

使用 C++ 扩展 Python 功能的方法

李先觉

(中国石油大学(华东),山东青岛 266580)

摘要:人工智能技术发展、大数据广泛应用,使C++与Python混合后,能够实现更多功能,可缩短高性能语言使用中模型检测时间。将脚本存在问题更加方便的解决,使人们在生活或者工作中能够将任务顺利开展,为人类带来更多方便。

关键词:C++;Python功能;方法

中图分类号:TP312

文献标识码:A

文章编号:1007-9416(2018)05-0168-02

在应用软件中常见功能需求相同现象,只有少数功能要求具有较大差异性,所以针对需求变化构建独立版本,会使未来工作量提升。采用脚本语言编写程序虽然在一定程度上步入系统编程语言,但是将代码修改后,能够使配置运行更加方便,这也是系统语言所不具备的能力。在系统编程语言中能够将用户平台修改,提供二次开发平台。混合编程为一种新型编程语言,能够在不同领域广泛应用,将C++与Python语言功能融合,能够满足更多用户需求,因此值得实践研究。

1 C++与Python混合语言编程方式

1.1 python语言分析

python语言为一种解释性、交互性、动态性语言脚本,在二十世纪末期衍生,经过多年发展后,已经成为世界范围内应用最广泛的跨平台语言脚本。具有免费开源、可移植性、可嵌入性等特点。

1.2 免费开源

Python为FLOSS自由开发软件源码之一,人们可以随意发布此软件,将代码阅读。并做出相应的改动,使其一部分完全应用于软件中。人们在使用python语言编写程序过程中。不需要考虑通过怎样方式将底层细节处理。

1.3 可移植性

Python自身本质问题,能够使其在不同平台上构建软件,其中包括Windows、MAC、Pocket PC等。Python能够提供类的继承等属性,将异常情况做到有效处理,选择最佳支持方法。

1.4 可嵌入性

正是由于python具备的可嵌入性,才能融入C++程序中,使程序脚本功能更加强大。若希望使关键代码运行速度更加迅速,可以将部分程序采取C++编写,然后融入到python程序重新使用。

2 C++扩展Python功能的方法

拓展以及嵌入能够使C++与python成为一个新的集合体。拓展方式需要将C++系统语言将python模块拓展,并在python中将模块功能利用。嵌入则是需要将python解释器融入于应用程序,使程序能够更好将python语言脚本解释。拓展以及嵌入需利用python中C语言应用程序作为接口。C与python间交互方式为数据格式转换方式。

若系统为C++开发主系统,一般会采取两种方式命令python脚本。可以通过系统命令调用以及嵌入解释器直接调用方式。第一种方式将参数传递只可采取命令方式,为字符串方式,功能具有局限性。第二种方式将python内置高级数据系统,可以将不同复杂程序传输。并且第一种方式在系统运行过程中对python解释器有较高要

求,需要将python安装实现,在产品升级过程中不可实现。第二种方式则只需要提供动态数据库以及压缩包即可,程序部署较为简单,能够使产品更加方便升级。

所以若想将C++将python功能拓展,实现混合编程。主系统可采取动态库导出函数方式直接使用,做到无缝衔接。

2.1 python解释器

将C++应用于python功能中,需要装设python解释器。具体步骤为:构建python运行环境,将python解释器功能初始化,对系统工作路径进行设置,将设定脚本导入其中,将参数输入实现数据转换,调用函数,对返回值进行检查,将所返回值进行转换,此解释器构建完毕。

2.2 python运行环境要求

相对于不同版本python应用程序,不要不同python动态库,常用python版本中含有Release版本python动态库,而Debug版python动态库则需要自行设定python源码。将动态库排除,需要将所用python库压缩成为一个zip文件,备用。在此环节工作中可以采取py2exe软件打包,或者采取直接压缩目录完成。

2.3 将python解释器初始化

将pythonC语言中API函数Py_Initialize可以将python解释器初始化完成。

2.4 构建系统工作路径

将python中C语言API函数PyRun_SimpleString调用后,将系统路径设定,为后续脚本运行做好基础。

2.5 将自定义脚本导入

将python的C语言API函数PyImport_ImportModule应用与自定义python脚本中,其中存在所有使用代码。

2.6 将参数数据转换

将上述调用的pythonC语言API函数Py_BuildValue C++数据输入参数转化为python能够接受的参数。此环节也是整个环节中工作量最大的部分,也是环节中最为重要的部分,贯穿于整个环节,对最终效果能够产生影响。

2.7 函数应用

将python中C语言API函数PyObject_CallObject应用于自定义脚本中,并将控制权交付于python解释器中,等到python脚本函数工作结束后返回,对PyObject_CallObject函数返回值做出检测,根据预先设定,将返回值检测,对结果进行处理。

