

文章编号: 1001 - 2486(2011)03 - 0056 - 10

科学 workflow 技术研究综述*

张卫民, 刘灿灿, 骆志刚

(国防科技大学 计算机学院, 湖南 长沙 410073)

摘要:科学 workflow 管理系统(Scientific Workflow Management Systems, SWfMS)为科学家提供一个管理大规模复杂科学计算的有效平台,通过管理复杂应用程序对海量科学数据进行管理、分析、仿真和可视化以帮助科学家进行科学发现。本文首先对科学 workflow(Scientific Workflow, SWF)的关键技术和研究现状进行全面综述,包括 SWF 模型、表示、语言、组合、验证、调度、数据来源管理以及系统的容错性与安全性等;然后对近两年的最新研究进行分析,并在此基础上指出该领域目前存在的不足和未来发展方向;最后对我国目前的研究现状进行分析并给出建议。

关键词:科学计算;科学 workflow(SWF);流程组合;调度;容错;来源管理

中图分类号:TP393 **文献标识码:**A

A Review on Scientific Workflows

ZHANG Wei-min, LIU Can-can, LUO Zhi-gang

(College of Computer, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: Scientific Workflow Management Systems (SWfMS) has become an effective platform for representing and managing large-scale complex scientific computing processes, which glue together a large number of complex applications for data management, analysis, simulation, and visualization, and then assist the scientists for scientific discovery. In this paper, the current studies of Scientific Workflow (SWF) are reviewed firstly, including SWF models, presentations, languages, composition, validation, scheduling and provenance, as well as fault tolerance and system security. Then, the latest studies in recent years are also discussed. Based on the analysis of the key technologies and the latest development, the drawbacks in this domain are pointed out and some suggestions towards the future development are made. Finally, the recent research status quo in China is presented and some suggestions are given.

Key words: scientific computing; scientific workflow (SWF); workflow composition; scheduling; fault-tolerant; provenance

作为继理论和实验之后的第三种科研手段,科学计算已经在科研领域中发挥着非常重要且不可替代的作用^[1-2]。科学计算通常需要组合不同领域、多个组织的复杂应用程序对海量数据进行查找、移动、分析、处理及可视化等操作。作为一种辅助科学家进行科学实验的手段,科学 workflow 管理系统(Scientific Workflow Management Systems, SWfMS)对这些复杂应用程序及各程序间的数据依赖关系进行组合,并控制各部分在时间、空间以及资源等约束条件下按序完成,已经成为复杂科学计算流程管理的必要手段,有效推动了科研进展。目前,国外多个实验室在一些大型科研项目中分别开发了面向各自研究领域的 SWfMS;然而,这些系统的关键技术和具体实现各不相同,各系统间缺乏统一标准且互操作也比较困难。目前一

些科研组织和个人对这些系统的关键技术和研究现状进行了综述和比较^[3-6],但随着时间的推移和技术的发展,各个系统在原有基础上均进行了一定程度的完善和改进,每年也会涌现出一些新的技术和系统;另外,目前国内对科学 workflow(Scientific Workflow, SWF)技术的重视程度远不如欧美一些发达国家,国内针对这一领域的研究也比较少。鉴于这一现状,作者在研读大量文献的基础上,对 SWF 的最新关键技术进行全面总结、比较和评价,并对最近的研究成果和国内的研究现状进行分析,旨在为从事相关研究的科研人员提供一定的参考和借鉴。

* 收稿日期:2010-11-08

基金项目:国家 863 高技术资助项目(2006AA01A123);国家自然科学基金资助项目(609030420)

作者简介:张卫民(1966—),男,教授,博士,博士生导师。

1 SWF 的起源与特点

1.1 SWF 的起源与发展

随着网格技术和 e-Science 等资源虚拟化平台和虚拟科研环境的发展,科研过程中的数据分析环境与分析过程日益复杂,为了减少科研人员在与科研环境和与具体计算相关的工作上的精力投入,各个科研组织亟须一种类似于管理业务流程的工作流管理平台来对复杂的科学计算流程进行定义和自动化管理。SWF 技术正是在这种需求下应运而生,它作为一种促进和保障协同开发的技术路线,为科学家提供一个科学计算复杂流程定义和自动运行的管理平台。

由于各个领域大型科研项目的需求和推动,目前在国外相继出现了多个比较有影响力的 SWfMS,如: Triana^[7]、Taverna^[8]、Pesasus^[9]、Kepler^[10]、ASKALON^[11]、Java Cog Kit^[12]、P-GRADE^[13]、GridFlow^[14]、K-WF^[15]等,这些系统在借鉴 20 世纪 90 年代发展起来的业务工作流(Business Workflow, BWF)的基础上,均针对各自的研究领域分别开发了适合于本领域科学计算和科学实验的流程管理系统。然而,遗憾的是目前并不存在一个适合于大部分科学计算特征的通用型 SWfMS,各个系统针对不同的应用背景进行设计和实现,在应用到新的领域时通常需要不同程度的修改。此外,对于 SWF 技术,目前国际上仍不存在一个统一的定义,这也从某种程度上反映了该研究领域的不成熟。尽管如此,随着 e-Science、e-Research 等虚拟科研环境逐渐融入科研活动中,SWF 也逐渐成为计算机应用领域的研究热点^[3-5,16-17]。

