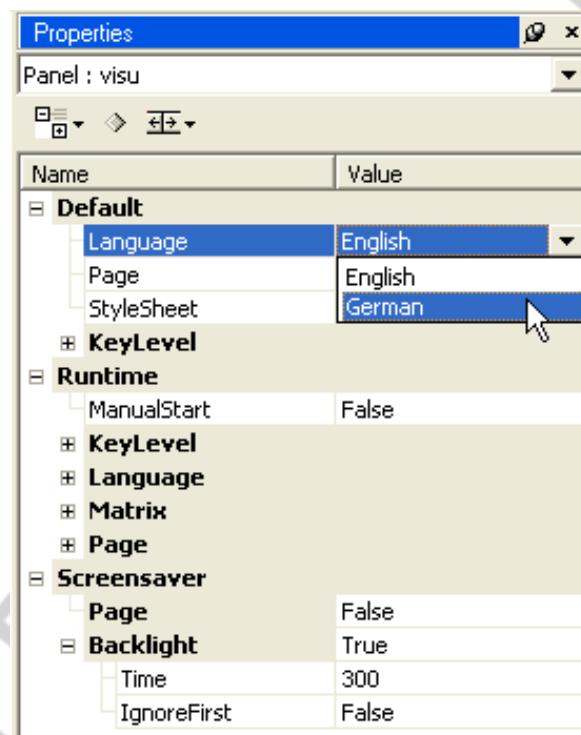


图. 5 项目属性 - Language

2.6.2 语言切换规则

在可视组件中切换语言时必须遵守几条规则：

- 语言的个数取决于目标系统
- 字母数字输入的进一步处理

语言的个数只受所用目标系统的限制。

如果要在可视化应用使用更多的语言和字体，那么目标系统就需要更多的内存（CF卡，RAM）。



图. 6 5CFCRD.2048-02

由于每个字符通过一个字节来管理，所以进一步处理ANSI文本对目标系统提出了很高的要求。

在编辑Unicode文本时，必须记住在Automation Runtime (AR) 系统中有一些限制：

- 在AR中没有Unicode数据类型。文本可以以2字节类型的数组来存储，并再次显示。AS里的AswString库用来执行用户任务中的评估操作。但记住如果真得有必要或有意义，可以在应用程序中用任何一种语言输入一个变量字符串。

2.6.3 在不同的运行系统中交换数据

当交换数据时，确保在AR中创建的文件是以FAT16文件系统保存的。这样可以防止文件名中包含Unicode字符（与FAT32 或 NTFS文件系统不同）。在命名文件时要考虑到这点，但并不限制文件的内容。

语言切换

2.6.4 添加一种新的语言

在默认情况下，VC项目包含两种语言，德语和英语。

如果选择了语言组件，可以点击键添加一种新语言。



图. 7 添加对话框

新语言可以从语言复选框中选择。

编译检查框可以定义是否在运行中编译语言，即只显示在目标系统上，或只为开发环境设置。

国家区域允许缩小语言范围，例如美国英语或英国英语。

索引定义了语言的顺序。当从数据点切换到语言时，索引用来选择语言本身。

2.6.5 使用数据点来切换语言

在工程开始后，所有的文本都以默认的语言输出（直到指定一个数据点来处理语言切换）

如果有切换语言的数据点，那么将它连接到全局项目属性下的Language/ChangeDatapoint属性。

激活的语言也可以从位于ChangeDatapoint属性中独立的数据点读出。

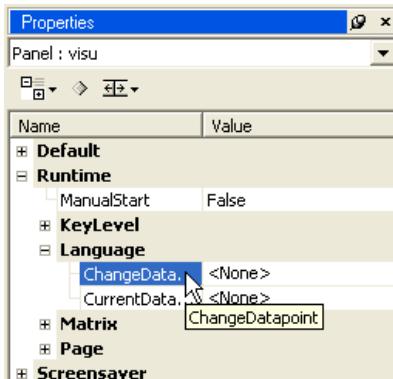


图. 8项目属性 – 语言切换

2.6.6 用不同的语言输入文本

在VC中，文本可以用不同的方式输入：

- 在文本属性中直接输入文本
- 当传送完文本，之后从语言资源（文本）中导入文本

在最佳的情况下（多语言设置），文本是在文本属性（图 9）或在不同的编辑器（图 10）中直接输入。

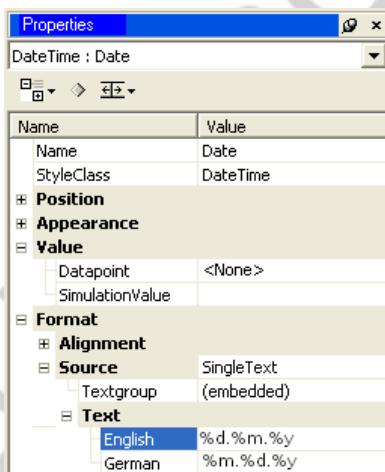


图. 9例子 – 转换日期

语言切换

Index ▲	English	German
0	Main Menue	Hauptmenü
1	Display Text and Values	Darstellung Text und Wert
2	Display Graphic Functions	Darstellung Graphikfunktionen

图. 10 例子 – 在文本组中转换

2.6.7 改变输入语言

Windows XP 和 Windows 2000都提供了改变输入语言的选项。

注意:

这是Windows的一种功能，而不是VC编辑器的功能。

如果这个选项已经安装到了Windows Control Panel (Windows / Control Panel / Regional and Language Options) 中，那么在VC编辑器中切换输入语言是可行的。

注意:

