科学工作流与高性能计算集成方案

赵 勇，李有福，李小龙，刘 鹏，田文洪

(电子科技大学计算机科学与工程学院 成都 611731)

【摘要】科学工作流为科学计算提供了工作流定义、流程管理和任务并行化等支持，高性能计算为大规模数据处理提供了集群管理、任务管理、资源调度等机制。如今正进入一个“大数据”时代，将科学工作流系统与高性能计算结合实现高性能计算平台上大规模并行计算具有重要意义。集成中间件与上层工作流系统和底层高性能计算平台进行交互，提供任务提交与状态监控功能。同时，集成方案为分布式集群中计算平台提供新的参考实现。基于上述分析以Swift科学工作流与Windows高性能计算平台集成方案为例，通过NASA MODIS图片处理工作流来分析并验证集成方案的可行性和性能。

关 键 词 计算平台； 分布式集群； 高性能计算； 大规模并行计算； 科学工作流

中图分类号 TP302.7 **文献标志码** A doi:10.3969/j.issn.1001-0548.2014.03.024

An Approach to Integrate Scientific Workflow with High Performance Computing

ZHAO Yong, LI You-fu, LI Xiao-long, LIU Peng, and TIAN Wen-hong

(School of Computer Science and Engineering, University of Electronic and Science Technology of China Chengdu 611731)

Abstract Scientific workflow provides scientific computing with workflow specification, workflow process management, task parallelism, etc. High performance computing provides mechanisms and development interfaces such as cluster management, task management, task scheduling, etc. to scientific computing. While we are entering into a “big data” era, it is necessary to integrate scientific workflow with high performance computing to implement the large scale parallel computing on high performance computing platform. The integration middleware interact with upper workflow systems and underlying HPC platform provides the support for task submission and status monitoring. The integration architecture will be a reference solution to the construction of computing platforms in distributed cluster environment. Taking Swift scientific workflow system and Windows HPC platform integration solution as references, a case study by using a NASA MODIS image processing workflow is presented to analyze and demonstrate the capability of the integrated system.

Key words computing platform; distributed cluster; high performance computing; large scale parallel computing; scientific workflow

我们正在进入一个“大数据”时代，全球产生的数据量呈“爆炸式”的增长。根据最近的IDC研究报告，在2010年全球的数据信息总和达到1 ZB (zettabyte)。Google和Bing等搜索引擎每天都会产生数TB的搜索日志。社交网络产生的数据量也十分巨大，Facebook每月产生300亿条内容，包括web链接、新闻、状态、博客文章和视频与图片的评论等[1]。科学界同样面临来自实验数据、模拟数据、传感器数据和卫星数据等“数据泛滥”问题[2]。欧洲核子研究组织的大型强子对撞机[3]每秒钟能够产生大于100 TB的碰撞数据；GenBank[4]是全球最大的DNA序列数据库之一，其中已经包含了超过1 200亿个碱基数据，并且这一数量每9～12个月翻一番。物理学、地球学、医学等许多领域的数据量也在快速增长。

科学工作流管理系统(SWFMS)对于科学计算有重要的意义，它们提供了工作流定义、过程协调、作业调度与执行、资源跟踪和容错等功能。Taverna[5]，Kepler[6]，Vistrails[7]，Pegasus[8]，Swift[9]，VIEW[10]等工作流系统在许多领域都有广泛的应用，如物理学、天文学、生物信息学、神经科学、地球学和社会科学等。同时，科学设备和网络计算的发展向可靠的工作流系统在数据规模和应用复杂度方面发起了新的挑战。

高性能计算(high performance computing，HPC)

收稿日期：2013－03－20； 修回日期：2013－09－06
基金项目：国家自然科学基金(61034005, 61073175)
作者简介：赵勇(1971－)，男，博士，教授，主要从事云计算、网格计算、高性能计算、大数据处理和工作流等方面的工作.