1.2 SWF 与 BWF 的区别

SWF 起源于业务工作流,与 BWF 不同的是,SWF 面向数据,并具有数据集成、计算集成、分析集成的特点。相对于传统的 BWF,SWF 技术目前还处于很不成熟的阶段,与 BWF 相比,SWF 系统具有如下一些特点。

首先,SWF 面向数据,而 BWF 面向控制。科学计算的一个显著特点是数据集成特征,计算过程中需要频繁地对大规模数据集进行操作,并按照一定的规则调用特定的应用程序在分布式计算资源上对这些数据进行处理、分析、移动或存储,其依赖关系着重表达了各任务间从数据生产者到数据消费者之间的数据流。在面向控制的 BWF 中,工作流活动之间的依赖主要表达了多个任务之间的逻辑结构和执行方式,如顺序、并行、条件

或循环等。

其次,SWF 具有大规模特征。在目前各个科研领域,如宇航学、天体物理、生物、核模拟、地震检测、气象、高能物理等,科学家需要组合成千上万的应用程序对大量的原始观测数据或中间数据进行分析、处理、仿真以及可视化操作,需要处理的数据量也多达几百 GB 甚至达到了 PB 量级。随着科学研究方法、实验设备以及实验环境的不断发展,科学实验的规模还将变得越来越大,而 BWF 的复杂程度及规模远远小于 SWF。

第三,SWF 系统需要支持科学家进行探索性的流程设计与修改。各个科学家根据自己的实验需求进行工作流流程设计,由于 SWF 的大规模特性,其执行阶段可能持续几小时到几个月甚至几年不等。科学家需在工作流的执行过程中根据部分中间结果对工作流流程进行动态修改,如参数设置、组件替换、数据更新或实验流程修改等,因此 SWF 系统需支持用户进行动态增量式的流程设计与修改。此外,进行实验流程设计的科学家往往不是专业的流程设计人员,因此 SWF 系统需提供灵活的简单易用的流程设计工具。而在传统的 BWF 中,专业的流程设计人员将业务流程编写好并通过测试之后便提交给工作流引擎进行执行,业务流程在较长一段时间内保持不变,因此 BWF 系统较少对执行过程中的动态变化因素提供支持。

第四,SWF 具有动态适应性。SWF 运行在网格环境上,因此需具有动态适应性以支持工作流执行过程中的动态资源绑定。SWF 在定义期对任务、任务间的逻辑关系以及逻辑数据依赖关系进行描述,这种描述往往是不完整的,在执行过程中需根据工作流的运行态势与资源的即时状态对定义期的流程描述进行动态修改。因此目前的 SWF 一般支持多层抽象,工作流信息在不同层以不同方式进行显示。BWF 主要关注工作流执行过程中的 ACID (Atomicity, Consistency, Isolation, Durability) 特性^[18],致力于保证工作流系统在存在并发访问以及某些活动需要作为一个整体完成时的正确性。

第五,SWF 需提供数据与工作流流程的来源管理。科学实验结果的一个重要特点是可验证性,实验过程以及实验结果必须具有可重复性。对于某个数据产品,SWfMS 需要自动记录原始数据的来源以及数据的演化过程。数据的来源管理也为科学家对数据产品进行质量评价、可靠性审核、数据共享和重组等提供重要信息。

第六,从用户的易用性与易操作性等角度来说,用户在 SWF 的执行过程中希望可以从流程中的任意任务开始重新执行,而不是每次都需从头重新开始,因此 SWF 需要满足用户方便灵活的重运行要求。

此外,在 SWF 中实现与 Web 服务的无缝集成,与遗传代码的无缝整合,并提供基于门户的访问方式都是 SWF 技术中需要研究的内容,而 BWF 主要面向文件处理、任务管理和流程控制,更加强调工作流的安全性和事务性等特征。由此可见,

SWF 虽然来源于 BWF,但其研究要点与 BWF 之间存在很大差别。

2 SWF 的关键技术与研究现状

SWfMS 支持科学家对复杂计算流程进行描述、修改、运行、重运行和监控,使用科学逻辑自动控制 workflow 任务的执行顺序并保证任务在多种约束条件下顺利完成。SWF 系统的主要功能模块如图 1 所示。

