

基于云环境的持续集成

姜文,刘立康

(西安电子科技大学 通信工程学院,陕西 西安 710071)

摘要:随着云计算技术的快速发展,软件开发的许多环节都在云环境中进行。持续集成作为软件开发过程中的重要组成部分,也需要在云环境中进行集成构建。为了在软件开发过程中正确应用这些新技术,需要开展这方面的研究工作。结合工作实践,以SVN作为配置管理工具,介绍了基于实体机持续集成的特点和运行机制。详细叙述了基于云环境的持续集成,包括在物理实体机上构建私有云平台;基于部门云持续集成的运行原理、部署和特点;基于公司云持续集成的运行原理、部署和特点;基于公有云持续集成的构建过程和特点;基于云环境的持续集成工程运行过程。最后介绍了一个典型的持续集成工程案例。工作实践表明在云环境中开展持续集成工作,可以节约计算机软硬件资源,提高软件开发效率,降低软件开发成本。

关键词:云环境;持续集成;部门云;公司云;公有云;私有云

中图分类号:TP311.5

文献标识码:A

文章编号:1673-629X(2018)01-0011-06

doi:10.3969/j.issn.1673-629X.2018.01.003

Continuous Integration Based on Cloud Environment

JIANG Wen, LIU Li-kang

(School of Telecommunication Engineering, Xidian University, Xi'an 710071, China)

Abstract: With the rapid development of the cloud computing, many links are proceeding in the cloud environment during the software development. The continuous integration as the important component in software development also needs to do the integration building in the cloud environment. In order to correctly use the new technology, the research in this field is needed to be done. In the combination of work practice, taking SVN as the configuration management tool, the characteristics and operation mechanism of continuous integration based on physical machine are introduced. The continuous integration based on cloud environment is described in detail, including the private cloud platform built in physical machine, the operating principle, deployment and characteristics of continuous integration based on department cloud and based on corporation cloud, the building process and characteristics of continuous integration based on the public cloud, as well as the operational process of continuous integration based on cloud environment. At last, a typical project case of continuous integration is introduced. Working practice shows that the continuous integration in cloud environment can economize the resource of computer hardware, enhances the development efficiency of software, and reduces the cost of software development.

Key words: cloud environment; continuous integration; department cloud; corporation cloud; public cloud; private cloud

0 引言

随着软件开发技术的不断发展,软件持续集成(Continuous Integration, CI)已经成为软件开发过程中的一个重要组成部分。持续集成技术已经由最初实现简单的软件编译验证,发展到目前可以进行软件源代码质量监控与管理、软件版本出包、软件版本集成验证等各个方面。

云计算^[1]是将大量的计算资源、存储资源等组合

形成资源共享池,能够根据用户的需求提供资源,具有便利性、可扩容性、经济性等优势。由于云计算的优点,目前基于云环境^[2-5]的持续集成已经广泛应用于各大型软件企业。

文中介绍了基于云环境的持续集成的运行原理、部署和特点,基于云环境的持续集成工程运行过程。以配置管理工具SVN和持续集成工具ICP-CI为主线叙述基于云环境的持续集成。

收稿日期:2017-01-21

修回日期:2017-05-23

网络出版时间:2017-09-27

基金项目:国家部委基础科研计划;国防预研基金项目(A1120110007)

作者简介:姜文(1986-),女,工程师,硕士,CCF会员(E200032324M),研究方向为图像处理与分析、数据库应用和软件工程;刘立康,副教授,研究方向为数字通信、图像传输与处理、图像分析与识别等。

网络出版地址: <http://cnki.net/kcms/detail/61.1450.TP.20170927.1000.080.html>

1 云构建环境

1.1 云构建环境的特点

随着云计算技术的快速发展,软件公司逐渐开始使用私有云虚拟机进行软件产品的持续集成,可以根据软件产品对持续集成构建环境的不同需求提供不同数量和各种规格的虚拟机。这些虚拟机可以有不同的操作系统、不同的系统盘与数据盘,灵活搭建持续集成的构建环境,从而降低软件产品持续集成的成本和提高工作效率。私有云虚拟机的硬件设备由公司进行统一分配与管理,降低了设备的管理成本。

