

xCloud: 支持异构服务与系统的自主弹性云平台

洪爵 杨聪 文高进 黄伟 须成忠

摘 要 云平台指对云资源、云服务提供调度、部署、控制等管理功能的平台系统,是当前云计算领域的研究热点之一。现有的云计算平台往往只考虑到对数据密集型应用的支持或者对交互服务的支持,并没有提供对异构服务及其运行系统的统一支持方案。相关的自主和弹性管理平台通常只关注于单一层面的管理功能,没有提供包括基础资源、集群资源和应用服务在内的统一管理平台,因此缺乏有效的全局资源管理优化机制。本文针对现有工作的不足,提出了一种支持异构服务与系统的自主弹性云平台:xCloud。xCloud提供了从基础资源到上层应用服务的资源动态弹性管理以及系统部署机制。同时,结合统一的监控管理模块,可以提供跨层的全局监控与性能调优,从而为实现自主管理策略提供了必要的机制。基于自主、可靠的管理机制,结合各种灵活高效的管理策略,xCloud可以对高性能计算、交互式云服务等异构应用及其运行系统提供良好的支持。本文将对xCloud云计算平台的总体设计进行详细阐述。

关键词 云计算管理系统; 异构服务; 自主弹性管理; 自动部署; 系统监控

1 简介

云计算是当前业界和学术界的研究热点,针对不同应用场景和需求的云计算相关平台系统层出不穷。云计算平台指对云资源、云服务提供调度、部署、控制等管理功能的平台系统,是当前云计算领域的研究热点之一。按照服务提供类型和资源管理边界,现有的云计算平台系统可大致分为以下几类:基础级管理平台(Infrastructure Management Platform),集群级管理平台(Cluster Management Platform)以及应用级管理平台(Application Management Platform)。基础级的管理平台主要用于IaaS服务场景,提供对数据中心机器资源、网络资源以及虚拟化资源的监控、管理与调度等功能。集群级管理平台以集群系统为管理对象,用于PaaS服务场景,提供对集群系统计算、网络、存储等资源的管理,以及集群系统间资源的分配和共享管理等。应用级管理平台主要用于在某一集群中对用户应用进行管理,用于SaaS服务场景,提供对不同应用之间的资源分配、执行、监控等管理功能。

当前针对这三种管理平台的研究和开发工作都取得了丰硕的成果。但是在云平台管理机制层面,往往只关注于单一层次的管理功能,没有提供包括基础资源、集群资源和应用服务在内的统一管理平台,因此缺乏有效的全局资源优化机制。同时,现有的云计算平台通常只提供了对数据密集型应用或者交互服务的支持机制,并没有提供很好的异构服

务支持机制,也很少考虑对异构服务运行系统提供相应的自动部署支持。具体来说,有待进一步研究和改进的问题如下:

- 在基础级管理平台中,如何针对异构服务的运行系统提供自动部署和弹性管理功能,并对系统性能进行监控,目前尚未有成熟的支撑机制。
- 在集群级的管理平台中,虽然最近出现了一些新颖高效的方案^[6],但是体系结构设计耦合度和开销之间的矛盾有待解决。相应的资源管理方式和调度算法也有待完善。
- 在应用级的管理平台中,当前工作的不足主要在于面对各种复杂类型应用时,难以提供统一的编程框架来实现复杂应用的执行逻辑。如何设计满足不同资源需要的调度策略是研究的难点。
- 当前三种平台的设计多遵循自治原则,其资源管理均局限于其管理边界内。不同层次平台间的资源需求变化情况不能共享。例如,应用的负载变化难以反映到集群规模的变化上来,集群系统的需求变化也难以反映到基础级管理平台上来,应用的负载变化到基础级管理平台的传递更是难以实现。
- 当前工作缺乏包括应用、集群系统、基础硬件资源在内的跨层次跨边界性能监控和反应机制,使得云计算平台缺乏一种整体考虑和优化的机制和手段。

针对现有工作不足, 本文提出了一种支持异构服务及其运行系统的自主弹性云平台xCloud。xCloud首先在基础资源管理层面上, 提供了基础管理系统的自动部署和弹性资源调度机制, 为集群系统的部署和运行提供了支持。同时, xCloud为集群平台系统提供了自动部署和自主资源共享机制, 为异构服务的运行提供了底层支持。通过结合从基础资源到集群资源的管理机制, xCloud可以为异构服务及其运行系统提供资源动态弹性和自主管理机制。同时, 结合统一的监控管理模块, 可以提供跨层的全局监控与性能调优, 从而为实现自主管理策略提供了必要的机制。基于自主弹性管理机制和各种灵活高效的管理策略, xCloud对高性能计算、交互式云服务、异构应用及系统提供了良好的支持。本文的后续部分将对xCloud的总体设计进行详细描述。