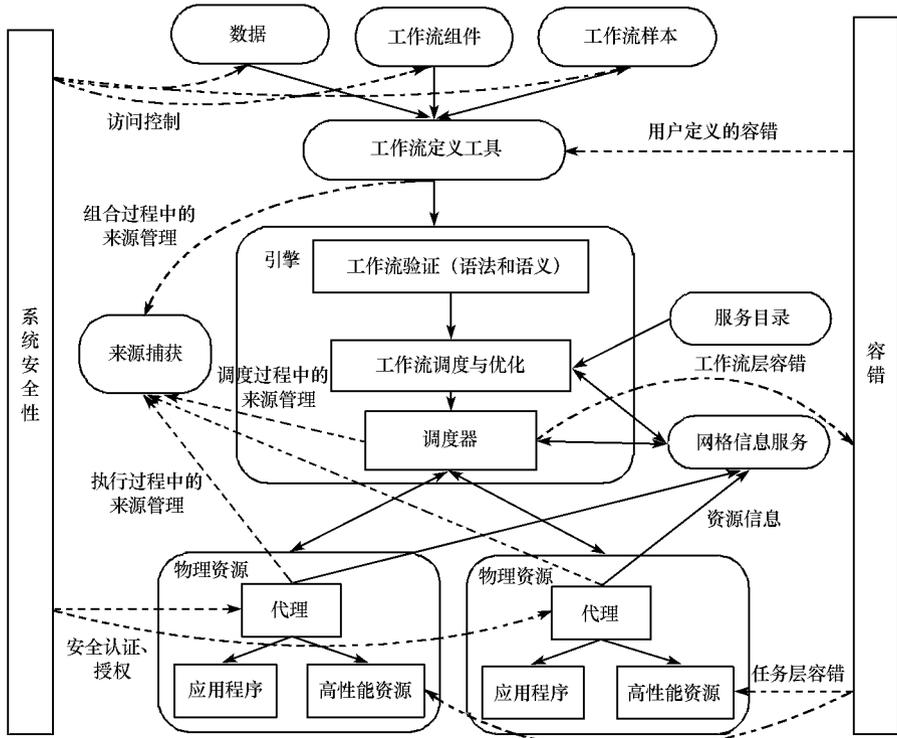


图 1 SWfMS 结构和功能
Fig.1 Architecture and functions of SWfMS

如图 1 所示,SWfMS 的功能模块主要分为流程定义和引擎两大模块。在流程定义过程中,用户通过流程定义工具从工作流样本库或工作流组件库中查找所需的样本、组件和数据,按需进行流程组合并形成抽象的 SWF 描述。这一阶段的研究内容主要集中在 SWF 模型、表示、语言、流程组合方式及流程验证等方面。在流程定义完之后,用户将 SWF 描述及用户定义的约束一起发送到工作流引擎,引擎对抽象工作流进行解析、调度、执行和监控。该过程中的研究内容主要包括工作流任务的解析、服务的查找和匹配、资源调度策略和调度算法、调度优化等。同时,在流程定义和执行过程中,需要对工作流组件、计算数据、工作流、计算资源、计算环境的元数据和来源信息进行捕获和记录,因此来源管理也是 SWF 技术中一个重

要的研究内容。另外,SWF 运行在网格环境下,由于网格环境中各种资源与生俱来的分布性、动态性和脆弱性,SWF 系统的容错性和安全性也是非常重要的研究内容。以下从 SWF 系统的各个功能模块对其主要研究内容进行研究和分析。

2.1 工作流模型

工作流管理系统中一个最重要的部分是流程建模,工作流的描述和表达、引擎的设计和实现都将随工作流模型的不同而不同。工作流模型一般可以分为面向控制、面向数据和混合模型这三类。BWF 主要关注业务处理过程中各个任务的变化,任务间的依赖关系着重于表达各工作流活动之间的控制依赖关系,如顺序、并行、选择、循环、条件结构等,因此 BWF 多采用面向控制的模型;而 SWF 主要关注被分析的数据在科研过程中各任

务之间的变化以及这种变化为科研带来的影响,后继任务的输入数据通常依赖于前驱任务的数据处理结果,因此 SWF 多采用面向数据的模型。然而,在 SWF 中通过单一的数据流进行控制无法满足 SWF 的流程控制需求,如无法直接表达任务同步、循环等结构,也需要一些控制结构来辅助控制计算流程,因此 SWF 中一般采用面向数据与面向控制的混合模型,主要由数据进行驱动,同时也提供一些简单的流程控制结构。

2.2 SWF 表示

在 SWF 中需要对工作流中的任务以及任务间的逻辑关系进行表示,其表示方法大体可以分为以下三类:基于有向图、基于 Petri 网以及基于 UML 的表示。

2.2.1 有向图

由于有向图具有简单与直观的特点,且能表达工作流图中节点间的依赖关系,因此有向图常被用于表达工作流。基于有向图的 SWF 表示可以分为基于有向无环图(DAG)和基于有向有环图(DCG)的表示,UNICORE^[19]、Condor DAGMan^[20]、Cactus^[21]、BPEL4WS^[22]、Scufl^[23]、GPEL^[24]中采用了基于 DAG 的表达式,能支持顺序、并行、条件等控制结构。与 DAG 不同的是,DCG 支持循环结构,在 Triana 与 GridAnt^[25]中采用了这种表示方式,能支持顺序、并行、条件、循环等控制结构。

2.2.2 Petri 网

Petri 网是一种严格的过程形式化体系,具有精确的定义,因此避免了其他非形式化语言的模糊性、不确定性和矛盾性。与基于 DAG 的表达式相比,基于 Petri 网的模型具有更强的表达能力,能表达顺序、并行、循环、条件等控制结构,也能表达工作流任务的状态,因此在面向控制的工作流系统中较常采用基于 Petri 网的表达。比较典型的有 YAWL^[26]、GJobDL^[27]以及 GWorkflowDL^[28]。