如果还没有安装这一特性，则需要Windows Setup CD。



图. 11 输入语言



图. 12 控制面板I -输入语言

2.6.8 导入/导出语言文本

对于转换可视化应用，最重要的工具是语言文本的导入和导出。

当导出文件时，它的参考信息——即可视组件中使用的控制名称和所选语言层的文本——存储在文本格式 (CSV) 中。

[Tools] / [Import texts] 和 [Export texts] 菜单项用于导入和导出文本。

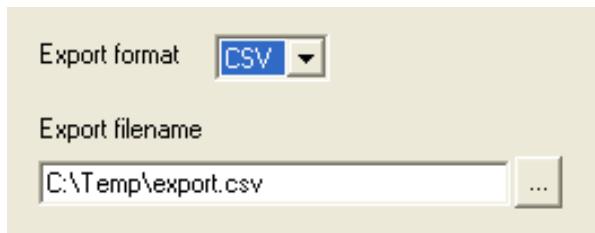
对话框可以指导用户完成这个过程，从而得到需要的结果。

Importiere Texte

Exportiere Texte

Importiert Objekte aus einem anderen Projekt

图. 13 工具菜单 – 导出 / 导入



给导出的文件定义一个名字，点击<Next>到下一个对话框。



在这个对话框中可以定义一个或多个参考语言。你可以选择是否导出只为其余语言留有的文本资源。

当<Next>确认所有的对话框后，<Finish>关闭最后一个导出窗口。

语言切换

导出格式:

下面显示了以文本形式导出的文本摘录。每个文本资源都有自己的ID，保证文本在导入时分配无误。

```
ID;German [de];English [en]  
Source[global].UnitGroup[Length].ID[115];Millimeter;millimeter  
Source[global].UnitGroup[Length].ID[116];mm;mm  
Source[local].Layer[Header].ID[87663];%m.%d.%y;  
Source[local].Layer[Header].ID[87905];%H:%M:%S;
```

警告:

在转换过程中绝对不能改变ID。

由于在导出时生成了CSV文件，导出文件可以用不同的程序例如Microsoft Excel再次打开、编辑和保存。

在保存文件时，确保文件格式没有改变并且使用Unicode编码(UTF-16)。

2.6.9 为文字数字输入改变键盘布局

切换语言时也切换键盘布局。

切换键盘布局只在输入文字数字的字符时有意义。不管使用哪种语言，都会发生数字输入。

注意:

在培训模块TM670中，描述了如何切换键盘布局。

在输入Unicode字符时有一些限制。虽然很多键可以用变化触摸板布局和切换它的选项来设置，但这只能用于字符代码的一小部分。

2.6.10 根据语言切换位图

语言切换不仅仅影响文本和键盘输入；位图也可以根据语言来切换。

在培训模块TM670中，有一个简单的应用是用来切换文字数字触摸板。

为位图组定义附属语言。

让我们看一下AlphaPad位图组属性。

如果LanguageDependent属性设为True，则每种语言的位图将显示在组位图画面上。在默认情况下，每种语言使用相同的位图。

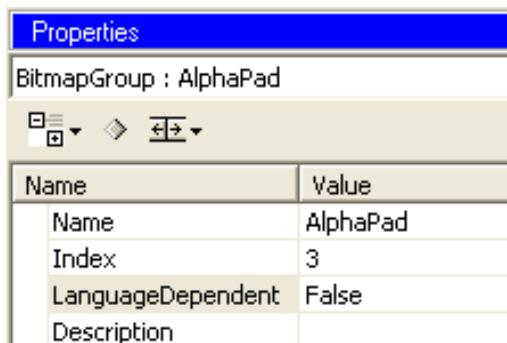


图. 14 位图组属性

Index	English	German
0	AlphaPadVGA	AlphaPadVGA
1	AlphaPadVGA_pressed	AlphaPadVGA_pressed

图. 15 AlphaPad 位图

在默认情况下，alpha板使用英文的QWERTY键盘布局。当前显示德文QWERTZ键盘布局的位图可以通过按压和释放状态，替换已存在的位图。

然后只需对各种语言设置按键功能了。

2.7 在VC上一种新的语言的作用

在可视化应用中添加新语言时，需要记住几点：

- 新的语言对控制器应用有什么影响？
- 需要几种语言？
- 默认的语言是什么？
- 文本是如何管理的？
- 需要什么字体和字号？
- 谁将转换可视化应用中的文本？
- 将使用什么机制来选择语言？
- 在目标系统上使用新的语言有什么效果？

在创建项目前，应该了解并理解项目中每种语言的要求。

下面的例子将演示如何在语言设置中安排文本及其影响。

- 默认文本: Notaus
- 英文转换: Emergency Stop
- 中文转换: 奥地利没有袋鼠

在创建项目时，如果只考虑默认的文本，那么将发生下面的错误：

Notaus

Emerg...

■ ■ ■

必须合理设置文本区域的大小从而能正确显示文本：

奥地利没有袋鼠

3、语言切换实例

这一示例项目使用了在TM610培训模块中创建的可视化项目。

那个培训模块包括一些可以独立执行的小任务。

以前章节关于语言切换和可视组件的帮助可以作为执行这个任务的基础。

"通过实践学习!"

在这些实例中包括以下主题:

- 为语言切换转换现有的默认文本
- 使用不同的字号
- 使用变量切换语言

3.1 转换默认文本

转换默认文本有几种不同的方式。如果文本为不同的组件输入到VC编辑器中（图片，文本组），那么建议在属性中转换文本。

另一种方法是导出文本，并且转换文本后再次导入它们。

任务:用不同的语言添加文本

例程（来自默认的项目）中已经包含了两种语言，德语和英语，并且文本已经按其中一种语言来显示。大家可以转换其中的一些文本（不检测转换的实际对错）。

你在哪里学的汉语？

Where did you learn Chinese?