2.7.1 返回值数据转换

根据返回值将结果转换为++数据类型。根据实际编码将任务执行一次。在其他步骤中将应用程序调用并在python脚本执行,后

收稿日期:2018-03-22

作者简介:李先觉(1998—),男,汉族,江苏涟水人,中国石油大学(华东)化工院本科在读,研究方向:环保设备工程。



续操作需要判断py-ISInitiazed是否实现初始化,防止出现重复初始。其他步骤需要在调用嵌入式python脚本执行。

2.7.2 数据转换

根据上述方式,C++与python混合编程工作核心为实现数据在C++以及python环境中能够良好传递。C++语言类型为整型、浮点型以及字符串型,C++中结构体数据类型涉及字符对其问题,若直接转换到python中并不方便处理,所以采取方式一般是转化为py-thon词典数据然后再处理。

将C++数据类型转换为python数据类型后,采取调用python中C语言API函数py-BuildValue方式将程序实现混合编写。常规情况下将python数据类型转化为C++类型后,能够使嵌入式python应用,在调用python脚本时返回。由于返回值类型为pyObject指针类型,所以需判断py-Check函数类型,结果检验后若无问题可以做函数转换。

在脚本语言混合过程中需要注意计数问题。由于py-thon垃圾收集根据对象引用计数实现,所以对象不能增加或减少计数,那么便会出现内存泄露问题,最终致使系统崩溃。

在python的C语言API中会提供两个引用加减的计数,在引用计数为0时自动将内存释放。在此过程中一般会采取拥有或者借用方式,对象拥有者不需要使用此对象时,需实现调用py DECREF。在

对象借用过程中借用方不能随意调用py DECREF。借用方式能够不需注意py DECREF何时被调用,也不用担心内存泄露问题,但是缺点较为明显,在一些特殊情形下,若拥有者调用py DECREF后使用此对象,便会出现野指针问题,所以在python的C语言API中,对象引用函数在传输过程中,拥有权不会调换,具有较高安全性。

3 结语

将C++拓展python功能,能够实现混合式开发,将产品更加方便的审计,并且不依赖python环境,并且脚本具有语言高效灵活特点,可以将程序功能完善。Python作为一种能力较强语言脚本,开发较为方便,所以在运行速度较慢时,将C++程序语言补充,能够使系统更具灵活性。

参考文献

- [1]陈婕.基于C++和MatLab混合编程的GM(1,1)模型实现与应用[J].电脑知识与技术,2017,13(33):124-126.
- [2]杨卫.面向深度学习扩展算法的编译优化方法[D].吉林大学,2017.
- [3]陶诚,陆从珍.基于C++和Python混合编程的WORD文档操作方法[J].信息化研究,2014,40(05):58-63.

The Method of Using C++ to Extend the Python Function

LI Xian-jue

(China University of Petroleum,Qingdao Shandong 266580)

Abstract:The development of artificial intelligence technology and the wide application of large data make C++ and Python mixed, can achieve more functions, and can shorten the time of model detection in the use of high performance language. The problem of script is more convenient to solve, so that people can carry out tasks smoothly in life or work, and bring more convenience to mankind.

Key words:C++; Python function; method

.....上接第167页

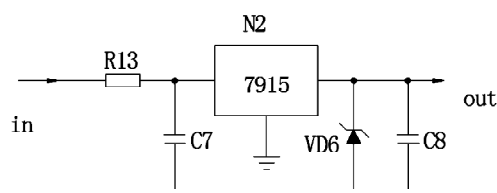


图5 G类电路设计图

能和高频干扰的影响,灵活运用屏蔽、滤波和接地等技术措施,减少传导干扰和辐射干扰,以保证两者之间信息交互的稳定性和可靠

性。此外对外接口电路的设计还应考虑电路和外部设备的电源接入方式,并为发射机和外部信号之间提供隔离和保护,同时为外部电缆的瞬态电压或意外不适当的电压提供保护。

对外接口电路是实现发射机自动监控不可或缺的重要组成部分,本文虽是以XAM-10A型中波广播发射机监测控制系统为对象介绍和解析其对外接口电路的设计方法,其他类型中波广播发射机尽管操作方式、故障种类和状态参量不尽相同,但其对外接口电路的设计方法和原理是基本一致的,因此在其他机型的对外接口电路设计中也可以进行参考。

Design and Implementation of External Interface Circuit in the Monitoring and Control System of MW Transmitter

ZHOU Xin-guo

(582 Station of State Administration of Radio and Television,Beijing 100050)

Abstract:The external interface circuit is a bridge and window for the transmitter to interact with the external software and hardware system equipment. It is an indispensable part of the automatic transmitter monitoring. In this paper, the design and implementation of the external interface circuit of the XAM-10A type medium wave broadcasting transmitter is discussed. The design and analysis of the key circuit of the external interface circuit is completed according to the status parameters involved in the transmitter and the type of the external input / output characteristic of the instruction information. The design method can be used as a reference for design of external interface circuit of the related model monitoring and control system.

Key words:external interface circuit;input/output;external interlocking;monitoring & control