

基于 Clips 的嵌入式专家系统开发方法

刘 昱 陈坚红 盛德仁 李 蔚

(浙江大学电厂热能动力及自动化研究所 浙江 杭州 310027)

摘 要 专家系统程序通常不是独立的,而是嵌入在其他的程序中。介绍了在 Windows 环境下利用 Clips 这种功能强大和应用广泛的专家系统开发工具结合 VC2005 开发基于 Clips 的具有图形界面的嵌入式专家系统的方法,并且给出了一个简单的专家系统的开发实例。着重于讨论 Clips 与 VC2005 的接口设计和具体实现,还介绍了一种算法来实现 Clips 对中文的支持。

关键词 专家系统 Clips VC 汉字处理

A METHOD TO BUILD EMBEDDED EXPERT SYSTEM BASED ON CLIPS

Liu Yu Chen Jianhong Sheng Deren Li Wei

(Institute of Power Plant Thermal Energy Engineering and Automation, Zhejiang University, Hangzhou 310027, Zhejiang, China)

Abstract Expert system is usually not independent but embedded in other applications. A method to build an embedded expert system with graphic user interface in Windows by VC2005 and Clips, a powerful and widely-used tool in expert system developing is presented. Then, a simple expert system developed by this method is provided. The design and implementation of the interface between Clips and VC2005 is discussed in detail, and an algorithm is presented to realize the function of Chinese process of clips.

Keywords Expert system Clips VC Chinese process

0 引 言

作为专家系统开发工具的一种,Clips(C Language Integrated Production System, C 语言集成产生式系统)的应用非常广泛,它是由美国航空航天局/约翰逊太空中心用 C 语言设计的,支持基于规则的、面向对象的和面向过程的开发方法。作为一个开放源代码的项目,每个人都可以从互联网上得到 Clips 的源代码,进行修改并且嵌入到自己的系统中去而无需付出任何费用。

Clips 支持三种专家系统的开发方法:简单的命令行模式、窗口模式和嵌入式模式。前两种方式都是在 Clips 环境下进行操作,开发出来的专家系统必须在 Clips 环境下进行使用,我们可以方便地在该环境下开发和调试专家系统的知识库和推理机,但是它的用户界面不够友好,所以直接使用的情况不多。最后一种方式是将 Clips 嵌入到其他程序编写的程序中去,在程序中使用 Clips 进行推理。因为 Clips 语言的强项在于强大的推理和表示能力,它可以简化我们设计专家系统时对于推理机的设计。所以说第三种开发方式即将 Clips 嵌入到其他程序中进行开发是一种最常用的方法,也是最通用的方法。

在目前的软件程序开发中,C++、C# 和 Java 是最流行和常用的开发语言。三者中,C# 和 Java 更适合 RAD (快速软件开发),开发效率都比 C++ 高。但是 C++ 开发出来的程序的运行效率却是最高,并且很多软件和设备都提供了和 C++ 语言的接口,更为关键的是 C++ 的编译器提供对 C 的代码的编译支持,这样我们就可以在 C++ 程序中直接使用 Clips。因此我们使用 Visual C++ 2005 作为我们的 C++ 开发工具,它是微软

公司的 C++ 开发环境的最新版本,不仅能够编译 C++ 和 C 代码,而且自带的 MFC 类库能够帮助我们快速地在 Windows 下进行软件开发。

因此,结合 VC2005 和 Clips 两者的优点,应用 Clips 强大的推理和表示功能开发专家系统的知识库和推理机部分,应用 VC2005 开发专家系统的人机交互界面,处理设备的数据交换和管理数据库,成为我们开发嵌入式专家系统的最佳选择之一。