是计算机科学的一个分支，可以最大限度提高系统的I/O、计算和数据传送性能。主要用于解决大规模科学问题的计算和海量数据的处理，如科学研究、气象预报、计算模拟、军事研究、CFD/CAE、生物制药、基因测序、图像处理等。

本文提出一个将科学工作流系统与高性能计算平台结合的方案，集成方案涵盖工作流定义与提交、流程解析、任务调度与执行以及状态监测等工作流管理涉及到的所有主要过程。既能灵活方便地描述大规模的应用流程，又能有效地利用高性能计算集群资源管理和任务调度功能，实现对大规模HPC应用并行化端到端的支持。

1 相关工作及研究意义

学术界和业界根据不同研究和应用方向开发出各具特点的工作流系统[11]，随着科学计算过程中数据信息的处理规模急剧增长，集群计算资源在科学工作流中扮演着越来越重要的角色。一些研究集中在基于Taverna工作流系统与网格环境协作，如UNICORE plugin[12]、gLite plugin[13]、caGrid plugin[14]等，它们使Taverna工作流系统能够便捷地访问网格计算资源；基于Windows平台工作流系统的研究专注于Windows Workflow Foundation (WWF)[15-16]的相关应用和平台架构[17]，如MyCoG。NET[18]实现WWF和Globus网格服务无缝结合，基于WWF的Trident[19]为NEPTUNE[20]海洋学项目、Pan-STARRS[21]天文学项目等科学研究提供高效的科学工作流平台。由于Windows HPC Server并不支持应用流程的管理和定制，使用WWF工作流工具并不能有效地支持大规模的并行应用，也没有实现和Windows HPC Server的有效的集成。

Swift工作流系统提供了可以实现和各种资源管理器和任务调度器协作的Provider接口，目前已经实现了的接口包括PBS[22]、Condor[23]、Globus Toolkit 4[24]等，它们也使Swift工作流系统能够便捷获取网格等计算资源；文献[25]研究了工作流系统与云计算的集成方案，详细描述了将工作流管理作为云服务的集成架构，并以Swift工作流系统与OpenNebula云平台集成为例，验证并分析集成方案的功能。这些研究主要集中在工作流系统与网格计算、云计算和分布式计算等计算资源的协作。工作流系统与高性能计算集群资源相结合方面的研究并不多见。

文献[26]等实现了基于MATLAB的SSH工具包，用户可以使用简单的MATLAB命令访问远程高性能计算资源，运行MATLAB应用并获取运行结果。文献[27]探讨了使用Windows高性能计算资源进行并行化地理空间分析，Windows HPC Server运行Inverse Distance Weighting (IDW)应用程序，IDW程序的运行的整体流程包括域分解、空间内插、输出采集及数据可视化。这些研究主要是基于科学应用访问Windows高性能计算资源，并没有深入探讨应用程序运行过程中的计算并行化与流程管理。

2 集成方案

本文首先介绍科学工作流系统与高性能计算集成的统一架构，并分析架构的重要组成子系统和组件，然后以Swift科学工作流管理系统与Windows高性能计算平台集成方案为例，通过对Swift与Windows HPC的架构进行分析，映射到参考架构中，从而进一步验证集成参考架构的可行性。

2.1 集成参考架构

科学工作流管理系统与高性能计算集成的参考架构可以作为一种规范化工作流系统与高性能计算集成的研究和开发工作的尝试，如图1所示，参考架构包含5个逻辑层和11个主要的功能子系统，自上而下涵盖从工作流定义、任务调度到最终大规模应用的整体过程。第一层是开发层，其中包括工作流的开发环境、提交软件工具等服务及相应的操作环境；第二层称作工作流管理层，这一层包括4个子系统：工作流引擎、任务管理、工作流监控和资源配置管理；第三层称作集成中间件层，由任务提交组件和计算资源供应服务组成；第四层为高性能计算管理层，由作业执行组件、资源调度系统和集群管理系统组成；最后一层为应用层，简要描述基于集成平台的科学应用。

参考架构允许科学工作流与高性能计算研究人员根据不同的工作流系统和高性能计算平台特性，定制可用的集成平台以满足大规模数据处理和科学计算等需求。

2.2 Swift架构

Swift系统作为科学工作流和并行计算之间的桥梁，是一个面向大规模科学和工程工作流的快速、可靠的定义、执行和管理的并行化编程工具。Swift采用结构化的方法管理工作流的定义、调度和执行，它包含简单的脚本语言SwiftScript[28]，SwiftScript可以用来简洁地描述基于数据集类型和迭代的复杂并行计算[29]，同时还可以对不同数据格式的大规模数