2 相关工作

目前的云平台工作大多集中在对单一层面的管理机制上。现有的云计算基础级管理平台系统主要关注于如何对各种实体硬件资源和虚拟化资源进行管理, 如服务器性能监控、虚拟机的创建运行和维护、服务器(虚拟机)网络资源的配置和变更、虚拟机的整合、调度和迁移等。典型的平台软件有Euclyptus^[1]、OpenNebula^[2]、VmWare ESX^[3]、Platform ISF^[4]等, 相应的开发软件包则有libvirt^[5]等。这些系统和软件包在对虚拟机管理器进行操作的基础上, 根据相应的策略来实现了一系列方便、强大的管理功能。结合相应的资源弹性管理机制和策略^[22], 这些系统可以提供灵活的混合云资源服务。

集群级管理平台是云计算研究的一个新兴热点方向, 以集群系统为管理对象, 主要关注点在于如何在同一服务器集群中运行多个集群管理系统, 以实现集群资源利用率的最大化。典型的系统有如mesos系统^[6], Platform LSF的hadoop支持方案^[7], 以及Condor的hadoop集成方案^[8]。这些系统均基于对集群管理系统实现资源按需分配的思路, 实现了多个集群管理系统的资源动态共享。同时, 在传统的网格研究中, 也有不少工作关注于集群管理系统之间的资源共享, 如Condor-G^[9]、VioCluster^[10]、Cluster-on-Demand(COD)^[11]等。这些工作通过在集群间设置某种代理, 来实现集群间机器的拆借, 实现了集群规模根据负载情况弹性伸缩的功能。

应用级管理平台和系统也是目前云计算研究中的热点之一。首先, 传统的批作业处理系统如Condor^[12]、Platform LSF^[13]、Torque^[14]等纷纷得到改进, 以满足云环境下用户作业和应用多样化的需求。这些系统增加了对具有依赖关系的作业、定时作业、常驻型任务等复杂类型应用的支持, 更好地满足了云计算用户的需求。其次, 新兴的集群系统如Hadoop^[15]、Dryad^[16]等, 放弃了传统集群管理系统的大而全的开发思路, 针对于搜索等当前互联网典型业务的特点设计独特的任务分配和执行模式, 简洁高效, 得到了更广泛的应用。

同时, 在对异构服务支持和混合云平台方面, 最近也涌现了不少新的工作。混合云方面典型的工作如OpenNebula利用Amazon EC2^[21]接口结合Haizea^[22]实现了混合云的调度与管理工作^[23]。在对异构服务的支持方面, [25][26]分别针对在云环境中运行基于Mapreduce的高性能计算策略进行了研究。[24]以Euclyptus与Amazon EC2为支撑, 提出了一种支持自适应服务的自主资源管理框架。Oracle在[27]中提出了一种结合虚拟化与集群化技术对异构应用进行支持的平台体系。[28]与[29]分别针对传统高性能计算任务在云环境中的部署提出了相应的支撑体系。特别的, [30]提出的Phoenix Cloud框架针对在云数据中心同时支持异构类型计算任务设计了相应的自动部署和弹性资源分配功能。但是, 上述工作只针对异构服务本身提供了支持, 并没有针对运行系统的弹性管理进行研究, 也没有给出一个跨层统一的系统管理方案。

3 设计目标

xCloud云计算平台旨在构建支持异构服务及其运行系统地的弹性云平台, 提供对包括IaaS、PaaS和SaaS在内的支撑管理功能。同时, xCloud还提供跨层次的资源管理机制, 以提高云计算环境各资源的利用率; 提供跨层次的全局性能监控机制, 以实现全局性能优化策略。xCloud的设计目标可总结如下:

- 针对IaaS类型服务, 提供数据中心机器的信息收集管理功能; 提供虚拟机资源的动态构建、配置、迁移和控制等管理功能; 同时提供对存储、网络等资源的虚拟化服务; 提供服务运行系统的自动部署及资源弹性分配机制;
- 针对PaaS类型服务, 提供各种平台系统(集群系

