

编译器的云计算模型设计与应用分析

潘峰^{1,2}, 林龄^{1,3}

(1. 贵州民族大学 模式识别与智能系统省级重点实验室, 贵州 贵阳 550025;

2. 贵州民族大学 信息工程学院, 贵州 贵阳 550025;

3. 贵州民族大学 理学院, 贵州 贵阳 550025)

摘要:为了解决当前许多编译工具只能在单机上使用的局限,提出了一个在线共享使用的网络化编译环境的设计方案。方案采用B/S体系结构,在服务器端使用中间组件将Web服务器与编译器等软件工具无缝集成,用户与系统间交互主要采用I/O重定向技术实现。编译程序时通过设计网络中间组件调用服务器端的编译器处理用户提交的程序,并采用I/O重定向方法实现程序运行时的输入与输出。该模型实现了编译工具的网络共享,为Internet用户提供了一种随时随地的在线式编译服务,可作为Online Judge系统的支撑框架。实验结果表明,该设计方案是有效可行的。

关键词:云计算模型编译器;中间组件;I/O重定向

中图分类号:TP301

文献标识码:A

文章编号:1673-629X(2014)08-0187-04

doi:10.3969/j.issn.1673-629X.2014.08.044

Design and Application Analysis of Cloud Computing Model of Compiler

PAN Feng^{1,2}, LIN Ling^{1,3}

(1. Key Lab of Pattern Recognition & Intelligent System, Guizhou Minzu University, Guiyang 550025, China;

2. School of Information Engineering, Guizhou Minzu University, Guiyang 550025, China;

3. School of Science, Guizhou Minzu University, Guiyang 550025, China)

Abstract: In order to solve the current limitation which many compilation tools can only be used on single, propose a design scheme of sharing online using the network of the compiler. This scheme adopts B/S architecture, Web server and the compilation tools are integrated as an environment with the intermediate components in the server side, interaction of user and the system mainly uses the I/O redirection technology. When compiling the program, design the network intermediate components to call compiler to process source program, and I/O redirection method is adopted to realize the input and output of running program. The model implements network sharing of the compiler tools, provides Internet users with an on-line compilation services anytime and anywhere, and is used as a support framework of Online Judge system. The experimental results show that the method is effective and feasible.

Key words: cloud computing model compiler; intermediate component; I/O redirection

0 引言

Internet的快速发展与应用的不断深入不仅改变了软件的设计模式,同时也改变了软件的应用模式,基于Internet、基于Web的软件系统已成为未来软件发展的主要研究方向^[1]。以编译器为代表的工具软件是进行软件开发的重要工具,当前它主要被集成在功能强大的集成开发环境(Integrated Development Environ-

ment, IDE)中。Visual Studio. NET、NetBeans、Sun Studio和Xcode等都是常见的集成开发环境,它们集编辑、编译、调试以及部署为一体,是桌面系统中进行软件开发的强大工具。这些IDE尽管功能强大,但也很复杂,而且只能在单机桌面系统中使用,不能多用户共享;同时,对于仅仅需要编译器的程序设计者和计算机科学研究者而言,IDE的冗余功能太多。文献[2-13]

收稿日期:2013-10-21

修回日期:2014-01-23

网络出版时间:2014-05-21

基金项目:贵州省科学技术联合基金(黔科合J字LKM[2011]18号,黔科合J字LKM[2013]20号);贵州民族大学教改项目(GUM2013JG18)

作者简介:潘峰(1974-),男,贵州瓮安人,副教授,CCF高级会员,研究方向为计算机软件、Web工程。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20140525.1242.013.html>

都是基于 Web 构建的软件系统,实现了多用户的在线共享,使用方便,无需维护。文中是将传统的桌面 IDE 实现网络化,进行精简,提炼出主要功能进行重新架构,构建一个能够网络共享,又能满足用户需要的在线式编译环境(Online Compiling Environment),实现用户随时随地的使用。

1 系统模型设计

1.1 系统建模

为实现在线式编译环境的用户方便性,使用基于 Web 的架构,它能使用户随时随地地使用编译环境。这种分布式结构的系统需要多台计算机协同工作,其工作流程要比单机结构复杂。文中以开发 Java 程序为例,使用 Petri 网分别实现单机模式和在线模式的系统建模。

1.1.1 单机模式流程

在单机模式中,程序的编辑、编译、保存及运行都在同一个操作系统中处理。编译时可能出现编译时错误,它作为标准错误显示在命令行窗口;运行时可能出现标准错误或标准输出,也显示在命令行窗口,运行时的数据可以来自命令行输入或数据文件。其类 Petri 网描述流程如图 1 所示。