据进行动态的数据集映射。运行时系统提供一个高效的工作流引擎用来进行调度和负载均衡，它还可以与PBS和Condor等资源管理系统进行交互，完成任务执行。

图2为Swift系统架构，由4个主要组件组成：工作流定义、调度、执行、资源供应。使用简单高效的脚本语言SwiftScript定义计算，SwiftScript脚本被编译成抽象的计算计划，然后被工作流引擎调度到分配的资源上执行。Swift中的资源配置非常的灵活，任务可以被调度到多种资源环境中执行，资源供应者的接口可以是本地主机、集群环境、多站点网格环境或Amazon EC2服务。

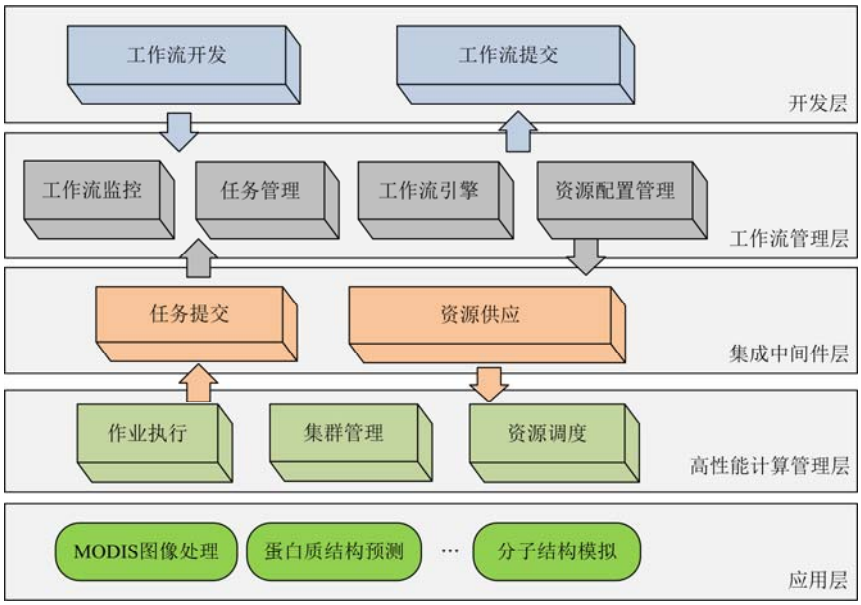

图1　集成参考架构

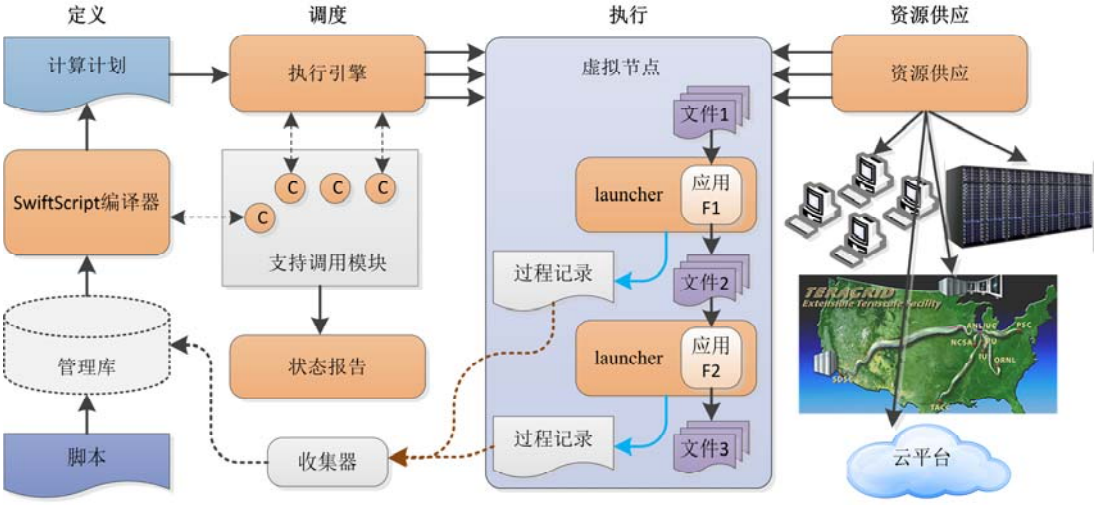

图2　Swift架构

2.3　Windows HPC Server架构

Windows HPC Server[30]可为以超级计算机为主的HPC环境提供企业级的工具、性能和伸缩性，而且是一个完整、综合的集群环境，包含操作系统、HPC工作调度器、消息传递接口第二版(MIP2)支持、集群惯例和监视、分布式Excel计算能力、空闲Windows 7系统工作站利用能力等等。