1.2 云构建环境持续集成工作流程

1.2.1 私有云环境平台构建

私有云的实现通常有两种模式:在物理硬件设备上采用虚拟化技术构建私有云环境;在公有云上申请私有云环境。

1.2.2 云构建环境申请

对于单机式持续集成可以在私有云平台上申请虚拟 CI 服务器完成持续集成工作。对于分布式持续集成需要申请一个 CI 系统(由一台主控服务器和多台代理服务器组成),通常 CI 系统通过一个虚拟局域网实现相互之间的信息交流。项目组根据软件产品对持续集成的需求确定虚拟机的数量和规格,在公司网上填写申请云虚拟机的申请单,提交产品经理、公司质量部门以及公司 IT 部门审批,由公司 IT 部门根据申请单提供产品部门所需求的云虚拟机。

1.2.3 云构建环境部署

在虚拟机上安装操作系统、持续集成工具、编译器、代码质量检查工具、打包工具以及一些辅助工具。

1.2.4 云构建环境上开展持续集成工作

在持续集成工具的页面上进行构建工程的搭建,构建工程包括产品源代码静态测试、源代码模块编译、版本包出包、版本包的自动化测试。

2 软件配置管理和持续集成

软件配置管理和持续集成^[6-13]有着密切的关系,配置管理的版本库中存放着软件产品源代码和各种重要的资料,持续集成是从版本库下载源代码进行集成构建和代码测试工作。

2.1 配置管理工具 SVN

SVN 是一个开源的版本控制系统,在服务器上安装 SVN 服务器软件,建立一个版本库(Repository),将数据(项目)导入版本库。在客户端使用 SVN 专门设计的协议访问的 SVN 服务器,提供两种服务模式:一种是 C/S 模式,安装 SVN 客户端命令行工具和图形工具 TSVN。另一种是 B/S 模式,与 Apache 集成,在客户端采用 Http 的扩展协议进行访问;安装 Rsync 工具,并

完成相关配置。用户通过个人密码登陆 SVN 服务器开展工作。

2.2 持续集成工具 ICP-CI

ICP-CI 工具的 CI 系统有两种部署形式:

(1)单机式:由一台 CI 服务器完成持续集成工作,用于代码量小于百万行的开发团队。

(2)分布式:CI 系统由一台主控服务器和多台代理服务器组成,主控服务器根据代理服务器的标签将持续集成任务下发到各个代理服务器,任务完成后将任务结果和日志文件发回到主控服务器。

CI 服务器可以自动检查版本库中的动态库变化,设定时间区间,定时自动完成集成构建工作,并以邮件的形式向开发团队反馈集成构建结果。

2.3 基于 SVN 的持续集成

持续集成包括以下基本要素:开发工程师、配置库、集成构建系统(CI 服务器)、构建工具和反馈机制。CI 服务器提供了针对 SVN 的接口。持续集成步骤通常如下:

(1)开发工程师通过 SVN 向代码库提交代码。

(2)锁库后,下载代码到 CI 服务器完成构建工作。若构建成功,提供新的代码版本。

(3)CI 服务器通过电子邮件向项目经理和开发人员反馈构建信息。

(4)CI 服务器轮询配置库中的变更。

CI 服务器和 SVN 组成的系统架构如图 1 所示。

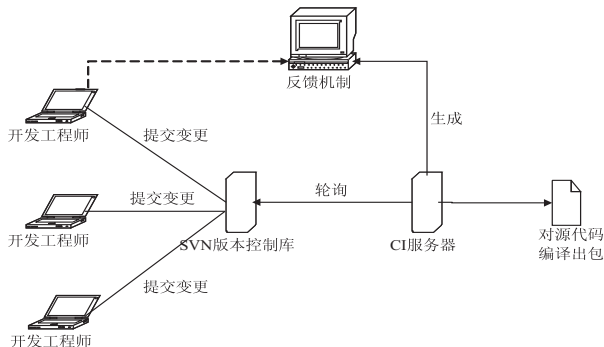


图 1 SVN_CI 系统架构

2.4 基于物理实体机持续集成的特点

(1)持续集成工程师需要完成集成构建工作,也要配置、管理和维护 CI 服务器系统。

(2)采用 C/S 模式,从 SVN 的版本库下载代码。

(3)主控服务器(Master)通过网络管理代理服务器(Agent),它们之间通常组成一个局域网。

(4)项目组的软件产品版本库存放在项目组管控的计算机上。

3 基于云环境虚拟机的持续集成

基于云环境虚拟机的持续检查首先需要构建私有

云平台,然后在私有云平台上配置云构建环境。

3.1 在物理实体机上构建私有云技术

3.1.1 采用 VMware 构建私有云平台

经过多年的发展,VMware^[14]在虚拟化市场处于领军地位,很多企业部署了VMware虚拟化方案。单位IT部门根据产品不同的需求,在统一使用的服务器上安装VMware软件制作不同操作系统、内存、数据盘的虚拟机。目前VMware通过集成OpenStack技术,能让客户将现有的VMware vSphere虚拟机导入OpenStack中,帮助开发者轻松地迁移其重要开发项目,采用OpenStack API来管理各项工作。

VMware虚拟交换机包括两种:vSphere标准交换机和vSphere分布式交换机。vSphere标准交换机的运行方式与物理以太网交换机十分相似。vSphere分布式交换机可以实现虚拟机跨所有成员主机的网络配置。通过vSphere进行虚拟局域网VLAN的配置和管理。

3.1.2 采用 OpenStack 构建私有云平台

目前也有许多单位采用OpenStack^[15]部署虚拟云环境。OpenStack是一个开源的云平台管理软件,正在逐步成为IaaS(Infrastructure as a Service,基础架构即服务)云计算事实上的标准。可以提供身份认证、授权及镜像管理等基础服务,也提供计算、存储和网络三大核心服务。OpenStack几乎支持所有的虚拟化程序,不论是开源的(Xen与KVM)还是厂商的(Hyper-V与VMware)。早期OpenStack是基于KVM开发的,KVM是默认的虚拟机管理程序。

通过Neutron软件提供虚拟网络服务,Neutron支持的用户网络类型有flat、local、VLAN、VXLAN和GRE。通常用的比较多的是VLAN、VXLAN和GRE。VLAN网络类型是OpenStack默认的网络模式。

3.2 基于部门云的持续集成

部门云是以项目组为单位构建私有云平台(部门云),私有云平台是为项目组软件产品开发服务。硬件设备由项目组统一管理。

3.2.1 部门云构建环境运行原理

部门云CI系统由主控服务器、代码中心以及代理服务器3个部分组成。

主控服务器:负责搭建持续集成工程;分配构建任务,反馈构建结果信息;

代码中心:是一种特殊的代理服务器,存放各种配置文件和运行脚本,从版本库下载源代码,转发给代理服务器;

代理服务器:进行集成构建以及相关的工作。

部门云CI系统的运行如图2所示。

图2中虚线框内为CI系统。

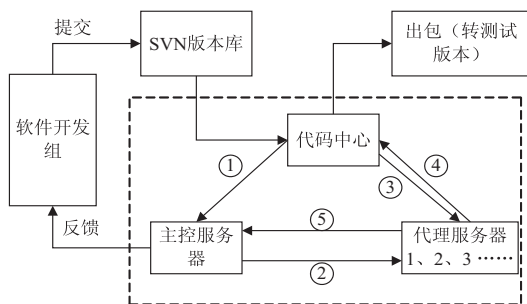


图2 部门云构建环境运行原理

(1)主控服务器搭建持续集成工程,从代码中心下载配置文件与任务的执行脚本。

(2)主控服务器将待执行的任务下发给代理服务器。

(3)代码中心使用SVN客户端工具Rsync从版本库下载产品代码,代理服务器从代码中心同步下载产品代码。

(4)代理服务器完成构建任务后,将构建结果与构建过程中生成的日志文件回传代码中心,代码中心将构建结果与日志文件传送到统一的归档路径下。

(5)代理服务器根据构建结果回传执行报告给主控服务器。主控服务器将相关信息反馈给开发组。

3.2.2 部门云CI系统部署

(1)主控服务器。

主控服务器选择Windows Server操作系统,安装ICP-CI-Windows-Master工具,在配置文件中配置相关的IP地址。执行批处理启动MySQL数据库后,启动ICP-CI的网络页面。