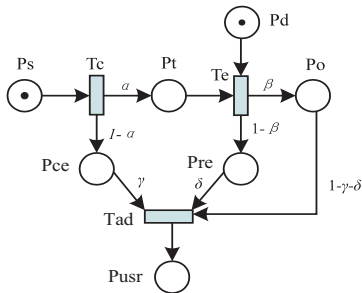


图 1 单机模式工作流程

Ps 表示源程序,经过编译器 Tc 编译,可出现两种情况:若 Ps 正确,则产生字节码 Pt,否则出现标准错误 Pce。Pd 表示运行时输入,Pt 被解释器 Te 解释执行,可能出现运行时错误 Pre 或标准输出 Po。Tad 代表汇集,它以 Pce 或 Pre 或 Po 中的一个为最终结果 Pusr。

1.1.2 在线模式流程

在线模式是单机模式的扩展,它以 Pw 页面为用户输入,提交给 Web 服务器 Tsa 后调用中间组件 Tm 处理,在客户端生成源文件 Ps 和运行时的数据输入文件 Pd(从 Ps, Pd 到 Pusr 与单机模式相同),最终结果 Pusr 由 Web 服务器 Tsb 返回客户端,如图 2 所示。

1.2 功能模块设计

用户在使用在线编译系统时,需要进行三个基本操作,即提交程序、录入数据和输入命令行。为此,可以设计三个页面和对应的处理组件实现。

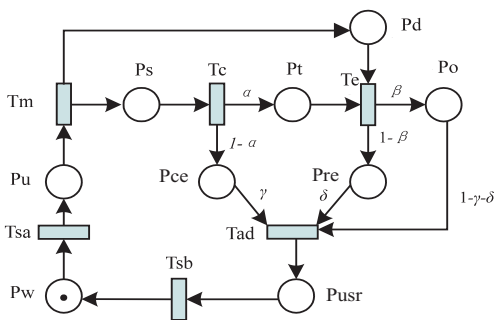


图 2 在线模式工作流程

1.2.1 用户界面

用户界面主要设计三个 Web 页面:提交程序页面、提交运行时数据页面和输入 Web 命令行页面。提交程序页面主要由一个多行文本区域控件、一个单行文本区域控件和一个提交按钮构成,多行文本区域控件用于编辑源程序,单行文本区域控件用于输入源程序的文件名,提交后将在服务器端存储源程序文件;提交运行时数据页面与提交程序页面类似,它的前一个文本区域控件用于输入数据,后一个用于输入数据文件名,其中输入的数据间使用空格隔开,提交后将在服务器端存储源数据文件;Web 命令行页面只需要一个文本区域控件,用于输入字符串格式命令行,该命令由网络中间组件在服务器端调用 Shell 执行。

1.2.2 中间组件

中间组件是沟通客户端与服务器端使用工具的中介。提交程序页面和提交运行时数据页面调用同一个中间组件,该中间组件被 Web 服务器调用,用于读取客户端的数据(源程序或运行时所需数据),并写入到文件中。处理 Web 命令行页面的中间组件只需读出输入的命令行,作为函数 system 的参数,该中间组件运行的结果是一动态生成的 HTML 文件,其中包含运行结果文件的超链接语句。

1.3 关键技术

在线编译环境采用浏览器/服务器体系结构,采用 Web 页面作为输入输出的界面。它需要解决三个基本问题:即数据的读取、编译工具的调用及运行结果的返回。用户数据的提交通过 Web Form 传递到服务器端,Web 服务器调用中间组件读取数据,并写入服务器端的文件中;编译器的调用在中间组件中通过系统函数 system(参数)实现,参数是字符串格式的编译命令行;运行时采用 I/O 重定向语句组成字符串参数,由 system 函数执行,从而实现程序运行时的输入与输出。

2 在线 Java 编译环境设计

2.1 用户界面与中间组件设计

提交程序页面的关键代码如下:

```
<form action="remoteSave.ch" method="POST">
```

```
<textarea name = “CODE”></textarea>
<input type = “text” name = “FILENAME”>
<input type = “submit”>
</form>
```

上述代码中 remoteSava.ch 为处理用户提交程序的中间组件,它是采用 C 语言编写的动态网页,在 Ch^[14]解释计算环境中执行。运行时提交数据的页面同样使用中间组件 remoteSava.ch 处理,其主要功能是获取提交内容,并写入到文件中。remoteSava.ch 的关键代码为:

```
char * codeString=request.getForm(“CODE”);
char * filename=request.getForm(“FILENAME”);
File * ft;
if(ft=open(filename)!=NULL)
fprintf(ft,codeString);
else return ERROR;
```

Web 命令行页面设计和提交程序页面类似,主要代码为:

```
<form action = “remoteCmd.ch” method = “POST”>
<input type = “text” name = “CmdString”>
<input type = “submit”>
</form>
```