2.2.3 UML

UML 是一种定义良好、易于表达且功能强大的面向对象标准化建模语言,它能表达不同活动之间的依赖关系,因此也适用于工作流建模。ASKALON 中采用基于 UML 的模型来对工作流进行描述^[29]。

2.3 SWF 语言

SWF 语言作为连接工作流表示与工作流引擎之间的桥梁,将流程中的任务和任务间的依赖关系表示成工作流引擎可以理解的形式化描述语言,不同系统对工作流流程的描述方法各不相同,

由于 XML 的规范性和可扩展性,目前 XML 在因特网上已经得到了广泛应用,如 Condor DAGMan 中的 DAX^[20]、Taverna 中的 Scufl^[42]、Askalon 中的 AGWL^[30]、Kepler 中的 MoML^[31]、GriPhyN 中的 DGL^[32]、LEAD^[33]中的 GPEL^[34],以及 BPEL4WS^[22]、SSDL^[35]、SWFL^[36]等。遗憾的是,虽然这些语言都采用了基于 XML 的技术进行开发,但是由于各种语言定义的范式和采用的标准各不相同,在多种语言间直接进行互操作也比较困难,一般情况下需针对一种语言实现一个引擎,而其他的引擎无法理解该语言所描述的工作流流程。在诸多标准各异的工作流描述语言中,BPEL4WS 是最有望成为 SWF 标准化语言的竞争者之一^[37]。

2.4 流程组合方式

按照用户在流程组合过程中所借助的工具以及流程组合的复杂程度,可以将流程组合方式分为基于文本、基于图以及基于语义的组合。

2.4.1 基于文本的组合

基于文本的组合是一种最简单的流程组合方式。BPEL4WS、SCUFL、DAGMan 和 GridFlow 中都支持这种基于文本的流程组合方式,用户完全通过手工方式来对文本进行编辑,这种方法对用户的要求比较高,且在组合复杂计算流程时容易出错。

2.4.2 基于图形的组合

为了减轻用户手工编辑流程的负担,目前的 SWfMS 一般提供基于图形的流程编辑方式,如 Triana、Kepler^[10]、VisTrails^[38]、P-GRADE、XBaya^[39]等,用户对工作流样本库或工作流组件进行查找,通过拖拽的方式将合适的组件移动到工作流面板中,并通过边将这些组件进行连接以表达任务间的依赖关系。在用户进行流程图图形化编辑的过程中,系统自动将流程更新且保存为相应的文档,并对工作流的数据接口进行语法验证和检查。

2.4.3 基于语义的组合方式

基于语义的流程组合方式是一种自动化层次较高的流程组合方式,采用基于本体的描述语言 OWL 对服务、组件和数据进行描述,并通过智能规划和自动推理技术来支持流程表达,使得工作流的表达非常清晰且具有较高的可读性。基于语义的流程组合方式在较高的抽象层上对工作流流程进行描述,通过元数据的语义描述来支持分布数据的组合,在工作流中使用语义表达来传递元数据,用户不需要对与具体计算相关的细节进行描述,而是把精力集中到实验流程的设计与规划上,大大降低了科学家进行流程定义的难度,也有

效减轻了科学家的负担。当 SWF 的作业规模达到几千个任务时,在流程设计和执行过程中采用语义技术来为科学家提供帮助是必然趋势。目前支持语义组合的工作流系统有 K-WF、Taverna、Pegasus、ICENI 等。

2.5 workflow 验证

workflow 验证是 workflow 管理系统的一个重要功能,也是流程定义过程中的一个重要环节,workflow 验证主要包括语法验证和语义验证这两方面。语法验证主要对 workflow 的结构以及数据类型进行验证,workflow 的结构验证包括检测流程定义中的死锁、活锁、不可达任务或无效任务等;数据类型验证检查数据生产者的输出数据类型是否与数据消费者的输入数据类型相匹配等。语义验证比语法验证复杂得多,有些语义错误需要在 workflow 执行过程中才能被检查出来,如在条件分支中如果没有覆盖所有情况,当出现其它未被定义的情况时,workflow 系统可能会发生异常,而这类异常通常需要在执行过程中才能被检测出来。

2.6 workflow 调度

作业调度是 SWF 技术中一个非常重要且复杂的问题,调度性能的好坏直接影响到 SWfMS 的性能。作业调度是对 workflow 结构进行分析,对调度环境中各网格资源的性能进行评估,并在满足 workflow 作业约束以及用户约束的条件下,将作业合理地分配到各资源的过程。国内外大量的研究组织对这一问题进行了深入的研究,针对不同的应用场景分别提出了多种调度算法^[40-42],也有一些研究组织对调度策略和调度算法进行了总结^[4,43-44]。由于篇幅有限,本文不做详细分析。