语言切换实例

3.2 使用不同的字号

可视化应用不仅仅由一种字体或字号组成。设置字体是用户的任务。我们将使用并显示三种不同的字号——LargeFont, MediumFont 和SmallFont。

任务: 创建三种不同大小的新字体



我们将使用默认的字体创建三种新的逻辑字体，字号大小为8, 14, 20。

任务: 应用字体



在可视化对象中应用其中的一种字体(如数字输出区域)。

结果



主页面中的输入输出区现在看起来是这样的:

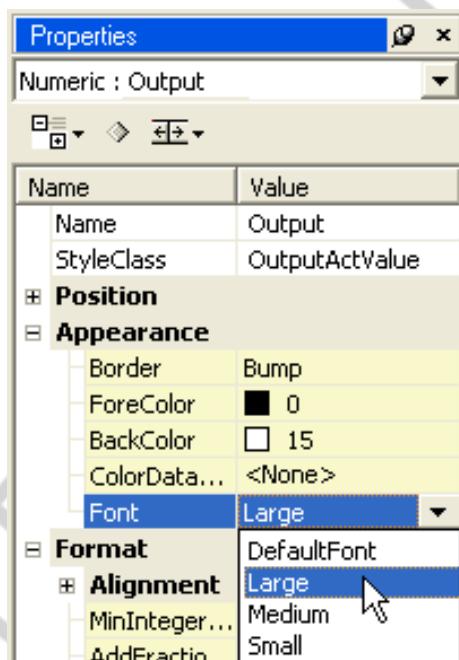
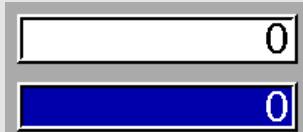


图. 16 设置字体

3.3 使用变量切换语言

运行中，使用语言的索引值来切换语言。新语言的文本在索引改变后会立即输出，而不是在页面改变时输出。

Index	Language	Country	Compile
0	German	<no country>	True
1	English	<no country>	True

图. 17 VC 项目语言

连接语言变量，可以在运行中切换项目中的语言。

因为切换语言是一个全局功能，所以应在全局项目设置中设置。

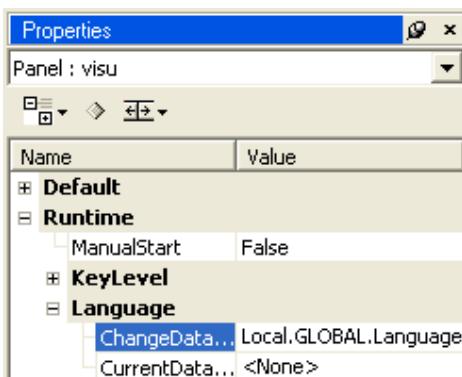


图. 18 连接语言变量

任务: 使用变量切换语言

首先在"panel"用户任务中创建一个新变量 Language，数据类型为UINT。
然后将这个变量与全局项目的属性连接。

实际的切换可以在用户任务的Watch窗口下实现，或者创建两个功能为设置数据点的按钮来实现。

另一种方法是连接ListBox的数据点。

结果

在运行中，将变量Language的值设为0时输出英文文本，而将变量值设为1时则切换到要转换的语言。

切换物理单位

4、切换物理单位

一个国际化的项目不只是切换文本；过程值的实际单位也必须转换。

在大多情况下，控制器程序只使用整数值工作。例如，一个温度传感器可能返回一个精度为0.1摄氏度的温度值。

然而，测量单位摄氏度不一定是所有国家的惯用单位。因此，温度应该也能用华氏来显示。

显示这个数值时需要以下功能：

- 显示带小数位的数值
- 使用不同的单位显示和转换数值

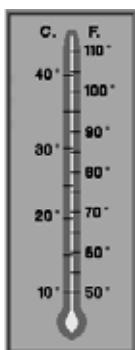


图. 19 单位显示

4.1 单位

测量单位用于确定事件的物理大小或量值。单位系统是一系列的规则，用来决定在科学和工程中意外出现的数值的。

全世界现在使用的单位系统叫作International System of Units（国际单位系统），或SI（来自于法国Système International d'Unités），它是自1960年第11届度量衡会议（CGPM）上引入的。此后，SI取代了很多不同国家的单位系统，尤其是那些用于自然科学的单位系统。所以在不同单位之间就不需要再进行的复杂转换了。新的国际单位系统分成两个单位级别：基本单位和导出单位。

基本单位:

单位	度量	缩写
Meter	长度	m
Kilogram	质量	Kg
Second	时间	s
Ampere	电流	A
Kelvin	温度	K
Mole	物质的量	Mol
Candela	亮度	Cd

直到1971年，第14届CGPM才接受mole为第七个基本单位。

将重量或长度单位由metric单位转换为UK/US单位的实例:

Metric 单位	UK / US 单位
1 克 (g)	0.032151 盎司金衡制 (t oz)
1 公斤 (kg)	35.273962 avdp. 盎司(avdp oz)
1 公斤 (kg)	2.204623 avdp. 磅 (avdp lb)
1 吨 (t)	1.102311 短吨
1 吨 (t)	0.984207 长吨
1 毫米 (mm)	0.03937 英寸 (in.)
1 米 (m)	39.37 英寸 (in.) = 3.28083 英尺 (ft.) = 1.0936码
1 千米 (km)	3280.83 英尺 (ft.) = 1093.61 码 = 0.62137 英里