统)的自动部署、配置、运行和回收功能;提供集群规模的动态扩容收缩功能;实现集群系统的资源按需分配和集群之间的资源共享功能;

- 针对SaaS类型服务,提供各种异构应用运行的支撑环境,包括对不同类型复杂应用的运行时支持,对不同应用资源调度策略的支持等;
- 通过实现不同级别管理域之间的接口,实现跨级别的资源管理机制,在私有云域和公有云域之间、集群系统之间以及各平台应用之间实现资源的共享;
- 提供不同级别管理域内的监控进程,实现各级别管理域内的监控机制;提供统一的监控模块,收集各级别管理域内的监控信息,以实现系统级别的监控功能;
- 提供各管理域内管理模块接口,通过系统级别的监控模块,实现系统级别的自主控制功能。

针对以上设计目标,xCloud计算平台拟依赖于底层分布式文件系统,分别实现基础资源(虚拟资源)管理模块、集群系统管理模块以及系统性能监控模块。基础管理模块(Infrastructure Manager-IM)负责基础资源和虚拟化资源管理,集群资源管理模块(Cluster Resource Manager-CRM)负责集群系统管理,系统性能监控模块(System Performance Monitor and Controller-SPMC)负责系统级性能监控和调节。同时,在基础资源管理模块和集群系统管理模块之间,提供资源申请、供应等管理接口,实现跨模块的资源共享。分别在集群系统管理模块和基础资源管理模块内部,提供模块性能监控机制,以便于实施各种性能调节策略。在模块外部,提供统一的信息监控模块,通过集群系统和基础资源管理模块中的性能信息收集接口获取系统性能信息,并通过各模块的控制接口进行系统级别的参数调整,

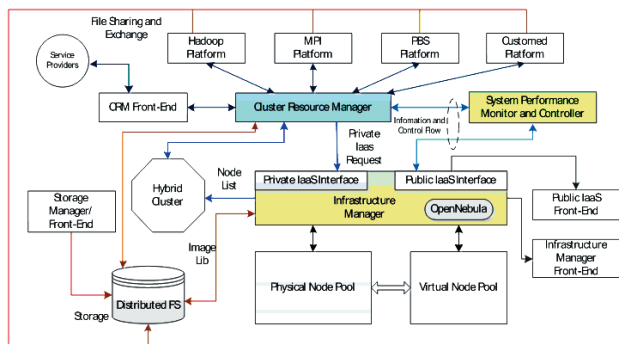


图1 xCloud系统总体结构

从而为实现云平台自主监控和管理提供基础机制。

4 xCloud总体设计

根据设计目标和解决思路,xCloud平台的总体设计方案可参见图1。

如上文所述,xCloud系统的主要模块为负责基础资源和虚拟化资源管理的Infrastructure Manager(IM),负责集群系统管理的Cluster Resource Manager(CRM),以及负责系统性能监控和调节的模块System Performance Monitor and Controller(SPMC)。公有云和私有云的服务器请求由IM模块进行处理(IaaS服务)。IM模块将根据需求构造一系列虚拟机或者在数据中心中选择一部分服务器,完成相应的配置,返回给用户。在此系统中,私有云的服务请求主要来源于CRM模块,即负责对集群系统进行管理和执行的模块。CRM实现了在单一集群服务器集合上同时运行多个不同集群系统的功能,并且能够在这些集群系统之间实现资源的共享。如图中所示,CRM在向IM申请到服务器资源后,可以在其上同时运行Hadoop、MPI、Condor等集群管理系统。通过CRM,可以灵活地为实现不同PaaS服务提供所需要的支持。在CRM的支持之上,将实现一个简单的作业处理系统来运行示范应用平台,向用户提供简单的示范性SaaS服务。

在xCloud系统中,分布式文件系统(Distributed File System-DFS)是一个重要的支撑结构,贯穿于整个系统的全流程。IM模块可以利用DFS来存储虚拟机镜像。CRM模块可以利用DFS来实现二进制可执行文件的传输和暂存。运行在CRM之上的各种集群系统可以利用DFS实现其内部的数据存储和传输。如Hadoop可以利用DFS来替代HDFS的功能;Condor可以利用DFS来替代NFS等。在另一方面,DFS还直接向用户提供了数据的分布式高效存储方案和应用。