2.7 来源管理

SWF 中的数据来源管理是近几年兴起的一个非常重要的研究领域,在对数据产品进行质量评价、来源审核、结果验证、数据共享和重组、workflow 失败原因诊断、分析服务或数据更新所产生的影响等过程中都发挥了重要作用,大多数的研究组织都对这一领域进行了深入研究^[45]。

SWfMS 中的来源管理主要分为 workflow 来源管理与数据来源管理两方面,workflow 来源管理对 workflow 生成和设计过程进行记录,包括记录 workflow 的版本号、由谁设计、在哪个 workflow 版本上进行修改或是采用了哪些版本的工作流组件等,在 workflow 流程进行重用和共享的过程中可以通过查看 workflow 的相关来源信息查看 workflow 的演化过程,进而对 workflow 的正确性和有效性进行评估;数据

来源管理是指对数据产品的来源信息进行记录,包括采用的源数据,数据在演化过程中所产生的中间数据,生成数据的工作流及 workflow 版本,workflow 中的组件及版本,各组件参数设置、执行主机、执行环境以及执行库的版本等。根据这些记录的来源信息可以对实验结果进行重复验证,也可以通过查看来源数据来确定实验结果是否可靠和正确,实验过程和实验条件是否合理等。

2.8 容错

在 SWF 长期的执行过程中,计算资源、应用程序、底层服务或网络都可能会遇到一些可预测或不可预测的异常,如不及时对这些异常进行处理,将可能阻碍 SWF 的正常执行。为了提高系统的可靠性,最大限度上保证流程的顺利完成,应尽量完善 SWfMS 的容错功能。各系统都提供了各自的容错策略^[46,11,8,47],本文不进行详细分析。

2.9 安全

在网格环境上运行的 SWfMS 往往跨越了多个领域和多个组织,在这些组织间共享数据和计算资源时需要为用户和资源所有者提供安全保障,因此在 SWF 中研究系统安全也是一个非常重要的课题。一般情况下,科学家都不愿意无条件地轻易共享他们数据,剑桥的一位基因学家说:“科学家宁愿共享他们的牙刷,也不愿意共享数据”^[48],由此可见,由于某种原因,科学家往往希望在一定范围内共享数据,而不是将数据进行无条件的全局共享,因此必须为科学家提供一个安全可靠的数据共享环境。除此之外,计算资源、实验设备等都分布在网格环境中,这些资源所有者同样希望对用户进行权限控制,因此对计算资源和实验设备等进行安全访问控制也是一个需要研究的课题。

3 SWF 的最新研究

近两年的 SWF 研究主要集中在 SWF 新应用的开发、新体系结构的研究和原有多个系统的互操作上。一些新的 SWF 系统不断涌现出来,这些系统大多是在应用背景的推动下对已有系统进行改造来解决新的问题。由于 Kepler 具有通用性与易改造等特点,目前很多新系统在 Kepler 的基础上进行二次开发,如 NCAR 为 CCSM (Community Climate System Model)^[49] 开发的工作流系统对 Kepler 的功能进行了继承和扩展,添加了一些新的组件如 MyProxyLogon、GridFTP2g、RSLGenerator 等,主要解决了用户在远程计算资源上进行作业

提交时,第三方支持软件以及系统环境变量的自动集成过程,也解决了在不同系统层面上收集来源信息的问题,目前该系统已经应用在 TeraGrid 上管理 CCSM 的执行过程。McPhillips 等人也在 Kepler 的基础上实现了一个新的 SWF 与数据管理框架 COMAD^[50],它是一个面向数据集的模型与设计方法,解决了 SWF 中的良定义性(能简单地设计定义良好且有效的工作流)、简明性(能创建具有自明性的工作流)、可预测性(在运行之前能简单预测工作流的工作)、可记录性(自动记录工作流在执行期间的工作)、可重用性(将已有工作流进行组合形成新的工作流)、科学数据模型(用户能表达数据结果的意义)以及自动优化(对工作流的性能进行优化)等要求,该框架在 pPOD 项目中得到了实现^[51]。此外,还有为 REAP(Real-time Environment for Analytical Processing)项目开发的工作流管理系统,它将 Kepler 应用于海底电缆传感器工作流分析和数据存档,实现对海洋数据有效、透明的实时事件探测^[52]。中科院的地理与自然资源研究所也基于 Kepler 开发了一个处理二氧化碳溢出数据的工作流系统^[53]。

除了面向具体应用的系统开发外,近两年也出现了一些新的通用型工作流系统,如微软工作组的 Barga 等人提出的 Trident^[54]是在 BWF 系统的基础上开发而来的一个通用型工作流系统。Trident 基于 Windows 工作流进行开发,提供一个基于门户的访问方式,用户可以通过门户来调度和管理工作流的执行,该系统致力于减少在高性能计算资源上运行大规模计算流程的复杂性。Trident 目前已经作为一个开源平台应用于 e-Science 项目研究,并与 Pan-STARRS 等项目进行了合作。Lin 等人针对 SWF 的需求和特点提出了一个 SWfMS 的参考模型 View,它基于 SOA 进行松散耦合的设计,参考模型由流程设计工具、工作流表达与可视化工具、工作流引擎、执行监控系统、数据管理子系统、来源管理子系统以及任务管理子系统这 7 个主要的功能模块组成,并对这些功能模块之间的 6 个接口进行了描述^[55],在目前 SWF 标准极不统一的情况下该工作是一个很好的尝试。