切换物理单位

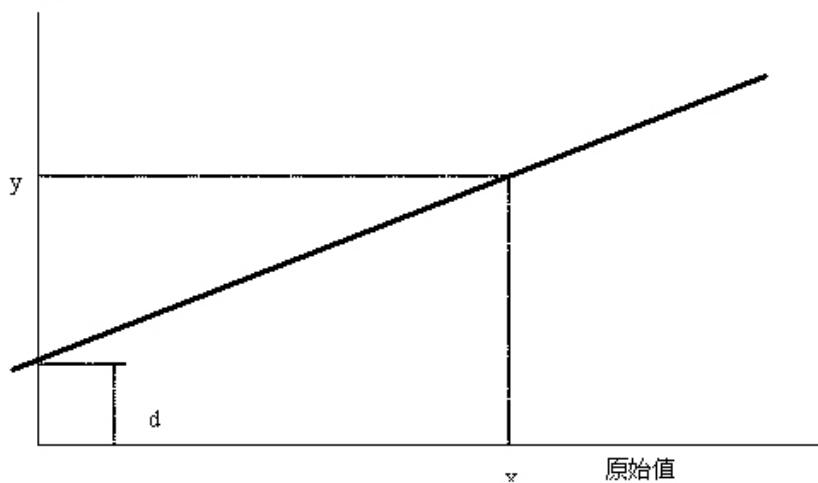
4.2 比例

比例是指将一个物理值（原始值）转变为一个可以显示的值（过程值）的过程。

比例使用以下的线性方程：

$$y = k * x + d$$

过程值



在温度例子中，我们得到一个精度为1/10摄氏度的整型原始值，即 20.0°C 时得到的整数值为200。

为了显示原始值20.0，可以使用上面给出的公式：

$$k = 0.1 = \text{斜率}$$

$$d = 0 = \text{偏移量}$$

$$y = 0.1 * 200 + 0 = 20$$

我们使用这个公式将温度转化为华氏度：

$$^{\circ}\text{F} = ^{\circ}\text{C} * 9/5 + 32$$

4.3 限制

原则上，数值受其数据类型的限制，但连接到生产过程或机器的因素不一定总是足够的。

既然如此，输入值应该受最大值和最小值的限制，这样在系统运行中，用户才不会输入一个不正确的值，或引起运行顺序上的错误。

限制可以具有不同的形式：

- 静态限制
- 根据机器设置的限制
- 动态限制
- 根据仪表/材料设置的限制
-

4.4 单位组

为了学习关于温度变量的例子，有以下要求：

- 按比例的显示过程值，将原始值转换为带小数位的值
- 切换单位，以[°]C 和[°]F 显示
- 定义最大值和最小值的限制
-

将所有这些合并为一个单位组来管理。

VC 单位组的节点管理所有可视化应用中的单位。

Name	Default Limit	Default PLC Unit	Default Display Unit
Length	None	PLCUnit	Meter
Mass	None	PLCUnit	Kilogramm
Memory	None	PLCUnit	Byte
Power	None	PLCUnit	Watt
Pressure	None	PLCUnit	Pascal
Temperature	None	None	Celsius
Temperatures	None	PLCUnit	Celsius
Volume	None	PLCUnit	Cubicmeter

图. 20 默认项目中的单位组

在这里显示的单位组已经是默认VC项目的一部分。

切换物理单位

4.4.1 单位组中的元素

一个单位组是由很多不同的单位组成。

单位是通过索引号、名称、长和短文本、精确度及其协调动作来定义的。

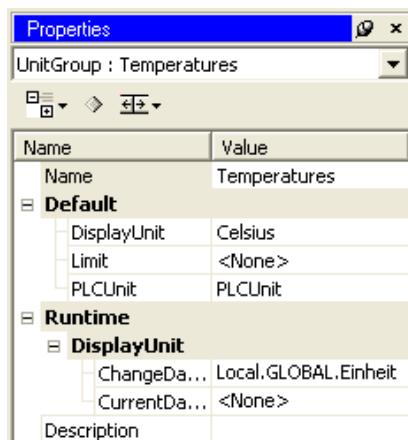
Unit Group: Temperatures				
Index	Name	Unit Abbreviation	Unit Description	Default Precision
0	Celsius	°C	Celsius	1
1	Fahrenheit	°F	Fahrenheit	1
50	PLCUnit			1

图. 21 Temperatures单位组

4.4.2 单位组属性

单位组属性描述了它在运行中是如何运转的。

只要单位组连接到数据点，那么就可以通过数据点的属性来估计单位组的属性。



默认属性:

- DisplayUnit
- 如果单位组没有连接数据点，那么目标系统以单位开始。如果单位组连接了数据点，那么在运行中目标系统会改变显示单位。这是因为DisplayUnit已经在应用程序中使用，且它的值初始化为0或其它数值。
- Limit
- 限制与Limit = <Default>下的数据点一起使用。
- PLCUnit

PLC单位与PLCUnit = <Default>下的数据点一起使用。

4.4.3 单位切换是如何工作的

上面的图像介绍了一些关于VC比例是如何工作。

VC比例实际上是两个步骤。首先将过程变量值（原始值）按比例转化成内部单位；然后显示计算值。

两个步骤进行比例缩放的优点：

- 变量以内部格式存储，应该尽可能早的总结出格式，并且格式不应该包含在整个比例缩放的过程中。
- 将变为同一单位组中的数据点的变量以不同的格式存储在应用程序中。例如，一个温度调节器可以按内部 $1/100^{\circ}\text{C}$ 的分辨率来工作，而另一个按 $1/10^{\circ}\text{C}$ 的分辨率来工作。但相同的单位和限制也会用于这些数据点上。