在云平台系统中,由于在很多模块和处理流程中需要提供一定的位置、命名透明性,以及可能需要进行分布式同步操作,因此需要提供名字服务、分布式锁等机制。xCloud拟采用Zookeeper^[17]来提供命名服务和分布式同步操作。Zookeeper是Yahoo对google的Chubby^[18]系统的开源实现,采用了Paxos算法提供了分布一致性的小数据存储功能,可以方便地用于实现名字服务和分布式协同操作等。

以下将分别针对xCloud的主要功能模块设计进行描述。

5 基础管理模块设计

基础管理模块(IM)的主要功能为对数据中心的机器进行管理,提供基础资源管理系统(如mesos、OpenNebula、Euclptus等)的自动部署和运行服务,并根据需求对虚拟机、虚拟网络、存储等资源进行创建、申请、控制和回收等工作。IM模块的框架如图2所示。

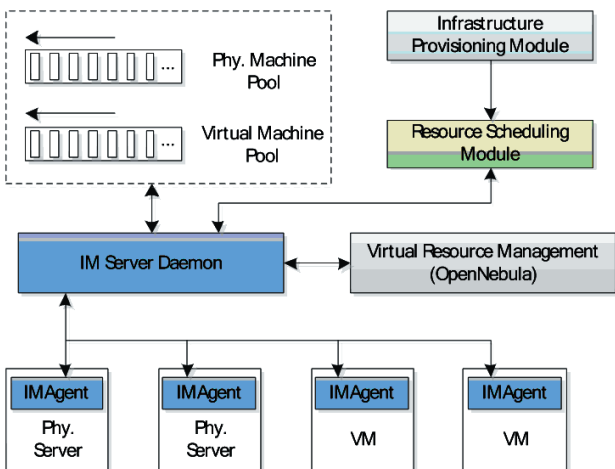


图2 基础管理模块结构图

IM模块主要包括以下四个子模块:

服务器管控子模块(Server Management Module - SMM):根据管理需求,在每台服务器上运行一个守护进程IMAgent,负责收集服务器的各种配置参数、性能参数向IM汇报。同时,IM可以通过IMAgent来更改机器的硬件配置信息,如各种网络参数、CPU电压、fd数目等系统参数等等。IM维护一监控队列(可用多种高效队列实现),存储各监控服务器的信息,并能够随时更新并提供查询等服务。每台服务器在格式化、安装好系统、加电启动后,自动运行IMAgent,加入IM的监控队列中。通过此机制,IM可以对数据中心的硬件服务器资源实现相应的基础管理,如对机器进行资源池的分配工作。IMAgent的运行可以采用进程级别的容错机制,在出错崩溃后,自动重启。

虚拟资源管控子模块(Virtual Resource Management Module - VRMM):IM拟采用如OpenNebula等虚拟化管理工具来提供虚拟机的创建、迁移、挂起、销毁等操作功能。同时还提供虚

拟网络的构建,以便于虚拟机之间、虚拟机和物理服务器之间根据需求组建网络。同时,借助于类似于OpenNebula提供的调度机制,可以实现构建虚拟机时刻以及虚拟机运行时刻的资源整合调度。

基础资源供应子模块(Infrastructure Provisioning Module - IPM):IM模块的用户可以划分为私有云用户和公有云用户两种。借助于虚拟资源管控子模块,IM可以直接通过前端接口(Public IaaS Interface)为用户按需构造虚拟机,并设置好相应的网络配置,实现IaaS服务。同时,IM通过内部接口(Pprivate IaaS Interface)为内部系统提供服务器资源的分配和回收功能。在基础资源供应子模块中同时提供软件部署功能,根据用户的需求为虚拟机和服务部署相应的软件环境,或者执行所要求的程序。

资源调度子模块(Resource Scheduling Module - RSM):IM面临着在内部私有云用户和外部公有云用户之间按照一定策略分配资源的问题。因此IM中设置一个资源调度子模块,利用多队列等数据结构将用户对基础服务的请求缓存,然后根据一定策略来进行处理。

6 集群系统管理模块设计

集群系统管理模块(CRM)负责对私有云域中所运行的集群系统进行资源的分配、管理、调度、回收以及系统的部署等功能。利用CRM,xCloud可以在单一服务器集合中启动多个不同的集群系统。这些集群系统间可以共享机器资源,提供了利用率。CRM模块的简要结构如图3所示。