上述代码中 remoteCmd.ch 组件调用 system(char * string)函数执行提交的命令行。若编译 Java 源程序,输入命令行 string 为“javac Test.java 2>CError.txt”;运行时输入命令行 string 为“java Test<input.data>RResult.txt 2>&1”。中间组件 remoteCmd.ch 的关键代码为:

```
char * cmdString=request.getForm(“CmdString”);
system(cmdString);
printf(“<a href=\\“CError.txt\\”>Compiled result</a>”);
printf(“<a href=\\“RResult.txt\\”>Executed result</a>”)
```

2.2 Online Judge 系统的原型设计

Online Judge 系统简称 OJ 系统,它是 ACM 组织竞赛的评判系统,也是训练学生学习编程的重要辅助系统,是在线编译环境的典型应用。在线编译环境是 Online Judge 系统的支撑环境,只需在此基础上设计一个编程问题的实例集合,即可构成 OJ 系统。

一个 OJ 系统的实例主要包括四个部分:问题描述、测试数据、标准输出数据、评判程序。

2.2.1 问题设计

它首先采用文字方式准确描述问题,每一个问题对应一个 ID;同时每个问题设计若干组测试数据和若干组标准输出数据,测试数据与标准输出数据是一一对应的。在提交程序正确通过编译的情况下,每一组测试数据作为程序的输入都产生相应的结果,然后判定程序逐一比较输出结果与标准输出数据,从而给出评判的结果。

2.2.2 OJ 中间件设计

OJ 中间件是一个处理中心,在其中集成一系列子操作,对于处理 Java 程序的算法为:

Step 1:读取源程序,并写入服务器端的文件 A 中;

Step 2:调用编译器编译文件 A,将编译时产生的提示重定向到文件 B;

Step 3:若文件 B 非空,返回程序错误信息;

Step 4:调用解释器解释运行字节码文件 C,将运行时产生的输出结果或产生的错误提示重定向到文件 E(由于有 N 组测试数据,所以产生 N 个文件 E,不妨记为 E_{1~N});

Step 5:调用判定程序读取文件 B、E_{1~N},给出评定。

3 实验结果与讨论

3.1 实验配置

在线式编译环境所在的服务器端硬件为:CPU Intel Core 8300, RAM 4 GB。软件:操作系统为 Windows 7;编译工具为 JDK1.7;Web 服务器使用 IIS6.0;动态网页技术使用 Ch 解释计算环境为中间组件的执行引擎。客户端浏览器为 Internet Explorer 9。

3.2 实验结果

以输入十个整数顺序排序为例,来观察实验结果。测试程序源文件为 Java 程序,文件名为 Test.java,将输出语句末尾的分号去掉来观察编译提示。表 1 中分别是程序有语法错误、程序正常运行和 OJ 系统测试的三种输出结果。

表 1 三种形式的实验结果

情况分类	显示结果
程序有语法错误	Test.java:10:需要‘;’
	System.out.print(num[i]+“ ”) 1 错误
程序正常运行	1 2 3 4 5 6 7 8 9 10
OJ 系统判定	Passed!

在上表中,程序正常运行时所需要的 10 个数据可以通过表单中的文本框按空格隔开的格式同时输入,把它作为一个字符串写入服务器端某个文本文件,然后使用重定向提供给运行程序使用;对于在线评定(OJ)问题,判定的程序读入的数据是系统事先设计好的,它们是用来测试程序是否能正确处理,得到结果是否符合标准,以此判定用户编写的程序是否正确。

3.3 讨论

在单机模式、在线模式及 OJ 系统三种情形中,程序运行的方法有很多不同之处。主要分析它的输入/输出的不同,其情形如表 2 所示。

表 2 三种模式程序运行时的输入/输出比较

模式	输入	输出
单机模式	标准输入	标准输出
在线模式	动态文件	动态文件
OJ 系统	静态文件	动态数据

在单机模式中,程序运行在命令行窗口。少量的数据输入通过标准输入设备键盘录入,结果通过标准输出设备屏幕显示。对于数据量比较大的输入,可预先录入文件,运行时采用输入重定向到文件获取数据。同样,需要保存输出结果时,使用输出重定向到文件的方法即可实现。

在线模式中,程序运行在远程服务器端,输入数据需要从客户端(如 Web Form)提交,并存储在服务器端文件中,该文件是中间组件运行时动态生成的。程序运行时使用重定向输入方法获取数据,标准输出也重定向到一个文本文件,该文本文件的 URL 返回到客户端,即是在 Web 客户端看到的是程序运行的结果。