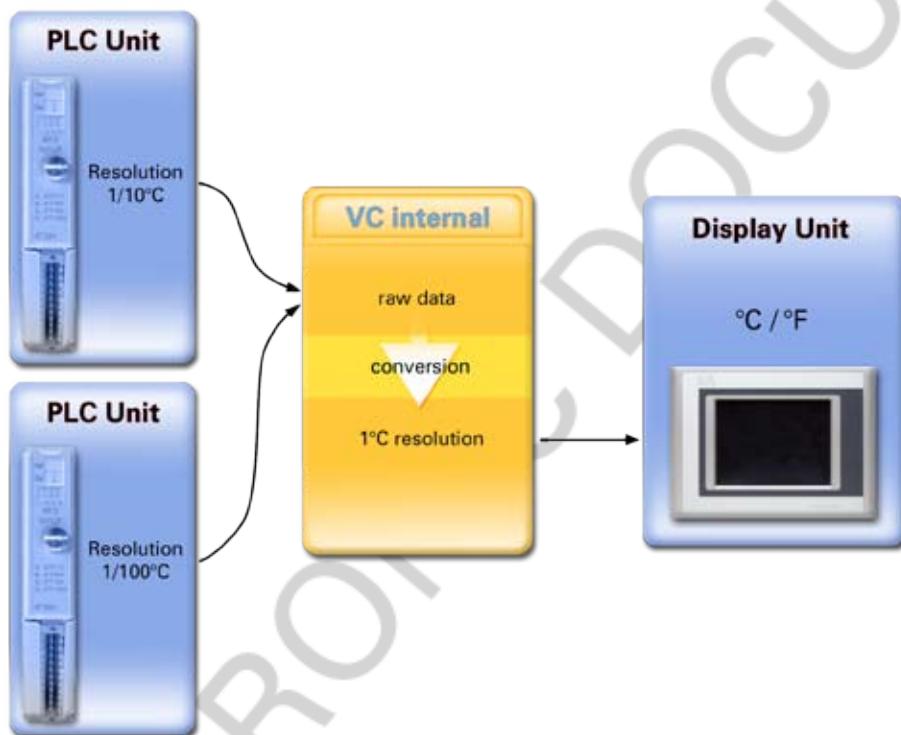


Fig. 22 单位转换

切换物理单位

对于温度变量，这意味着：

- 原始值以1/10度转换。在比例转换的第一步，将原始值转换成一个整数的度数
 \leftrightarrow PLCUnit。
- 摄氏度单位总是使用整数值——即1比1的比例——而华氏度单位也使用整数值，内部的摄氏值。
-

这就是说过程数据从1/10度缩放为整度数（根据相应的小数位），然后转化为最终的显示单位。

注意：

在比例缩放的第一步将原始值转化为VC-internal值，然后在第二步中转化所显示的值。

4.4.4 比例单位

在Temperatures单位组中的单位有如下的转换：

单位	单位转换	效果								
PLCUnit		将有1/10° C分辨率的原始值通过因子10转化为整数° C（比例缩放的第一步）								
Celsius		直接由° C转化而来，即使用1比1的转化。								
Fahrenheit	<p>公式： ${}^{\circ}F = \frac{9}{5} \cdot {}^{\circ}C + 32$</p> <table border="1"><tr><td>Value[0]</td><td>Internal: 0</td><td>Scaled: 32</td><td>基于 ° C.</td></tr><tr><td>Value[1]</td><td>Internal: 100</td><td>Scaled: 212</td><td></td></tr></table>	Value[0]	Internal: 0	Scaled: 32	基于 ° C.	Value[1]	Internal: 100	Scaled: 212		
Value[0]	Internal: 0	Scaled: 32	基于 ° C.							
Value[1]	Internal: 100	Scaled: 212								

4.5 连接单位组和一个数据点

如果需要将数据点显示为所需的单位，那么必须将单位组连接到一个数据点上。连接属性并正确地设置属性是单位切换或识别比例的唯一途径。

Data Source: Local					
Name	PLC-T...	VC-Typ..	Unit Group / Sub...	Limit	PLCUnit
GLOBAL					
panel					
ActTemperature	INT	SCALED	Temperatures	None	Default

下面的表格使用了三个例子，解释Temperatures单位组中的单位组属性和数据点属性的附属关系。

单位组属性	数据点属性	效果
Display Unit = Celsius PLCUnit = PLCUnit	PLC Unit = Default	在比例缩放的第一步是转换PLCUnit单位。如果单位切换没有连接数据点，那么在运行中就使用摄氏度。
Display Unit = Celsius PLCUnit = PLCUnit	PLC Unit = None	在比例缩放的第一步按比例1比1转换。如果单位切换没有连接数据点，那么在运行中就使用摄氏度。
DisplayUnit = Celsius PLCUnit = <None>	PLC Unit = PLCUnit	直接指定的PLC单位可以代替在单位组中指定的PLCUnit。

切换物理单位

4.6 简单的比例缩放

简易的比例缩放，对大多数应用已经足够了，可以代替高级的比例缩放。

与高级比例不同：

- 在比例缩放的第一步，VC按1比1的比例自动执行转换。对于用户只需要执行第二个步。
- 这些单位不一定要从整数值进行缩放，原始值也可以使用。

例子：简单的摄氏度单位组

在第一个应用中，我们需要一个摄氏度单位。为了显示数值，我们使用一个小数位。

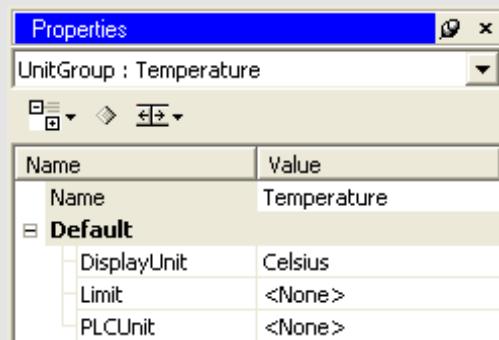
Index	Name	Unit Abbreviation	Unit Description	Default Precision
0	Celsius	°C	Celsius	1