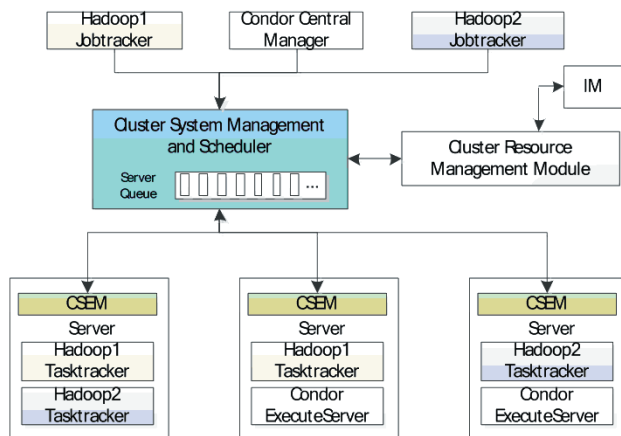


图3 集群系统管理模块结构图

CRM主要包括以下三个主要子模块:

集群资源管控子模块(Cluster Resource Management Module - CRMM): 集群资源管理子模块负责管理CRM中的机器资源。CRM中的机器资源所构成的集群称为混合集群(Hybrid Cluster), 可以由不同配置的物理服务器和虚拟机构成。作为上层的集群系统, 不会察觉这两种机器类型的区别。CRMM负责在初始化时向IM模块申请机器资源。申请到的机器资源已经由IM根据申请需求配置好了网络和相关的软件环境。同时CRMM还提供向IM申请和回收机器的接口, 以便于实现集群规模的扩容和收缩功能。

集群系统管理调度子模块(Cluster System Management and Scheduler - CSMS): 当集群机器资源申请完成后, CRMM将在预选的一个服务器上执行集群系统管理调度子模块CSMS。CSMS一方面负责通过心跳机制与集群中所有机器联系, 收集记录机器信息; 另一方面负责接收不同集群系统的资源请求, 并根据一定的资源共享策略在集群系统间进行资源分配, 选择请求进行处理, 然后向集群系统执行子模块提交执行相应集群系统任务。

集群系统执行子模块(Cluster System Execution Module - CSEM): CSEM为运行在Hybrid Cluster中几乎所有服务器上(一般运行了CSMS进程的服务器除外)的执行守护进程, 向CSMS汇报, 构成master-worker的工作方式。当CSMS选择了某个集群系统的任务后, 将向CSEM提交。CSEM中的执行引擎将负责启动此任务并监控其运行过程直至结束, 随后进行环境清理工作。CSEM还提供对任务的运行状况的监控和挂起、停止等功能。另外, CSEM还向运行的集群系统任务提供一定的API, 以便于集群系统能够向CSEM汇报其相关的运行信息和性能信息, 如是否错误、负载情况等。

7 性能监控系统设计

系统性能监控系统作为一个较为独立的模块存在于xCloud中。监控系统主要完成两大功能: (1) 监控信息的持久化, 即收集各节点的系统和应用性能数据, 保存到数据库中作为历史信息; (2) 系统监控与报警, 即根据收集到的系统或者应用数据, 并结合各种故障预测模型进行监控与报警。依据以上两个主要功能需求, 性能监控系统主要由以下主要子模块构成: 节点守护进程、消息队列、监控与报

警模块、数据持久化模块以及终端可视化模块。

节点守护进程运行在每个节点服务器中, 主要完成以下几个方面的功能: 获取本服务器的监控信息; 对监控信息进行本地判断, 如果发现有数据超过了预警值, 则对外发出警报信号; 定期将监控到的信息向消息队列模块(如RabbitMQ [19]); 听外部监控请求并在本地监控信息队列中读取监控信息并对外发送。

数据持久化模块主要用于从消息队列中获取数据, 并保存于数据库中以用于历史信息查询。在监控系统中, 节点守护进程会周期性的将监控到的信息发送到消息队列。随后数据持久化模块从消息队列中接收这些监控数据信息, 进行过滤后保存于数据库中。

监控与报警模块主要包括web服务模块与智能终端服务。监控报警模块监听查询请求并进行分析, 决定是从集群中还是从历史数据库中进行查询。查询结果经过组织后传送到客户端进行显示。同时, 模块随时监听从集群中产生的报警信息, 并将信息通过推送至智能终端等设备上。终端可视化模块将根据终端类型对报警信息进行相应展示。