在众多的 SWF 系统中,各个系统都有各自的优缺点,因此如何将多个系统的优点进行融合也是近两年涌现出来的一个新的研究热点。Tan 等人在基于服务的网格框架 caGrid 中融合了 Taverna 与 BPEL 的优点开发了一个工作流管理系统^[56],Taverna 具有面向数据的特点,且对于流程

组合时的服务发现以及执行阶段的结果分析也提供了强大支持;而 BPEL 提供了丰富的控制结构,使得在表达复杂的过程逻辑时非常灵活,其调度引擎由于运行在专门的应用服务器中,对工作流运行过程中的运行状态提供了持久化存储,因此具有较强的可靠性和可扩展性,这对于具有长期运行特点的计算集成实验过程是非常重要的。caGrid 结合了 Taverna 与 BPEL 的优点,通过多阶段流程设计的方法将 BPEL 与 Taverna 进行集成,科学家使用 Taverna 对实验流程进行简单直观的设计,caGrid 系统自动将实验流程转化为标准的 BPEL 模型,并提交给 BPEL 引擎进行执行。Mandal 等人^[57]也提出了类似的方法来集成 Kepler 与 Pegasus,Kepler 为用户提供一个直观的图形用户界面来进行流程设计,而 Pegasus 中的作业管理系统 DAGMan 则提供了强大的执行期作业调度和容错功能。系统融合 Kepler 与 Pegasus 的优点,使用户能在 Kepler 的图形用户环境中生成 Pegasus 类型的抽象工作流,而调度过程由 Pegasus 进行实现。在英国的 e-Science 项目中建立了一个 Link-up 项目,该项目中集成了 myGrid、Kepler 以及 Wings/Pegasus 等多个系统的优点来提供一个功能比较强大的 SWfMS。此外,在多个系统之上开发一些专门的工作流框架来提供一个整合所有不同工作流系统的框架,并提供这些工作流系统之间的互操作,也是一种解决办法,如 Telescience^[58-59]与 BIRN^[58,60]。

4 存在的问题与发展趋势

4.1 存在的主要问题

虽然 SWF 的概念早在 1996 年已经由 Vouk 等人提出,但是 SWF 技术在近几年才逐渐成为研究的热点,目前仍处于不太成熟和完善的阶段,在将 SWF 应用于管理大规模科学计算和帮助科学家进行科学发现的时候,目前还面临不少问题和挑战。总结来看,主要存在的问题有如下几个方面。

(1)各系统缺乏统一的标准,系统间的互操作比较困难

最为明显的问题是目目前 SWF 中尚缺乏一个被学术界认可的标准,不管是 SWF 模型、表示、语言,还是执行引擎,各系统都针对各自不同的应用领域和底层资源采取不同的技术和结构进行开发。此外,在不同的系统中也实现了一些辅助工具来帮助用户进行流程设计、作业调度、来源管理和状态监控等,各系统中所强调的技术和功能各不相同,用户基于已有技术进行新系统开发时,很

难选择一个合适的原型系统;由于各系统的标准不统一,在多个系统间实现系统互操作就目前来说还比较困难。目前采取的方式是在选定的两个或少数几个系统中建立一个类似于适配器的功能部件,将某个工作流系统中定义的工作流描述自动转化为另一个系统能理解的语言,在目前工作流标准极不统一的情况下,通过适配器机制在少数几个系统中实现系统互操作是一个比较有用的尝试。然而,要实现多个系统间的互操作,就需要一个标准化组织来对 SWF 的标准进行统一,该目标的实现就目前的研究现状来说仍比较困难。

(2)SWF 缺乏多层抽象表达能力,无法同时满足多类用户的不同用户体验

SWfMS 需为不同层次的用户提供不同的用户体验,如就领域专家来说,他们希望将精力专注于与具体领域相关的研究上,而不是与具体计算相关的底层资源的收集或作业的分配和提交上;同时他们熟悉并习惯了传统的基于命令行或脚本方式的作业提交方法,并不愿意花太多的时间和精力来学习和使用其他复杂工具。对于这类专家,系统需要提供一个简单易用的基于功能的流程定义工具,使领域科学家能在较高抽象层上对工作流进行描述和定义。然而,对于与具体计算相关的科学家,如工作流系统设计者、网络设计者以及资源管理者来说,他们希望了解工作流执行过程中的具体信息,如服务所部署的资源,任务执行过程中所需的计算环境,任务调度的资源等。对于这类用户,SWfMS 需要提供与具体计算相关的描述信息,提供给用户查看底层资源与服务信息的方式。因此 SWfMS 需为不同类型的用户提供不同的用户体验,并在多层抽象层上为不同类型的用户提供不同交互接口。目前 SWfMS 都能在一定程度上提供工作流的抽象化表达,但是为不同层次的用户提供不同的用户体验和用户接口就目前来说仍比较困难。