图. 22 Celsius单位组

默认的精度设置为1。

单位组属性

Default – DisplayUnit属性可以设置为Celsius，这样可以以单位Celsius来显示数据点。



让我们详细地看一下摄氏单位：

我们用因子0.1转换控制器变量和显示器上的显示变量。

如果要指定factor，那么必须将Mode属性设置为Parameter。

结果

在运行过程中，将控制器变量127乘以0.1，考虑到精确度为一位小数点，因此以12.7显示。

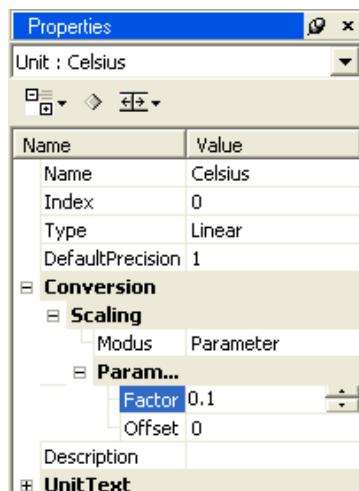


图. 23 单位属性对话框

注意：

根据国家设置在开发用的计算机上输入小数点，并且必须是一个逗号或小数点（句点）。

在Conversion Test区域执行一个测试。

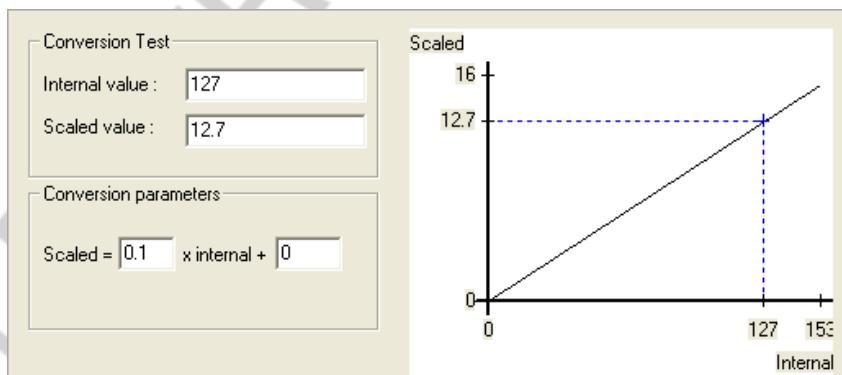


图. 24 转换测试

切换物理单位

4.6.1 只有一个单位的单位组

在运行过程中，为了以摄氏度来显示温度，必须将这个输入组与一个数据点连接。

Name	PLC-...	VC-Typ..	Unit Group / Sub...	Limit	PLCUnit
GLOBAL					
Language	UINT	INTEG...	None	None	None
ListboxIndex	UINT	INTEG...	None	None	None
PicNumber	UINT	INTEG...	None	None	None
SetTemperature1	INT	SCALED	Temperature	None	Default
SetTemperature2	INT	SCALED	Temperature	None	Default
SetTemperature3	INT	SCALED	Temperature	None	Default

图. 25 将单位组连接到一个数据点

将Celsius单位连接到数据点，需要两个步骤：

- 将VC类型转换为SCALED，有这样才可能实行转换。
- 如果单位没有和单位组属性下作为Default – PLCUnit来连接，那么<Default>或<None>都可以设置为PLC 单位。这是因为单位组的<Default>设置为<None>。
-

4.6.2 包含多个单位的单位组

这是指包含两个或多个单位的单位组。在运行过程中，切换单位是通过将单位索引值赋给单位数据点来实现的。

Index	Name	Unit Abbreviation	Unit Description	Default Precision
0	Celsius	°C	Celsius	1
1	Fahrenheit	°F	Fahrenheit	1

图. 26 单位组 - Celsius 和 Fahrenheit

让我们详细地看一下华氏温度：

转换的基础是分辨率为1/10摄氏度的整数值。

当在摄氏度↔华氏度之间进行转换时，可以使用一个因子或指定协调动作。

如果使用value pair来转换，就必须将Modus的属性设置为Values。

任务

开发者可以决定value pair在使用线性方程的属性中是如何执行的。

Conversion Test可以用于检验结果。

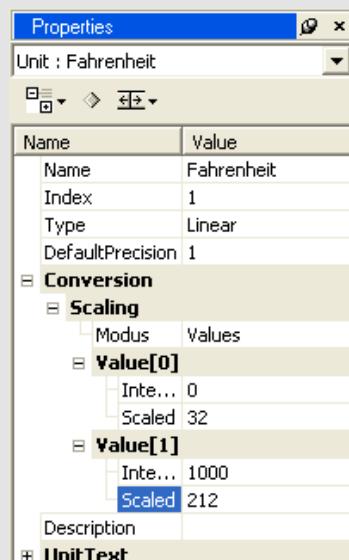


图. 28 单位属性对话框

与只有一个显示单位的单位组不同，在运行过程中，我们使用数据点在单位之间进行切换。

将这个数据点与单位的Runtime – DisplayUnit –

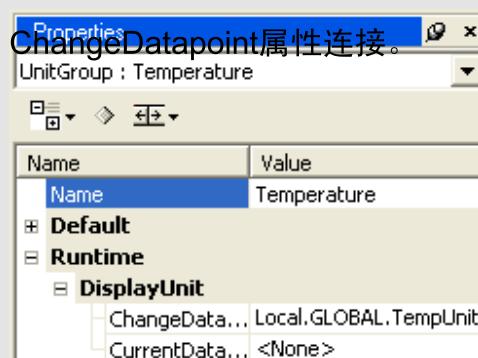


图. 28 Runtime – DisplayUnit

切换物理单位

如果在运行过程中将单位索引赋给连接到ChangeDatapoint属性的数据点，那么单位就会改变。

Index ▲	Name
◆ 0	Celsius
◆ 1	Fahrenheit

Value = 0 ↔ 用摄氏度显示

Value = 1 ↔ 用华氏度显示

5、单位切换实例

这一示例项目使用的是TM610培训模块中创建的可视化项目。

那个培训模块由几个任务组成，并且可以单独执行。

以前的章节中的单位切换和VC帮助可以作为执行这一任务的基础。

"通过实践来学习!"

在这些实例中包括以下的主题：

- 用整数值显示带小数位的值
- 从摄氏度切换到华氏度
-

单位切换实例

5.1 用小数显示一个数字值

在TM610的例子中，所有显示的值都没有小数位。为了显示有小数位的值，现在我们在Text and Values页面中提供三个带有小数点的输入区域SetTemperature1，SetTemperature2和SetTemperature3。

任务：创建一个单位组和单位



首先创建一个带有单位Celsius的单位组。数值会以0.1的因子缩小。

这个单位分配给相应的数据点。

Name	PLC-Typ.	VC-Typ.	Unit Group / Sub...	Limit	PLCUnit
GLOBAL					
Language	UINT	INTEG...	None	None	None
ListboxIndex	UINT	INTEG...	None	None	None
PicNumber	UINT	INTEG...	None	None	None
SetTemperature1	INT	SCALED	Temperature	None	Default
SetTemperature2	INT	SCALED	Temperature	None	Default
SetTemperature3	INT	SCALED	Temperature	None	Default