Windows HPC Server集群架构由一系列节点、组件、服务及接口组成。集群中关键组件包括Head Node、Compute Node、Job Scheduler和Broker Node(用于支持SOA集群)：

1) Head Node：作为管理单元，对集群进行作业调度。它提供了故障转移和控制，并调节集群资源访问。

2) Compute Node：执行需要执行的计算任务，这些任务由作业调度器分配到计算节点中。

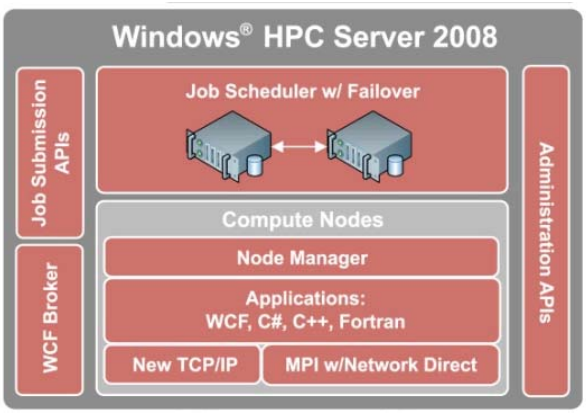

图3 Windows HPC Server架构

3) Job Scheduler：将作业和其相关的任务进行排队，它给这些作业分配资源，在计算节点上加入新的任务，并且对作业、任务和计算节点进行状态监控。

4) Broker Node：在应用程序和服务之间扮演中介的角色，代理对服务进行负载平衡，最后将结果返回到应用程序。

2.4 Swift与Windows HPC Sever集成架构

Swift工作流管理系统提供结构化的方法管理工作流的定义、调度和执行；Windows HPC提供基于Windows平台的集群管理、任务管理、任务调度等机制和开发接口。将Windows HPC Server与Swift工作流并行计算系统映射到集成参考架构中，可以实现Windows平台上的大规模并行计算与工作流应用，相应的实例集成架构描述如图4所示。

工作流开发层：提供工作流定义脚本SwiftScript的开发环境，并提供接口用于提交工作流。

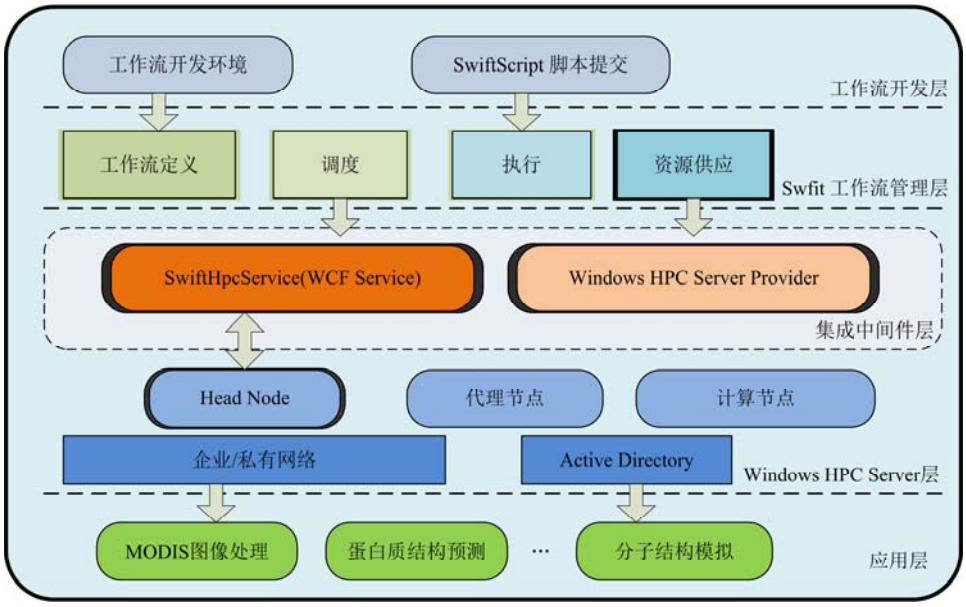

图4 Swift与Windows HPC Server集成架构

Swift工作流管理层：Swift通过解析脚本语言SwiftScript的工作流流程定义、数据调用和配置信息，工作流引擎将整个工作流任务进行分片，并通过定制的接口与集成中间件层进行交互，在任务调度器的调度下，使得数据处理在集群中并行执行。