(2)代码中心。

①代码中心选择Windows操作系统,安装ICP-CI-Windows-Agent工具,在配置文件中配置相关的IP地址。

②安装Rsync工具,并完成相关配置。创建相关的目录,用于放置产品代码、构建结果与构建日志。

③在相关目录存放各种构建任务的配置文件和执行脚本,包括代码自动更新、模块编译、版本包打包、版本验证以及各种代码检查任务的配置文件和执行脚本。

(3)代理服务器。

代理服务器可以根据各类任务的需要选择操作系统,安装ICP-CI-Windows-Agent工具或者ICP-CI-Linux-Agent工具,在配置文件中配置相关的IP地址。安装编译器、测试、代码检查、出包相关的工具。

3.2.3 持续集成的特点

(1)对于分布式持续集成,主控服务器(Master)和代理服务器(Agent)组成一个虚拟局域网。

(2)采用B/S模式,在其中一台代理CI服务器

(代码中心)上安装 Rsync 工具,从 SVN 的版本库下载代码。

(3)持续集成工程师登陆 Master 通过网络管理 Agent完成集成构建工作。

(4)持续集成工程师负责分布式 CI 系统的配置、管理和工具软件的安装。

(5)项目组的软件产品版本库存放在项目组管控的私有云上。

3.3 基于公司云的持续集成

公司云是在部门云的基础上进一步开发形成的私有云平台。通常有多个软件开发项目,在公司云上进行软件开发、测试、版本管理和持续集成等工作。公司云管理中心统一管理分配公司云中的虚拟机和各种资源。

3.3.1 公司云构建环境运行原理

公司云构建环境由资源管理调度中心、主控服务器以及代理服务器组成。

资源管理调度中心主要由四部分组成:

(1)资源管理中心:所有资源均在此注册,使用资源需要申请。

(2)资源调度中心:云构建环境中的各种工作任务统一在此排队安排。

(3)POM 服务器:相当于部门云中代码中心的角色,不同之处在于该服务器可以管理多个软件项目的持续集成工作。该服务器存放多个软件项目的各种配置文件和运行脚本,可以从多个软件产品版本库下载源代码,而且存放构建过程中间载体。该服务器可以通过 pom.xml 文件提供多种管理(源代码、配置文件、开发者信息、问题追踪、项目相关信息等)。