1 Clips 嵌入 VC2005 开发的设计及实现方法

1.1 嵌入式专家系统的结构设计

考虑到 Clips 和 VC2005 各自的特点,本文设计嵌入式专家系统的结构如图 1 所示。

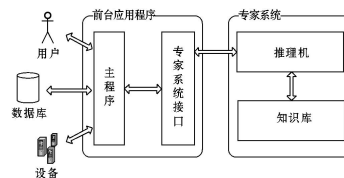


图 1 嵌入式专家系统结构

1.2 Clips 嵌入到 VC2005 工程中的实现

Clips 作为一个开源项目,我们在网上可以得到它的源代码和它的其它不同形式,在本文中我们将直接使用 Clips 源代码。

收稿日期:2006-04-18。刘昱,硕士生,主研领域:联合循环机组辅助决策系统。

再将代码包含在 VC2005 工程中以后,有两种方法使他们进行交互,第一种方式是直接在 C++ 代码中使用 Clips 提供的函数,对 Clips 进行操作,第二种是将 Clips 包含在某个类中,在类中封装 Clips 的函数,然后通过调用类的方法对 Clips 进行操作。两种方法的实现步骤大同小异,因为封装类的过程也是对 Clips 提供的函数的使用过程,只是使他更符合面向对象程序设计的要求,本文中阐述的重点在于将 Clips 怎样嵌入到 VC2005 的工程中,因此采用了第一种方式来详细说明 VC2005 和 Clips 通过函数调用进行交互的过程。

1.2.1 声明 Clips 头文件

让 VC2005 工程识别 Clips 并且使用 Clips 提供的函数,必须在工程中增加对 Clips 的声明,通常的做法是在“stdafx.h”的最后加入下列代码使 VC2005 可以调用 Clips 提供的函数。

```
extern C {
    #include [源代码所在文件夹]\clips.h
}
```

1.2.2 声明 VC2005 中的函数

VC2005 函数在能够被 Clips 识别和使用之前必须在 Clips 的 UserFunctions 中进行声明,这个函数在 main.c 文件中,main.c 文件是起到初始化 Clips 环境的作用,开发嵌入式的专家系统时,不需要初始化环境,必须把 main.c 删除或者注释掉。把 UserFunctions 复制到工程中的主 C++ 文件中,然后在该文件的头文件中包含如下声明

```
extern C void UserFunctions(void);
```

在 UserFunctions 中声明 VC2005 函数的语句如下

```
int DefineFunction( functionName, functionType, functionPointer, actual-
FunctionName);
```

第一个参数表示在 Clips 中调用该函数的函数名;第二个参数表示在 Clips 中函数的返回值,通常有“s”(字符串)、“w”(符号)、“i”(整型)等;第三个参数是指向 VC2005 中函数的指针,第四个参数是一个字符串,表示 VC2005 中的函数名。

也可以用 DefineFunction2 进行声明,它带有五个参数,前四个参数与 DefineFunction 相同,最后一个参数表示 Clips 允许传递的参数个数和类型。

1.2.3 Clips 传递参数到 VC2005

Clips 是通过上一节所述的方式调用 VC2005 中的函数,它又通过下列几个常用的函数把参数传给 VC2005 中的函数。

```
int RtnArgCount();
char * RtnLexeme( argumentPosition );
double RtnDouble( argumentPosition );
DATA_OBJECT * RtnUnknown( argumentPosition, &argument );
```

第一个函数的作用是在 VC2005 的函数中判断 Clips 调用该函数时传递的参数个数;第二到第四个函数分别返回第 argumentPosition 个参数的值,不同类型的参数必须使用相应的函数,如第二个返回字符串类型,第三个返回浮点数类型,最后一个返回的是一个通用类型,通用类型不能直接使用,必须转换为具体类型后才能使用。

1.2.4 Clips 获取 VC2005 函数的返回值

在 1.2.2 节的声明函数中给定了返回值的类型,而返回的值则根据类型用不同的函数实现。地址值可以直接返回。在实际运用中返回字符串、符号和数值是最常用的,它通过下列函数实现。

```
void * AddSymbol( string );
return string;
```

根据前面返回值的类型,Clips 会自动把返回的字符串转化为 Clips 的类型。

1.2.5 从 VC2005 中运行 Clips 进行推理

将 Clips 嵌入到 VC2005 工程中以后,我们必须在前台应用程序运行的时候对 Clips 进行初始化,初始化函数是 InitializeEnvironment()。通常我们可以把这个函数放置在 VC 工程的起始窗体的构造函数中,这样在应用程序初始化的同时也对 Clips 进行了初始化。初始化完成之后,通过 Load 函数加载知识库,Reset 函数重置环境,Run 函数运行,Clips 便可以根据知识库中的规则开始推理,推理过程中与前台应用程序的交互根据上述的函数调用实现。

2 基于 Clips 的嵌入式专家系统的中文支持

Clips 是用 ANSI C 开发的,最初是作为美国航空航天局/约翰逊太空中心开发专家系统的工具,目前最新的 Clips 版本是 6.2.3,它都是把美国标准信息交换码(ASCII 码)作为编码标准,即只支持普通字符串,而不支持能够表示汉字的宽字符集。