集成中间件层：该层中包含Windows HPC Server Provider和SwiftHpcService两个组件。Swift提供的Provider接口可以实现和各种资源管理器和任务调度器相互协作的功能。Provider接口定义了跟任务运行相关的一些功能，包括任务提交、任务结束、任务取消和获取任务状态等。Windows HPC Server Provider是针对Windows HPC Server平台的Provider接口的具体实现。本文开发了基于Windows Communication Foundation (WCF) Service的SwiftHpcService服务并部署在集群的Head Node中，Windows HPC Server Provider组件通过调用SwiftHpcService提供的相应服务，将任务提交到Windows HPC计算集群中，Windows HPC Server层根据服务配置信息返回计算任务状态给Swift工作流管理层。

Windows HPC Server层：提供了完善的Windows平台上的集群管理、任务管理、任务调度等机制和开发接口，通过定制的接口与集成中间件层进行交互，Job Scheduler组件调度由中间件层提交的计算任务，分配相应的计算资源，完成工作流任务执行。

应用层：主要是描述可以在此集成方案的架构下运行的高性能并行计算应用，如MODIS图片处理、蛋白质结构预测和分子结构模拟等。

3　实验与数据分析

本文通过NASA MODIS图片处理工作流分析并验证Swift工作流系统与Windows HPC Server集成的功能。输入数据为120个大小为5.5 MB左右的卫星航拍数据块，数据块中含有水域、沙地、绿地和城市等地质特点，计算这些数据块中城市面积最大的前12个地区。

3.1　实验配置

使用5台计算机，其中包括1台Swift Client、1台HPC Head Node和3台Compute Node，其中Head Node中还部署有Broker Node、Active Directory服务器和NFS服务器端。集群环境和节点配置如图5所示。

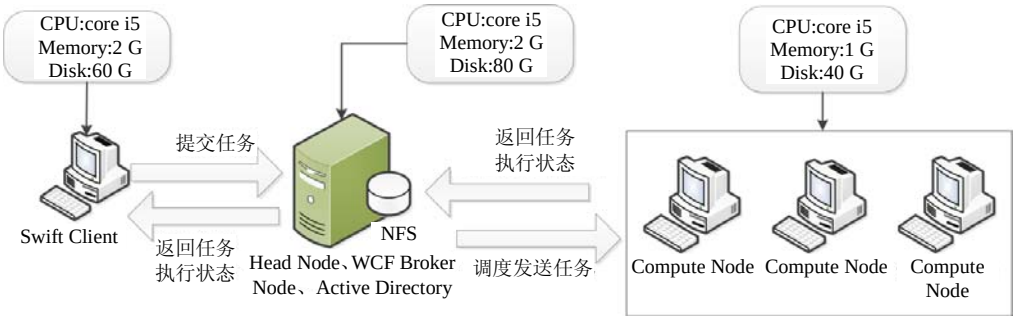

图5　集群环境配置

3.2　实现过程

实验中通过图片的像素和颜色计算城市的面积，再获取面积最大的前12个地区。首先将图片文件存储在NFS共享文件系统中，配置Swift与Windows HPC Server交互接口；然后执行SwiftScript工作流脚本NASA MODIS图片处理工作流为：

```
app (landuse output) getLandUse (imagefile input, int sortfield){
    getlanduse @input sortfield stdout=@output ;
    }
    app (file output, file tilelist) analyzeLandUse (landuse input[], int usetype, int maxnum){
    analyzelanduse @output @tilelist usetype maxnum @filenames(input);
    }
    app (imagefile output) colormodis (imagefile input)
    {
      colormodis @input @output;
    }
    app (imagefile output) assemble (imagefile input[]){
    assemble @output @filenames(input);
    }
    Imagefile geos[]<filesys_mapper;location="\\headnode\tmp\modisdata_s",suffix= ".tif">;
    Landuse land[]<structured_regexp_mapper;source=geos,match="(h..v..)",transform= "landuse/\\1.landuse.byfreq">;
    foreach g, i in geos {
    land[i] = getLandUse(g,1);
    }
    imagefile urbanMontage<single_file_mapper; file=@strcat(odir,"urban.png ") >;
    urbanMontage = assemble(recoloredImage);
    int N = 12;
    int UsageTypeURBAN=13;
    (bigurban, urbantiles) = analyzeLandUse(land, UsageTypeURBAN, N);
```