(3)与遗留代码的集成比较困难

目前各个应用领域中都存在大量的遗留代码,这些代码通过 C、Fortran、C++ 或 Java 进行编写,这些应用程序一般具有环境依赖特性,应用程序的执行往往依赖于底层硬件与软件环境,并且与编译软件或操作系统的版本相关,在 SWfMS 中如何对这些应用程序进行简单集成是一件比较困难也必须解决的事情。目前各个系统都采用了多种方法对遗留代码进行简单集成,如 LEAD 中实现了一个专门的 GFac 工具对遗留代码进行 Web 服务封装,Taverna 与 Triana 中则需提前将应用程

序进行 Web 服务化并对服务进行注册,但为各应用领域的应用程序开发者提供一种简单的服务封装和部署方法也比较困难。

(4)系统的可靠性有待提高

由于网格环境的动态性和自主性,科学计算流程在执行过程中不可避免地会发生一些故障或异常,如不及时对这些异常进行检测和处理将无法保证系统的可靠性。目前各个系统都在不同层次,如硬件层、任务层、工作流层以及系统层等提供了一些容错策略,包括重试、冗余、替换以及检查点恢复等。然而,随着流程规模的日益增大以及网格环境的日益复杂,SWF 在长期的执行过程中所产生异常的种类和来源越来越多,异常情况也越来越复杂,SWfMS 需对这些异常进行及时检测、分析和处理,以进一步提高系统的可靠性。

(5)缺乏自动有效的来源数据提取方法

科学实验的一个重要特点是实验过程的可重复性和实验结果的可验证性。目前各个 SWfMS 都提供了流程定义过程和数据演化过程中的来源数据自动捕获和记录功能,以帮助科学实验的重运行。然而,随着科学实验规模的不断扩大,在长期的实验过程中可能需要科学家进行一些手工操作;在 SWF 执行过程中也会出现一些系统无法自动捕获的事件,如电子邮件、数据库查询操作等;同时,底层资源上服务与应用程序的性能可能不断发生演化,物理资源的状态也不断发生变化,这种情况下要保证实验结果的可验证性就目前来说还比较困难。

(6)需要动态高效的调度算法的支持

SWF 具有计算集成、数据集成与大规模并行化等特点,其任务也具有较强的环境依赖特征,因此 SWF 的调度问题与传统的 DAG 调度之间存在着许多不同之处,许多经典的 DAG 调度算法在 SWF 调度中可能不太适用。同时,随着效用网格与 SOA 的融合与发展,用户的调度目标也日趋多元化,调度过程中不仅需要考虑执行时间,同时也需要考虑执行费用、系统可靠性等多种性能参数。此外,SWF 运行在动态异构的网格环境下,在 SWF 长期的执行过程中,底层资源的状态可能随时发生改变,我们需要根据资源的实时状态进行动态调度,以满足 SWF 的动态适应性需求。

(7)与门户集成比较困难

科学家们一般不愿意在本地机器上下载并安装一个新的客户端来进行流程编辑和管理,而是希望通过一个统一的访问平台对 SWfMS 进行方便快捷的访问。目前的 SWfMS 一般都架构在底

层网格资源上,用户与网格资源的交互都通过网格平台进行,将 SWfMS 集成到网格平台中对推广系统的使用和降低系统使用门槛是一种行之有效的方法。目前与门户的集成应用较成功的系统是 P-GRADE,它不仅整合了多种网格平台,包括 GT2、GT4、LHC 以及 gLite 等,同时将 SWfMS 建立在 Gridsphere 平台上,用户可以通过门户访问 SWfMS。此外,Taverna 和 K-wf 系统也在尝试与门户集成,但是更多的系统只能在专门的客户端中进行操作。

4.2 未来发展方向

由以上分析可知,目前 SWF 技术中还存在着许多问题。随着观测设备和实验条件的不断改进和完善,SWF 的规模将越来越大,主要表现为科学数据呈指数级增长,对这些数据进行分析和处理的应用程序越来越复杂,参加实验的科研人员越来越多,用于科学计算的高性能计算资源的规模也将越来越大等。SWF 将为科学家提供更好的用户体验,使得科学家能从不同抽象层次对 workflow 进行描述和监控,在为科学家隐藏复杂计算环境底层技术细节的同时为底层资源提供对应接口,而在某些特殊情况下,科学家希望对特定资源或计算环境进行细节描述,并与执行状态的 workflow 进行交互,根据计算的部分中间结果或执行状态来决定下一步的操作等,或在执行过程中对 workflow 进行动态增量式的设计与修改。在系统功能上,SWF 系统的兼容性与可靠性将进一步提高,系统能无缝整合底层不同环境下的网格资源和应用程序,根据资源的实时状态进行动态的资源层认证和协商,进而在执行过程中动态决定作业的最佳资源;当 workflow 的执行过程中发生异常时,系统对异常进行检测,并对异常类型进行判断和处理。而在系统性能方面,workflow 系统将朝着更可靠、更安全以及标准化的方向发展。