(4)PROXY 服务器:承担虚拟机代理服务器的管理工作,向代理服务器下发构建任务,并全程跟踪任务执行。代理服务器通过 PROXY 服务器获取网络信息。

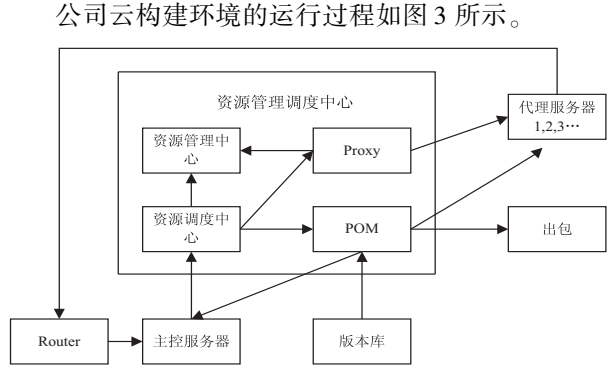


图 3 公司云构建环境运行原理

(1)主控服务器向资源调度中心发送任务请求,准备启动构建任务。

(2)资源调度中心向资源管理中心请求 PROXY

环境,并初始化 PROXY 环境。

(3)PROXY 向资源管理中心申请代理服务器资源,并与 POM 同步配置。

(4)资源调度中心向 POM 下发从版本库更新代码的任务。代理服务器从 POM 同步下载产品代码。

(5)PROXY 向代理服务器下发构建任务,开始构建工作。

(6)构建结束后,代理服务器向 POM 上传构建结果。

(7)POM 将构建结果与日志文件传送到统一的归档路径下。

(8)代理服务器根据构建结果回传执行报告给主控服务器。主控服务器将相关信息反馈给开发组。

3.3.2 公司云构建环境部署

主控服务器和代理服务器的部署与部门云中的部署方式相同,下面简要叙述资源管理调度中心的部署。

(1)资源管理中心与资源调度中心。

资源管理中心与资源调度中心通常采用通用的云计算技术配置。

(2)POM 部署。

POM 服务器选择 Windows 操作系统,安装 POM 工具,完成资源管理中心的 IP 地址配置,安装版本库远程登录工具。

(3)PROXY 部署。

PROXY 服务器,选择 Windows 操作系统,安装 PROXY 工具,完成资源管理中心的 IP 地址配置。

3.3.3 持续集成的特点

(1)持续集成工程师负责向公司云管理中心申请持续集成环境,包括虚拟机规格和各虚拟机安装的各种工具软件。

(2)公司云管理中心提供定制的持续集成环境。

(3)持续集成工程师登陆主控服务器(Master),通过网络管理代理服务器(Agent)完成项目组集成构建工作。

(4)持续集成工程师也可以承担其他项目组的持续集成工作,资源调度管理中心也可以把空闲的虚拟机分配给其他工作使用。

(5)项目组的软件产品版本库存放在公司管控的私有云上。

3.4 基于公有云的持续集成

3.4.1 构建过程

在公有云上构建的持续集成云构建环境过程如图 4 所示。



图 4 公有云上构建的持续集成云构建环境过程

虚拟私有云(VPC)是用户申请公有云^[16]账户后获得的虚拟网络。它在逻辑上与公有云中的其他虚拟网络隔绝。可以在VPC内进行VPC的各种配置,选择IP地址范围、创建子网以及配置路由表、网络网关和安全设置。这个虚拟网络与传统网络运行方式极其相似。

采用此种模式,项目组软件开发、版本管理、持续集成和软件测试都在私有云的网络上进行。开展持续集成工作通常采用分布式版本管理工具Git和持续集成工具Jenkins。Jenkins可以在私有云的虚拟局域网上构建分布式持续集成环境,开展持续集成工作。开源的软件开发通常采用此种模式。商业软件开发由于顾忌软件产品的保密问题,通常不采用这种方式。

3.4.2 持续集成的特点

- (1)项目组的软件产品版本库存放在公有云上。
- (2)开发人员可以把版本库克隆到本地,进行软件开发;工作完成后提交到版本库进行分支合并。
- (3)对于分布式持续集成,持续集成工程师需要申请一个虚拟局域网开展集成构建工作。主控服务器(Master)通过网络管理代理服务器(Agent)。
- (4)采用B/S模式,需要在其中一台代理CI服务器上安装远程登录工具,从版本库下载代码。

4 基于云环境的持续集成工程运行过程

4.1 搭建持续集成工程

在ICP-CI的页面上创建名称为“软件产品名称+版本号”的持续集成构建工程。构建工程创建成功后,分别配置代码更新、进程编译、出包、自动化用例测试等工作内容。

表 1 持续集成计算机规格表

| 类型 | 操作系统 | 核数 | 内存/G | 系统盘/G | 数据盘/G | 数量 | 功能 |
|--------|--------------------|----|------|-------|-------|----|------------|
| 实体机网络 | Windows 7 | 4 | 4 | 40 | 200 | 9 | 代理服务器 |
| | Windows 7 | 4 | 8 | 40 | 500 | 1 | 主控服务器 |
| 部门云虚拟机 | Windows2003 Server | 4 | 4 | 40 | 200 | 18 | 代理服务器 |
| | Windows2003 Server | 4 | 8 | 40 | 500 | 2 | 代码中心与主控服务器 |
| | Windows2008 Server | 4 | 8 | 40 | 200 | 50 | 代理服务器 |
| 公司云虚拟机 | Windows2008 Server | 8 | 8 | 40 | 500 | 1 | POM |
| | Windows2008 Server | 8 | 8 | 40 | 500 | 1 | PROXY |
| | Windows2008 Server | 8 | 8 | 40 | 500 | 1 | 主控服务器 |