如图6所示，Swift能够根据输入目录下的modis数据文件的数量，自动动态地将工作流解析成为对这120个图片进行处理的执行计划，并把并行的任务发送到Windows HPC Server的Head Node，然后Job Scheduler根据资源使用情况为任务分配计算资源。卫星云图加载后，对每张图片的计算被识别为任务并提交给getLanduse接口进行城域面积的分析，然后提交给analyzeLandUse接口对图片中的陆地部分进行进一步的分析和计算，得出面积最大的12张图片，将其文件名列表存入urbantiles文件中，并将其逐一转换成png文件，最后合成一张整图，如图7所示。

3.3　案例结果分析

实验过程中，可以不断向集群中动态添加计算节点，集群性能也不断提高，动态添加计算节点性能增长如图8所示。随着节点数的增加，运行时间也在相应缩短，获得的加速基本呈线性增长，且接近

理想值(虚线为理想加速值，实线为实际加速值)。同时，随着节点的不断增多，性能的增加趋于平缓，理想加速值与实际加速值的差不断扩大，在集群规模不断扩大的情况下，节点间的通信开销和NFS作为共享文件系统所带来的开销逐渐成为制约集群整体性能提升的瓶颈。所以当处理的数据规模一定时，用户需要综合考虑任务处理规模与数据量来决定集群规模，这样才能获取更高的性价比。

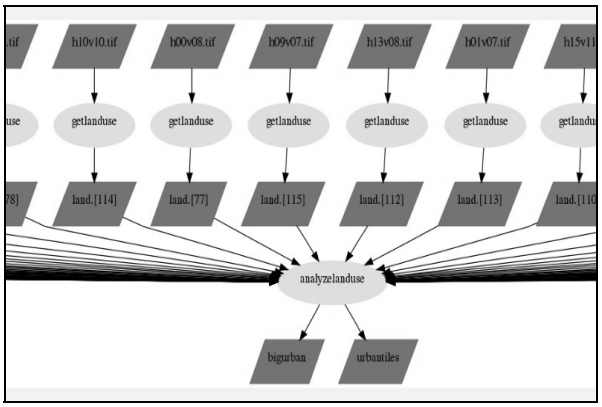

图6 工作流计算流程(局部)