OJ 系统中,处理每个问题的程序如果编译通过,运行时的数据来自系统设计时预置的数据文件,通常有 1-N 组,它们是预先设计好的,不可更改,所以称之为静态文件。它们通过输入重定向传给程序,同时分别将输出结果重定向到 1-N 个文本文件。最终的输出结果由判定程序比对预置的 1-N 组标准输出与运行时产生的 1-N 个文本文件内容,从而给出评判结果。这种反馈的结果不是程序运行结果,而是有限个可能出现的结果标志,它们由系统设计者制定。

4 结束语

在线编译环境实现了软件的网络化和服务化。网络化特性为用户提供了即上网即使用的方便性,实现了多用户共享使用,提高了软件的利用率。服务化特性体现在使用模式上,它是一种 SaaS(Software as a Service)模式的云计算实现方式,用户可以按需使用,按时付费,不需要安装和维护。

参考文献:

[1] Wilz Sr D M, Knowles C H. Web-based system and method

(上接第 186 页)

[12] 周树德,孙增圻. 分布估计算法综述[J]. 自动化学报, 2007,33(2):113-121.
[13] 樊 玮,苏秋波. 基于分布估计算法的多航段座位分配模型[J]. 信息与控制,2012,41(6):774-778.
[14] 高 尚. 武器-目标分配问题的分布估计算法及参数设计[J]. 东南大学学报(自然科学版),2012,42(S1):178-

for enabling a viewer to access and display HTML-encoded documents located on the World Wide Web (WWW) by reading URL-encoded bar code symbols printed on a web-based information resource guide; U. S. ,6,076,733[P]. 2000-06-20.

[2] Murugesan S. Web application development: challenges and the role of web engineering[M]//Web engineering: modelling and implementing web applications. Berlin:Springer,2008:7-32.
[3] Han M C, Li D Y, Liu C Y, et al. Networked characteristics in software and its contribution to software quality[J]. Computer Engineering and Applications,2006,42(20):29-31.
[4] He Keqing, Wang Jian, Liang Peng. Semantic interoperability aggregation in service requirements refinement[J]. Journal of Computer Science and Technology,2010,25(6):1103-1117.
[5] He Keqing, Peng Rong, Liu Jing, et al. Design methodology of networked software evolution growth based on software patterns[J]. Journal of System Science and Complexity,2006,19(2):157-181.
[6] 潘 峰,席 泓,张儒良. 一种分布式编译平台的设计与实现[J]. 西南师范大学学报:自然科学版,2010,35(4):141-144.
[7] 王 腾,姚丹霖. Online Judge 系统的设计开发[J]. 计算机应用与软件,2006,23(12):129-130.
[8] 喻擎苍,翁秀娟,赵 匀,等. 交互式开放结构计算机视觉平台 Ch OpenCV[J]. 计算机工程与应用,2006,42(23):78-81.
[9] 赵剑冬,林 健. MATLAB 在基于 Web 的决策支持系统中的应用研究[J]. 计算机应用研究,2007,24(12):260-262.
[10] 陈文鑫,项剑波,陈军敢. 基于 Web 的 MATLAB 远程命令窗口的设计与实现[J]. 计算机应用与软件,2009,26(2):138-141.
[11] 潘 峰. 绘图脚本自动生成器的设计及其应用[J]. 计算机技术与发展,2010,20(10):74-77.
[12] 郭秀娟. 基于 B/S 模式的地下水资源管理系统设计[J]. 计算机技术与发展,2011,21(12):198-200.
[13] 李养群. 基于 Web 的 EPON 业务配置管理系统设计与实现[J]. 计算机技术与发展,2012,22(4):153-156.
[14] SoftIntegration Inc. Ch CGI user's guide[EB/OL]. 2009. http://www.softintegration.com/.

181.

[15] Pelikan M, Godberg D E, Cantu-Paz E. Linkage problem, distribution estimation, and Bayesian networks[J]. Evolutionary Computation,2000,8(3):311-340.
[16] 何大勇,鄂明成,查建中,等. 基于空间分解的集装箱布局启发式算法及布局空间利用率规律[J]. 计算机辅助设计与图形学学报,2000,12(5):367-370.

编译器的云计算模型设计与应用分析

作者:

潘峰, 林龄, [PAN Feng](#), [LIN Ling](#)

作者单位:

[潘峰, PAN Feng \(贵州民族大学 模式识别与智能系统省级重点实验室, 贵州 贵阳 550025; 贵州民族大学 信息工程学院, 贵州 贵阳 550025\), 林龄, LIN Ling \(贵州民族大学 模式识别与智能系统省级重点实验室, 贵州 贵阳 550025; 贵州民族大学 理学院, 贵州 贵阳 550025\)](#)

刊名:

[计算机技术与发展](#)

英文刊名:

[Computer Technology and Development](#)

年, 卷(期):

2014(8)

本文链接: http://d.wanfangdata.com.cn/Periodical_wjz201408044.aspx