监控系统的一个结构图如图 4所示。监控系统的构成主要包括四部分: 节点服务器集群、功能服务器集群、数据库、监控端。其中, 节点服务器集群的粒度可以为Federation(Cluster集群), 也可以为Cluster(节点服务器集群)。功能服务器集群包括监控服务器与消息队列服务器, 其中, 监控服务器为一台, 消息队列服务器可以为多台。数据库可以采用MongoDB [20]等noSQL类型数据库。监控端可以为PC(采用PHP网页形式呈现)或智能终端(采用WAP与手机客户端的方式呈现)。

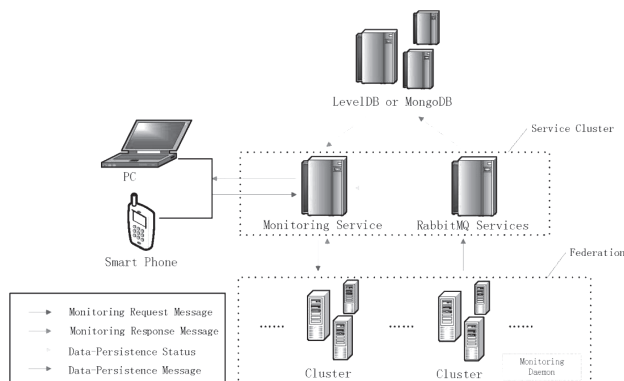


图 4 监控系统设计图

监控系统的一般流程如图 4 中所示。如对于监控业务流程, 首先手机用户会在客户端选择要监控的对象, 之后手机会将该对象请求信息传送给 Monitoring Service(红线), Monitoring Service 会判断需要的监控信息来源, 如果从 MongoDB 中读取, 则会直接读取 MongoDB 中的相关数据(绿线)。如果从节点中读取, 则会给节点发送命令(红线), 节点在接到命令后会对相关的监控信息进行读取, 之后传送给 Monitoring Service(绿线), Monitoring Service 在对传送过来的数据进行整合后发送给手机监控端(绿线)。同样, 对于报警业务流程, Cluster 节点与 RabbitMQ Service 会将报警信息传送给 Monitoring Service, 之后 Monitoring Service 在将数据进行处理后推送给手机监控端进行报警。对于在数据持久化业务流程, Cluster 中的每个节点会周期性的将监控到的信息传送给 RabbitMQ Service, 之后 RabbitMQ Service 会将整理后的数据插入到数据库中进行持久化。与此同时, RabbitMQ Service 会更新内部的 NodeTable, 并对其进行监控, 如果发现问题会及时将错误信息以报警的方式发送给 Monitoring Service, Monitoring Service 之后将报警信息推送给手机。

8 总结与展望

在本文中, 我们详细描述了 xCloud 系统的总体设计。从总体设计中可以看出, 通过基础管理模块、集群资源管理模块以及系统性能监控模块, xCloud 针对异构云服务提供了资源的自主管理、弹性伸缩、动态自动部署管理等机制, 并提供了跨层全局性能监控调优机制。在这些机制的基础上, 结合未来的策略研究, xCloud 将会对异构服务的自动部署和运行提供完善的支持。在未来的研究工作中, 我们将在以下方面开展研究:

- 云平台多粒度资源共享, 包括基础服务器资源、虚拟机资源、容器(resource container)资源级别的共享算法;
- 云资源规模动态扩容和收缩机制, 包括资源规模估算、资源自动获取回收、弹性分配管理等机制与算法;
- 复杂需求云服务应用群组调度, 包括复杂云服务的运行时支持、云服务群组调度算法等;
- 系统自动监控与故障恢复机制与策略, 包括系统性能-参数模型、故障预测与恢复的机制与策略;