2.1 Clips 中文支持方法概述

在嵌入式专家系统中,Clips 因为它强大的推理和表示能力主要用于推理机和知识库的开发,与外部的数据交换则由前台的应用程序完成,因此要在 Clips 中增加对于汉字的支持有下面三种方法。

第一种方法是修改 Clips 程序的源代码。这个是解决问题的根本方法,要使 Clips 在嵌入式专家系统中能够处理汉字,可以只修改 Clips 负责类型管理和存储管理的相关文件,但是因为每个 Clips 文件都是由上千条的语句构成,因而工作量非常大,同时如果你对 C 语言和它的源代码的掌握没有达到一定程度,修改后版本的稳定性和可用性将大大降低。当然这个方法是解决该问题最佳对策。

第二种是使用数据库建立中英文对照词典。用户在使用前,将中文对照的英文输入词典中,在前台应用程序中使用输入的中文,经过词典翻译成相应的英文后进入 Clips 进行推理,同样 Clips 的结果在进入前台应用程序之前也由词典翻译成为对应的中文。这种方法的优点是比较简单,也是一种通用的方法。但是局限性很大,并且每次在 Clips 和前台应用程序交换数据前,要对数据库进行操作,开销比较大。

第三种方法将汉字转换成 Clips 可以识别的符号。它和第二种方法比较类似,都是通过将汉字转换为 Clips 能够识别的字符,不同的是第二种方法通过词典进行映射,而第三种方法是根据某种算法用一个符号来表示汉字,汉字和符号之间的转化是唯一并且可逆的。这种方法相对简单,系统资源的开销也不是很大,并且适用于后续版本 Clips。本文中我们通过这种方法实现 Clips 对中文的支持。

2.2 汉字和对应符号的转换算法

根据我国《信息交换用汉字编码字符集——基本集》(GB2312-80)规定“对任意一个图形字符都采用两个字节表示”,所以每个汉字都由两个字节组成,如果把组成一个汉字的 2 个字节即 16 位分成四段,每一段 4 位,这样每一段正好由 16 进制中的一个数字(0~F)表示,而 0~F 都是 ASCII 码中的符号。也就是说可以把一个汉字转换为 4 个 0~F 的符号,而一个

英文字母和数字是用 1 个字节表示,也可以转换为 2 个 0~F 的符号。例如汉字“汽”可以表示为“C6FB”,英文字母“a”可以表示为“61”,数字“1”可以表示为 31。虽然任何可打印的 ASCII 字符开头的符号都可以作为 Clips 数据类型符号(Symbol),但是为了和数字区别,建议在转换后的符号前面加上一个标示符以示区别。这样我们就得到了将任意由汉字、字母、数字和标点组成的符号转换为由 0~F 表示的符号的算法,在 VC2005 中的具体步骤如下:

1) 将输入的符号存储在字符串中,每次取字符串中的一个字符,在 C++ 语言中一个字符由 4 位构成。

2) 应用 sprintf 将该字符转换为一个用十六进制表示的字符 hexbuf。

3) 定义返回字符串 RtnValue = “CNHZ”(CNHZ 为标示符),在 RtnValue 后面加入 hexbuf。

4) 取一个新字符,重复 2)、3) 步骤,直至字符串转换完毕。

如果要把 Clips 中存储的由 0~F 组成的字符串转换成为原来的字符串只要实施上述算法的逆算法就可以了,但是在转换过程中要注意判断这个字符是汉字还是英文字母,所以算法稍微复杂,VC2005 中具体步骤如下:

1) 读取去掉标示符的欲转换字符串的前两个字符,分别存储在字符 chr1 和 chr2 中,chr1 和 chr2 共同构成了一个 8 位的字符。

2) 将 chr1 和 chr2 表示的十六进制数转换为一个字符 chrbuf,方法是 chrbuf = chr1 * 16 + chr2。

3) 得到 chrbuf 后要判断 chrbuf 是否小于十六进制数 80H (十进制 128),即判断这个字符是否是英文字母或者数字,如果小于 80H 表示它是一个英文字母或者数字,转换为 BYTE 数组后添加到返回字符串中。