结果

输入值可以带有小数位。AS中的Watch窗口可以监控这个输入值。

5.2 从摄氏度切换到华氏度

为了执行单位切换，前面例子中创建的单位组必须扩展，包含第二个单位。

任务: 将华氏度添加到单位组

到现在为止，单位组只包含摄氏度单位。必须将单位组扩展为，包含第二个单位及其比例参数。

必要的步骤:

- 添加第二个单位，Fahrenheit
- 创建一个用于切换单位的变量
- 将数据点与单位组属性连接

结果

将单位切换到Fahrenheit的索引后，数值就以华氏度显示及输入了。

摄氏值0.0转化并显示为32.0。

5.2.1 转换精确度

比例缩放会导致输入值在转换时进位或舍位成下一个整数。

因为输入值在转换后会立即进行计算，所以可以正确的显示它，有可能是这种情况，输入值被整数化了。

例如:

输入值0华氏度有时得到的是-17.7。根据PLC变量的类型（INTEGER），这个值必须是整数。所以得到的结果是原始值-178。

在重新计算后，如果指定了两位小数位，那么将显示值-0.04。

单位切换实例

5.2.2 显示单位文本

单位切换的附加特征是定义单位文本及其缩写。

这些文本直接定义为单位属性。

UnitText	
Abbreviation	
Text	
English	°F
German	°F
Full	
Text	
English	fahrenheit
German	Fahrenheit

图. 29 单位文本

单位文本及其缩写形式可以在数字的输入和输出区显示。

优点

除了在数字区域显示单位文本以外，单位没有必要设置成文本。当单位切换时，单位文本也会自动的显示。

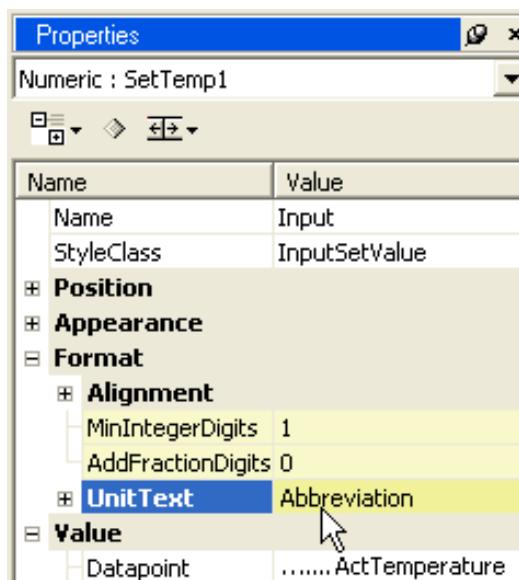


图. 30 带单位文本的输入区域

任务: 显示单位文本/缩写

缩写文本——° C或 ° F ——与三个输入区域一起显示。



6. 总结

现在我们讲的都是共同的语言。由于对简单例子和术语定义的学习，现在每个人都能设置复杂的语言和单位切换，以此来满足在可视组件的可视化应用中的需要。

如果还有遗留的问题，VC帮助文档会详细介绍这个主题：

可视组件：语言和单位切换

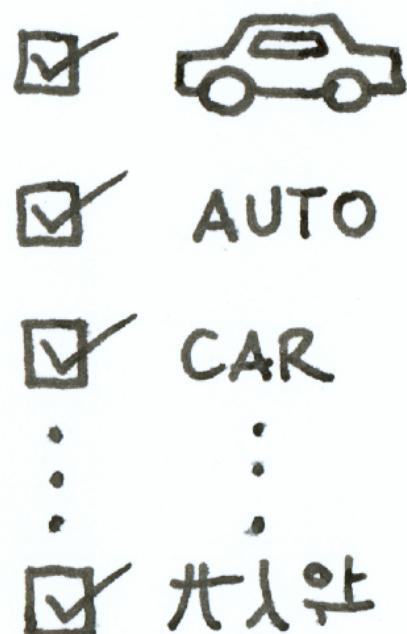


Fig. 31 可视组件国际化

总结

Notes

ELECTRONIC DOCUMENT

Notes

ELECTRONIC DOCUMENT

总结

培训模块综述

- | | |
|-------------------------------------|-------------------------------|
| TM200 – 贝加莱B&R 公司介绍** | TM600 – 图文显示的基础 |
| TM201 – 贝加莱B&R 产品系列** | TM601 – 贝加莱人机界面产品** |
| TM210 – Automation Studio™ 基础 | TM610 – ASiV 的基础 |
| TM211 – Automation Studio™ 在线通信 | TM620 – ASiV 的维护* |
| TM212 – 自动化对象 (Target) ** | TM630 – 图文显示的编程规则 |
| TM213 – 自动化运行 (Runtime) 系统 | TM640 – ASiV 报警系统 |
| TM220 – 维护信息* | TM650 – ASiV 的国际化操作 |
| TM221 – 自动化组件和出错信息查询* | TM660 – ASiV 的远程操作 |
| TM223 – Automation Studio™ 诊断 | TM670 – ASiV 高级应用 |
| TM230 – 结构化软件编程 | TM700 – Automation Net PVI |
| TM231 – 面向机器设备的Automation Studio™ * | TM701 – PVI 通信* |
| TM240 – 梯形图(LAD) | TM710 – PVI DLL 编程 |
| TM241 – 功能块图 (FBD)* | TM711 – PVI 的服务 |
| TM242 – 连续功能图 (CFC)* | TM712 – PVIControl.NET |
| TM243 – 顺序功能图 (SFC)* | TM720 – PVI 维护和诊断* |
| TM245 – 指令表 (IL)* | TM730 – PVI OPC |
| TM246 – 结构文本 (ST) | TM800 – APROL 系统概念 |
| TM247 – Automation Basic (AB)* | TM801 – APROL 工程设计基础 |
| TM248 – ANSI C | TM810 – APROL 安装, 配置和恢复* |
| TM250 – 内存管理和数据存储 | TM811 – APROL 运行(Runtime) 系统* |
| TM260 – Automation Studio™ 函数库I | TM812 – APROL 操作员管理 |
| TM261 – Automation Studio™ 函数库 II* | TM813 – APROL XML 查询* |
| TM264 – 定时处理单元 (TPU) * | TM814 – APROL 审计追踪* |
| TM400 – 运动控制的基础 | TM820 – APROL 维护* |
| TM401 – 贝加莱B&R 运动控制产品** | TM830 – APROL 项目工程设计 |
| TM402 – 运动控制系统的计算* | TM840 – APROL 参数管理和配方 |
| TM410 – ASiM 的基础 | TM850 – APROL 控制器配置和INA 通讯 |
| TM440 – ASiM的基本功能 | TM860 – APROL 库设计 |
| TM441 – ASiM多轴运动功能 | TM861 – APROL 通讯互联* |
| TM445 – ACOPOS ACP10 软件 | TM865 – APROL 库指导手册 |
| TM446 – 电子凸轮* | TM870 – APROL Python 编程* |
| TM447 – ACOPOS 智能过程技术 (SPT) * | TM880 – APROL 报表* |
| TM450 – ACOPOS 控制理念和控制器设置 | ** 查看产品目录 |
| TM460 – 启动B&R 电机* | * 即将出版 |