- 系统架构和性能优化, 包括海量数据处理和云服务的高性能架构与算法。

参考文献

- [1] Eucalyptus. <http://www.eucalyptus.com>
- [2] OpenNebula. <http://opennebula.org>
- [3] VmWare ESX. <http://www.vmware.com/products/vsphere/esxi-and-esx/index.html>
- [4] Platform ISF. http://platform.com.cn/pdf/Platform_ISF.pdf
- [5] The libvirt API. <http://libvirt.org/>
- [6] B. Hindman, A. Konwinski, M. Zaharia, A. Ghodsi, A. D. Joseph, R. Katz, S. Shenker, and I. Stoica, "Mesos: A Platform for Fine-Grained Resource Sharing in the Data Center", Proceedings of NSDI 2011.
- [7] Platform LFS Integration with Hadoop. http://info.platform.com/WP_Platform_LSFIntegrationwithHadoop.html
- [8] Condor and Hadoop Map Reduce integration. <https://issues.apache.org/jira/browse/HADOOP-428>. 2009.
- [9] J. Frey, T. Tannenbaum, I. Foster, M. Livny, and S. Tuecke, "Condor-G: A Computation Management Agent for Multi-Institutional Grids", Proceedings of HPDC10, 2001.
- [10] P. Ruth, P. McGachey, D. Xu, "VioCluster: Virtualization for Dynamic Computational Domains", Proceedings of CLUSTER'2005.
- [11] J. Chase, L. Grit, D. Irwin, J. Moore, and S. Sprenkle, "Dynamic Virtual Clusters in a Grid Site Manager", Proceedings of HPDC12, 2003.
- [12] The Condor Project. <http://www.cs.wisc.edu/condor/>
- [13] Platform LSF. <http://www.platform.com/workload-management/high-performance-computing>
- [14] The Torque Resource Manager. <http://www.clusterresources.com/products/torque-resource-manager.php>
- [15] Hadoop home. <http://hadoop.apache.org/>
- [16] M. Isard, M. Budi, Y. Yu, A. Birrell, and D. Fetterly, "Dryad: Distributed Data-Parallel Programs from Sequential Building Blocks", Proceedings of Eurosys 2007.
- [17] P. Hunt, M. Konar, F. P. Junqueira, and B. Reed, "ZooKeeper: wait-free coordination for internet-scale systems", Proceedings of USENIX ATC 2010.
- [18] M. Burrows, "The Chubby lock service for loosely-coupled distributed systems", Proceedings of OSDI 2006.
- [19] RabbitMQ - Messaging that just works. <http://www.rabbitmq.com/>
- [20] MongoDB. <http://www.mongodb.org/>
- [21] Amazon EC2. <http://aws.amazon.com/ec2/>
- [22] B. Sotomayor, K. Keahey, I. Foster, "Combining batch execution and leasing using virtual machines", Proceedings of HPDC17, 2008.
- [23] B. Sotomayor, R.S. Montero, I. M. Llorente, I. Foster, "Virtual Infrastructure Management in Private and Hybrid Clouds", IEEE Internet Computing, Vol. 3, Iss.5, Sep. 2009.
- [24] Q. Zhu, G. Agrawal, "Resource Provisioning with Budget Constraints for Adaptive Applications in Cloud Environments", Proceedings of HPDC 19, 2010.
- [25] J. Qiu, J. Ekanayake, T. Gunaratne, J. Y. Choi, S. Bae, H. Li, et al., "Hybrid cloud and cluster computing paradigms for life science applications", Proceedings of BSOC 2010.
- [26] S. Shrish, G. Mackey, J. Wang, and J. Bent, "MRAP: A Novel

MapReduce-based Framework to Support HPC Analytics Applications with Access Patterns”, Proceedings of HPDC19, 2010.

- [27] P. Bramey, D. Wojciechowski, “Cloud Infrastructure – Deploying an elastic and heterogeneous application with IBM system z running linux”, http://www.zseriesoraclesig.org/2011presentations/2011_459Collaborate_cloud_16x9_pbversion2.pdf
- [28] G. Juve and E. Deelman, “Wrangler: Virtual Cluster Provisioning for the Cloud”, Proceedings of SC 2010.
- [29] Fox, G. C., A. J. Younge, G. von Laszewski, A. Kulshrestha, and F. Wang, “Rain: Dynamically Provisioning Clouds within FutureGrid”, Proceedings of SC 2010.
- [30] J. Zhan, L. Wang, B. Tu, Y. Li, P. Wang, W. Zhou, and D. Meng, “Phoenix Cloud: Consolidating Heterogeneous Workloads of Large Organizations on Cloud Computing Platforms”, Proceedings of CoRR, 2009.

作者简介

洪 爵 男，助理研究员，南京大学博士，研究方向：分布式处理与并行计算，云计算。

杨 聪 男，中科院深圳先进技术研究院客座学生，研究方向：云计算系统监控，移动云计算。

文高进 男，副研究员，中科院计算所博士，研究方向：高性能计算，云计算，计算机图形学。

黄 伟 男，副研究员，中科院计算所博士，研究方向：高性能计算，云计算，云存储。

须成忠 作者简介详见封2页。