- (1)基于实体机的持续集成。
项目初期,使用实体机网络开展持续集成工作。执行一次持续集成完整构建大概需要4个多小时,执行一次完整的代码质量检查需要8个小时以上。
- (2)基于部门云的持续集成。
项目初期,利用部门云虚拟机开展持续集成工作。

4.2 更新配置文件

执行构建任务之前都需要先更新配置文件。配置文件包括所有构建任务的出包脚本、*.ln文件、所有构建任务的配置文件、静态检查的模块配置文件、构建任务运行环境的配置文件。

4.3 代码更新

使用配置管理工具SVN代码更新命令“update”更新代码。每天在版本库开库到锁库的时间段,开发工程师提交代码到版本库中。版本库锁库后,持续集成工程师更新代码,开始执行集成构建工作。

4.4 模块编译

各模块的编译脚本路径、编译脚本与编译器需要在构建工程的任务页面上完成配置,同时配置任务的类型和任务级别。产品代码在编译过程中为产品版本包生成相关文件,同时也提供各种编译错误与编译告警的信息。

4.5 版本包出包

编译任务完成后,运行构建工程的版本出包步骤,生成产品的软件包。软件包存放在指定的目录中。

4.6 版本包验证

在自动化测试用例执行环境下完成版本包自动加载、自动安装和自动化测试用例的初步测试。这个过程被称为HLT(HIGH LEVEL TEST)自动化测试。

5 典型案例

某大型软件公司有一个软、硬件结合的大型软件项目,代码量大概为2 000多万行,在项目开发期间分别使用实体机、部门云虚拟机以及公司云虚拟机进行持续集成。持续集成使用的计算机机规格如表1所示。

- 执行一次持续集成完整构建需要3个多小时,执行一次完整的代码质量检查需要4个小时。
- (3)基于公司云的持续集成。
后来随着软件代码规模的迅速扩张,在公司云上申请了53台虚拟机开展持续集成工作。执行一次持续集成完整构建需要1个多小时,执行一次完整的代

码质量检查需要 2 个小时。

工作实践表明,在虚拟云环境中可以根据软件代码量的变化随时改变虚拟机的数量和规格,灵活开展持续集成工作。

6 结束语

云构建环境中开展持续集成工作,从设备层面来讲,可以对私有云的虚拟机进行统一部署、管理与监控,提高资源利用效率。集中管理可以提高管理人员技术水平,提高服务质量,节约人力和物力。在软件产品开发层面,各产品可以根据软件源代码规模灵活申请云虚拟机的数量和规格,提高软件持续集成的速度与效率,从而节约软硬件资源,提高软件开发效率,降低软件开发成本。

参考文献:

- [1] 陈国良,明仲,冯禹洪.云计算工程[M].北京:人民邮电出版社,2016.
- [2] 魏志华,赵强强.构建企业私有云软件测试平台[J].电子技术与软件工程,2015(9):63-64.
- [3] JUN W, MENG Fanpeng. Software testing based on cloud computing[C]//International conference on internet computing and information services. [s. l.]:[s. n.],2011:176-178.
- [4] 孟祥超.云计算环境下的软件测试服务研究[D].大连:大

(上接第 10 页)