全球总部

Bernecker+Rainer Industrie-Elektronik Ges.m.b.H.

B&R Straße 1

A-5142 Eggelsberg 奥地利

Tel.: +43(0)7748/6586-0

Fax: +43(0)7748/6586-26

info@br-automation.com

www.br-automation.com

中国总部

贝加莱工业自动化（上海）有限公司

上海市漕宝路70号光大会展中心C座16楼

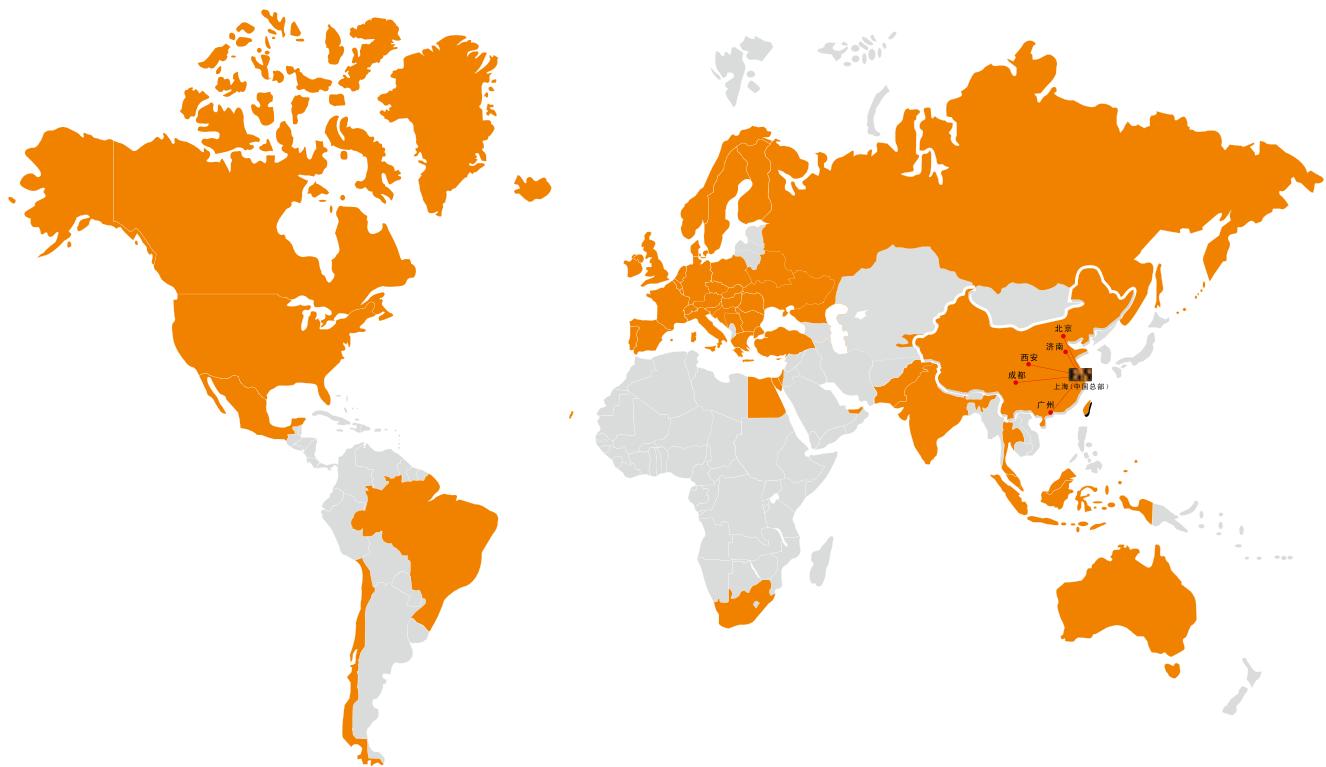
Tel.: +86/(0)21/6432 6000

Fax: +86/(0)21/6432 6108

info.cn@br-automation.com

www.br-automation.cn

全球50多个国家超过120个分支机构 www.br-automation.com/contact



中国总部



中国办事处

Austria · Australia · Belgium · Belarus · Brazil · Bulgaria · Canada · Chile · China · Croatia · Cyprus · Czech Republic · Denmark · Egypt · Emirates · Finland · France · Germany · Greece · Hungary · India · Indonesia · Ireland · Israel · Italy · Korea · Kyrgyzstan · Malaysia · Mexico · The Netherlands · Norway · Pakistan · Poland · Portugal · Romania · Russia · Singapore · Slovakia · Slovenia · South Africa · Spain · Sweden · Switzerland · Thailand · Turkey · Ukraine · United Kingdom · USA