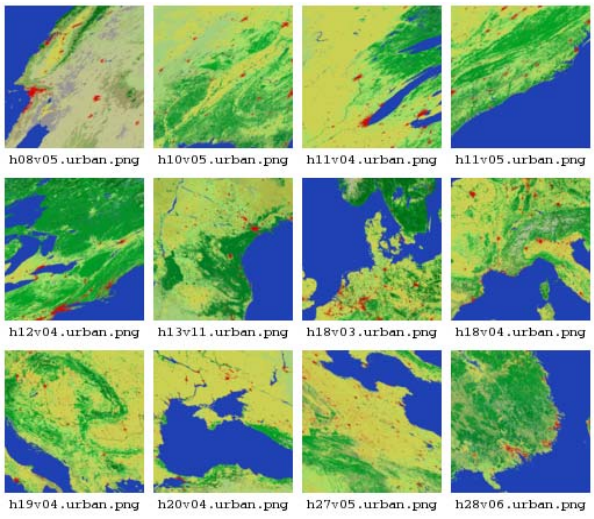

图7 工作流计算结果

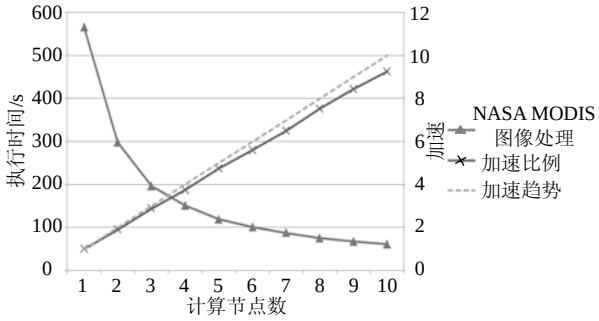

图8 图片处理与性能

通过这个应用实例，本文演示了Swift和Windows HPC Server的集成过程，集成的成功应用、Windows HPC Server本身对计算节点的调度、以及用HPC集群所获得的线性加速。同时，Windows HPC Server可以从微软Azure云平台中获取计算资源，Swift不仅可以利用Azure云平台提供的伸缩性和资源按需分配等优势，而且可以为Azure提供一个灵活的工作流应用定制前端和界面。

4 结 束 语

数据与计算的大规模化趋势对人们生活的影响越来越深入，相应的技术与概念也不断涌现，基于海量数据的计算从数据存储到并行化处理，整个过程需要不同的技术支撑，产生了许多基于不同系统与架构的解决方案。

本文提出科学工作流系统与高性能计算平台相结合的集成参考架构，实现高性能计算平台上的大规模并行计算，在提供资源管理和集群调度的同时，为用户提供方便的应用定制和管理前端，实现对大规模HPC应用的端到端的支持。以Swift工作流系统与Windows HPC Server集成的方案为例，通过NASA MODIS图片处理工作流来分析并验证集成方案的可行性和性能，以及对应用的线性加速效果。Swift系统和Windows HPC Server的有效集成能促进更多领域、更大规模的HPC应用运行在Windows的集群和云平台环境中。同时，集成参考架构的提出能够为规范化工作流系统与高性能计算平台的集成研究提供参考与实例，结合工作流系统与高性能计算的优势与特点以应对科学计算日益增长的规模与复杂度。

在后期的研究工作中会考虑使用更高效的分布式文件系统来进行数据的存储，突破由NFS所带来的性能瓶颈。同时，在现有统一集成框架的基础上，进一步研究工作流系统与其他高性能计算平台集成的实现，实现高性能计算平台上的大规模并行计算与应用流程管理等功能。

参 考 文 献

[1] ROGERS S. Big data is scaling BI and analytics[J]. Information Management, 2011, 21(5): 14.

[2] BELL G, HEY T, SZALAY A. Beyond the data deluge[J]. Science, 2009, 323(5919): 1297-1298.

[3] Conseil Européen pour la Recherche Nucléaire(CERN). Large Hadron Collider[R/OL]. [2012-03-02] http://lhc.web.cern.ch.

[4] National Center for Biotechnology Information(NCBI). GenBank Overview[R/OL]. [2012-03-03]. http://www.ncbi.nlm.nih.gov/genbank

[5] HULL D, WOLSTENCROFT K, STEVENS R, et al. A tool for building and running workflows of services[J]. Nucleic