- [3] SIMONCELLI D, DUSI M, GRINGOLI F, et al. Scaling out the performance of service monitoring applications with BlockMon [C]//Proceedings of the 14th international conference on passive and active measurement. Hong Kong: IEEE Press, 2013:253-255.
- [4] ZAHARIA M, DAS T, LI H, et al. Discretized streams: an efficient and fault-tolerant model for stream processing on large clusters[C]//USENIX conference on hot topics in cloud computing. [s. l.]:USENIX,2012.
- [5] 张鹏,李鹏霄,任彦,等.面向大数据的分布式流处理技术综述[J].计算机研究与发展,2014,51:1-9.
- [6] 崔星灿,禹晓辉,刘洋,等.分布式流处理技术综述[J].计算机研究与发展,2015,52(2):318-332.
- [7] GÜNDÜZ S, ÖZSU M T. A web page prediction model based on click-stream tree representation of user behavior [C]//Proceedings of the ninth ACM SIGKDD international conference on knowledge discovery and data mining. [s. l.]:ACM, 2003:535-540.
- [8] BABCOCK B, BABU S, DATAR M, et al. Models and issues in data stream systems [C]//Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems. [s. l.]:ACM,2002:1-16.
- [9] 孙大为,张广彬,郑纬民.大数据流式计算:关键技术及系

连海事大学,2013.

- [5] 严宇平,王学文,陆璐.基于云平台的软件自动持续集成研究[J].信息通信技术,2014,8(1):50-54.
- [6] DUVAL P M, MATYAS S, GLOVER A. 持续集成软件质量改进和风险降低之道[M].北京:电子工业出版社,2012.
- [7] 兰洋,温迎福.持续集成实践[M].北京:电子工业出版社,2015.
- [8] 姜文,刘立康.基于 ClearCase 的软件配置管理与持续集成[J].计算机技术与发展,2016,26(1):10-17.
- [9] 姜文,刘立康.基于 SVN 的软件配置管理和持续集成[J].电子设计工程,2016,24(2):1-5.
- [10] COLLINS-SUSSMAN B C, FITZPATRICK B W, PILATO C M. Version control with subversion for subversion 1.5 [M]. [s. l.]:[s. n.],2005.
- [11] KUNG S, ONKEN L, LARGE S. Tortoise SVN version 1.5.2 [M]. [s. l.]:[s. n.],2005.
- [12] 姜文,刘立康.软件配置管理中的基线问题研究[J].计算机技术与发展,2016,26(6):6-10.
- [13] LOELIGER J, MCCULLOUGH M. Git 版本控制管理[M].王迪,丁彦,译.第2版.北京:人民邮电出版社,2015.
- [14] 王春海.VMware 虚拟化与云计算应用案例详解[M].北京:中国铁道出版社,2013.
- [15] 张小斌.OpenStack 企业云平台架构与实践[M].北京:电子工业出版社,2015.
- [16] Amazon virtual private cloud user guide [R]. [s. l.]:Amazon Web Services, Inc.,2013.
- 统实例[J].软件学报,2014,25(4):839-862.
- [10] PERNG C S, WANG H, ZHANG S R, et al. Landmarks: a new model for similarity-based pattern querying in time series databases [C]//16th international conference on data engineering. [s. l.]:IEEE,2000:33-42.
- [11] BABCOCK B, DATAR M, MOTWANI R. Sampling from a moving window over streaming data [C]//Proceedings of the thirteenth annual ACM-SIAM symposium on discrete algorithms. [s. l.]:[s. n.],2002:633-634.
- [12] ZHU Y, SHASHA D. Statstream: statistical monitoring of thousands of data streams in real time [C]//Proceedings of the 28th international conference on very large data bases. [s. l.]:[s. n.],2002:358-369.
- [13] 熊全洪,魏娟,刘武.即席查询研究[J].现代商贸工业,2008,20(12):345-346.
- [14] CHANDRASEKARAN S, FRANKLIN M J. Streaming queries over streaming data [C]//Proceedings of the 28th international conference on very large data bases. [s. l.]:[s. n.],2002:203-214.
- [15] 何小东,尹海波.基于共享缓冲区的数据流处理框架设计与实现[J].计算机工程与设计,2012,33(11):4398-4401.
- [16] 陈晗鸣,罗威,李明辉.分布式系统中基于主/副版本的实时容错调度综述[J].计算机应用研究,2012,29(11):4017-4022.