4) 如果 chrbuf 大于 80H,继续读取字符串中的第三位和第四位到 chr3 和 chr4 中。根据步骤 2) 把它也转换为一个字符 chrbuf2。将 chrbuf 和 chrbuf2 同时转换为 BYTE 数组后添加到返回字符串。

5) 重复上述步骤,直至转换完毕。

通过上述算法我们成功地在嵌入式专家系统中实现把前台应用程序中由汉字、字母、数字和标点组成的符号转换成为 Clips 可以支持的由 0~F 组成的符号,或者进行反向变换,该方法相对来说比较简单,系统开销比较小,但是字符的长度会增加,增加了存储空间。因为符号的长度对 Clips 系统的效率影响很小,所以这种方法是利大于弊的。

3 简单应用

根据本文上述的 Clips 和 VC2005 结合开发嵌入式专家系统的方法和步骤,笔者开发了一个简单的具有学习功能的识别动物的专家系统,该专家系统不断地提问动物具有的属性,通过用户选择来判别动物,在给出答案以后让用户判断答案是否正确,如果答案不正确,用户可以输入正确答案和提示问题让专家系统学习,这样在下次判别的时候,专家系统就能够识别该动物。

该专家系统是 Clips 嵌入式开发的典型例子,下面着重举例介绍了 Clips 嵌入到 VC2005 中开发时怎样在他们之间传递数据这一关键的问题,并且表明怎样利用上述汉字处理算法来对

中文进行处理。用 Clips 开发专家系统的知识库和推理机,VC2005 开发的应用程序完成任务调度并处理界面和人机交互。为了接收从 VC2005 制作的人机交互界面中传送过来的用户的输入,在专家系统的知识库中我们定义了下面两个函数。

```
(defunction ask-question (? question ? type)
  (bind ? answer1 (cpp_GetValue ? question ? type))
  ? answer1)
(defunction bind-new-animal (? new ? old)
  (bind ? answer2 (cpp_newAnimal ? new ? old))
  ? answer2)
```

其中的 cpp_GetValue 和 cpp_newAnimal 即为用户自定义函数,用来接收用户在人机交互界面中输入的信息。函数 cpp_GetValue 用于接收用户点击了对话框中的是或者否按钮,并将结果返回给 Clips。cpp_newAnimal 用于接收对话框中用户输入的区别新动物的答案和问题。输入的问题和答案通过该函数返回到 Clips。

然后在 VC2005 的程序中加入下列语句声明这两个函数,让 VC2005 能够识别它们。

```
void UserFunctions() {
  DefineFunction ( cpp_GetValue , w , PTIF cpp_GetValue , cpp_GetValue );
  DefineFunction ( cpp_newAnimal , w , PTIF cpp_newAnimal , cpp_newAnimal );
}
```

最后在 VC2005 中可以具体地定义这两个函数的作用,因为本文的重点在于 Clips 和 VC2005 的集成,限于篇幅,具体的人机交互过程略去。

这样我们就完成了 Clips 与 VC2005 结合开发嵌入式专家系统的关键步骤,我们可以看到在开发过程中,clips 的数据库的设计和 VC2005 中程序的编制是紧密联系在一起,两者互相调用、互相影响的。

4 结 语

Clips 作为一个优秀的专家系统开发工具,它不仅具有强大的知识表示能力和推理能力,而且是一个开放源代码的软件,我们可以把它嵌入到自己项目中去,将它嵌入到 VC2005 软件开发的过程中后,又可以充分发挥 VC2005 擅长软件界面开发与外界进行数据交换优点,正好弥补了相互的缺点,发挥了各自的优点,因而运用 Clips 和 VC2005 是开发嵌入式专家系统的好搭档。本文重在阐述怎么结合 Clips 和 VC2005 开发专家系统的具体步骤以及一种 Clips 的汉字处理方法,这些方法和思路也可以用到相关的其它领域上。

参 考 文 献

- [1] Chris Culbert, Gary Riley, Brian Donnell Advanced Programming Guide [R/OL]. Clips Reference Manual 2005, II
- [2] Joseph Giarratano, Gary Riley 专家系统原理与编程 [M]. 北京:高等教育出版社, 2000.
- [3] Costin Badica, Amelia Badica, Valentin Litoiu Development Of An Embedded Expert System For Credit Evaluation [J/OL].
- [4] 王宇宙, 赵宗涛. CLIPS 中文应用开发环境设计与实现 [J/OL]. 中国期刊网, 2003, 2.