Acids Research, 2006, 34(suppl 2): 729-732.
[6] LUDÄSCHER B, ALTINTAS I, BERKLEY C, et al. Scientific workflow management and the Kepler system[J]. Concurrency and Computation: Practice and Experience, 2006, 18(10): 1039-1065.
[7] FREIRE J, SILVA C T, CALLAHAN S P, et al. Managing rapidly-evolving scientific workflows, provenance and annotation of data[J]. Lecture Notes in Computer Science, 2006(4145): 10-18.
[8] DEELMAN E, SINGH G, SU MH, et al. Pegasus: a framework for mapping complex scientific workflows onto distributed systems[J]. Scientific Programming, 2005, 13(3): 219-237.
[9] ZHAO Y, HATEGAN M, CLIFFORD B, et al. Fast, reliable, loosely coupled parallel computation[C]//2007 IEEE Congress on Services. Salt Lake City: IEEE Computer Society, 2007.
[10] LIN C, LU S Y, LAI Z Q, et al. Service-oriented architecture for view: a visual scientific workflow management system[C]//Proc of the IEEE 2008 International Conference on Services Computing (SCC). Honolulu: IEEE Computer Society, 2008.
[11] 罗海滨, 范玉顺, 吴澄. 工作流技术综述[J]. 软件学报, 2000, 11(7): 89-90.
LUO Hai-bin, FAN Yu-shun, WU Cheng. Overview of workflow technology[J]. Journal of Software, 2000, 11(7): 89-90.
[12] HOLL S, ZIMMERMANN O, HOFMANN-APITIUS M. A unicore plugin for hpc-enabled scientific workflows in taverna 2.2[C]//IEEE Congress on Services (SERVICES). Washington D C: IEEE Computer Society, 2011.
[13] MAHESHWARI K, GOBLE C, MISSIER P, et al. Medical image processing workflow support on the EGEE grid with Taverna[C]//IEEE International Symposium on Computer-Based Medical Systems (CBMS). Albuquerque: IEEE, 2009.
[14] TAN W, MADDURI R, NENADIC A, et al. CaGrid workflow toolkit: a taverna based workflow tool for cancer grid[J]. BMC Bioinformatics, 2010, 11(1): 542.
[15] Windows Workflow Foundation(WWF). Windows Workflow Foundation Introduction[R/OL]. [2012-03-02]. http://www.windowsworkflowfoundation.eu/
[16] 杨利国. 基于WF工作流技术研究及应用[D]. 武汉: 武汉理工大学, 2008.
YANG Li-guo. Research and application based on windows workflow foundation technology[D]. Wuhan: Wuhan University of Technology, 2008.
[17] ZAPLETAL M, Van der Aalst W M P, RUSSELL N, et al. An analysis of windows workflow's control-flow expressiveness[C]//Seventh IEEE European Conference on Web Services. Eindhoven: IEEE Computer Society, 2009.
[18] PAVENTHAN A, TAKEDA K, COX S J, et al. Leveraging windows workflow foundation for scientific workflows in wind tunnel applications[C]//22nd International Conference on Data Engineering Workshops. Atlanta: IEEE Computer Society, 2006.
[19] BARGA R, JACKSON J, ARAUJO N, et al The trident scientific workflow workbench[C]//Fourth International Conference on Science. Indianapolis: IEEE Computer Society, 2008.
[20] BARGA R, JACKSON J, ARAUJO N, et al. Trident: scientific workflow workbench for oceanography[C]//IEEE Congress on Services - Part I. Honolulu: IEEE Computer Society, 2008.
[21] SIMMHAN Y, BARGA R, Van INGEN C, et al. Building the trident scientific workflow workbench for data management in the cloud[C]//Third International Conference on Advanced Engineering Computing and Applications in Sciences. Sliema: IEEE Computer Society, 2009.
[22] BODE B, HALSTEAD D M, KENDALL R, et al. The portable batch scheduler and the maui scheduler on linux clusters[C]//ALS'00 Proceedings of the 4th Annual Linux Showcase & Conference. Berkeley: USENIX Association, 2000.
[23] BERMAN F, FOX G, HEY A J G. Grid computing: making the global infrastructure a reality[M]. Chichester: John Wiley & Sons, Ltd, 2003.
[24] FOSTER I. Globus Toolkit version 4: software for service-oriented systems[J]. Journal of Computer Science and Technology, 2006, 21(4): 513-520.
[25] ZHAO Y, LI Y F, TIAN W H, et al. Scientific-workflow-management-as-a-service in the cloud[C]//2012 Second International Conference on Cloud and Green Computing (CGC). Xiangtan: IEEE Computer Society, 2012.
[26] NEHRBASS J, SAMSI S, CHAVES J C, et al. Interfacing PC-BASED MATLAB directly to HPC resources[C]// HPCMP Users Group Conference. Denver, Colorado: IEEE, 2006.
[27] XIA Y J, SHI X M, KUANG L, et al. Parallel geospatial analysis on windows HPC platform[C]//2010 International Conference on Environmental Science and Information Application Technology (ESIAT). Wuhan: IEEE, 2010.
[28] WILDE M, HATEAGN M, WOZNIAK J M, et al. A language for distributed parallel scripting[J]. Parallel Computing, 2011, 37(9): 633-652.
[29] ZHAO Y, DOBSON J, FOSTER I, et al. A notation and system for expressing and executing cleanly typed workflows on messy scientific data[J]. ACM SIGMOD Record, 2005, 34(3): 37-43.
[30] Microsoft. HPC - Technical Overview of Windows HPC Server 2008 R2 [R/OL]. [2012-03-01]. http://www.microsoft.com/download/en/details.aspx?id=434

编 辑 税 红