5 国内的研究现状与建议

5.1 国内的研究现状

相对于欧美一些先进国家来说,我国 SWF 技术的研究及应用尚处于比较初级的阶段,目前也陆续出现了几个 SWF 系统,如北航的 GPEL^[61],国防科大的 EPSWFlow^[62]以及武汉大学的 TotemSWF^[63]等。GPEL 在 BPEL4WS 的基础上改进而来,支持多种原子活动和多种结构化活动,与基于 BPEL 的 OMII-BPEL 系统没有太大的区别。EPSWFlow 是一个专用于气象领域中管理集合预

报流程的 SWF 系统,采用了基于 BPEL 的 workflow 管理技术,主要解决了流程定义中作业的抽象描述和执行过程中作业的服务匹配问题。TotemSWF 是一个基于对象代理数据库的 SWF 管理平台,它采用代理模型描述一系列科学计算的执行流程,使得 workflow 管理系统操作采用类似于传统数据库的方式来完成,TotemSWF 在一定程度上和规模上能提供对科学分析过程的有效管理,在提高数据来源管理效率方面效果尤为明显。然而,TotemSWF 系统试图用数据库所能提供的有限的流程设计与执行功能来代替 workflow 组合工具和工作流引擎的工作,这在一定程度上限制了科学计算流程的规模,也无法支持 SWF 的动态适应性和各层的抽象化表示,使得多个科学家在网格环境下的协调开发和共享,资源的优化配置变得比较困难。由于这些先天的局限性,目前国际上比较著名的 SWF 系统都很少采用基于数据库的设计。除此之外,国内其他相关机构的研究也都停留在研究水平,离实际的具体应用还存在较大的距离。

造成这种现状的原因是多方面的:首先,相对于发达国家来说,我国在大规模协同开发基础平台上的投入比较少。美国早在 2001 年就启动了 Science Grid 计划、GIG 计划以及 TeraGrid 等;欧盟于 2000 年和 2001 年先后启动了包括 EuroGrid、DataGrid、EGEE 以及 CoreGRID 等在内的数十个网格重大项目;英国也于 2001 年启动了 e-Science 计划。目前欧美的主要大学和科研机构间已基本形成了基于网格进行资源共享与协同开发的实验环境。近年来我国也相继实施了多项网格研究计划,如 CNGrid,ChinaGrid 和 NSFCGrid,但是跨组织的协同开发环境基本上停留在科研层面,在实际应用和科研中发挥的作用还比较小。其次,各科研机构的科学家对 SWfMS 的重要性认识还不够,跨学科、跨领域、跨组织的大规模协同开发和交流合作并不是很频繁,目前在我国尚没有出现一个比较成型的 SWF 系统,因此也无法体验到 SWF 技术潜在的优点,因此目前 SWF 技术的研究和应用在我国还不是特别深入和普遍。

5.2 建议

目前我国千万亿次计算机的研制成功代表着我国的高性能计算能力已经进入了世界前列,但是超级计算能力的应用推广与跨领域跨组织的协同开发水平还有待进一步提高。笔者认为在大型基础网格平台上建立一个协同开发环境,开发一个 SWfMS 提供给网格用户使用,将有效降低科学

家使用底层网格计算资源的门槛,提高高性能计算资源的利用率,增加多个科学家之间的协调与交流,并促进科学发展。

6 总结

SWF 作为一种管理大规模复杂科学计算流程的新兴技术,在帮助科学家进行科学发现,促进科学发展等方面起到了非常重要的推动作用。本文对 SWF 的关键技术和相关研究内容进行了比较全面的总结,对各个系统采用的相关技术进行了分析和评价,指出了目前 SWF 技术的不足之处以及 SWF 技术未来的发展方向。最后针对国内 SWF 技术相对比较落后的研究现状进行了分析,并提出了建议,旨在为我国建立一个比较有影响力的 SWfMS 提供必要的借鉴和参考。

参考文献:

- [1] 余德浩. 计算数学与科学工程计算及其在中国的若干发展[J]. 数学进展, 2002,31(1-6).
- [2] Gore A. The Digital Earth: Understanding Our Planet in the 21st Century[J]. Australian Surveyor, 1998,43(2):89-91.
- [3] Deelman E, Gannon G, Shields M, et al. Workflows and e-Science: An Overview of Workflow System Features and Capabilities[J]. Future Generation Computer Systems, 2009, 25(5):528-540.
- [4] Yu J, Buyya R. A Taxonomy of Workflow Management Systems for Grid Computing[J]. Journal of Grid Computing, 2006,3(3-4):171-200.
- [5] Barker A, Hemert J V. Scientific Workflow: A Survey and Research Directions[J]. Parallel Processing and Applied Mathematics, 2008, 4967:746-753.
- [6] Deelman E. Looking Into the Future of Workflow: The Challenge Ahead[C]//Workflow for e-Science: Scientific Workflow for Grids, Springer 2007: 475-481.
- [7] Majithia S, Shields M, Taylor I, et al. Triana: A Graphical Web Service Composition and Execution Toolkit[C]//IEEE International Conference on Web Services (ICWS'04), San Diego, California, IEEE CS, 2004, 514.
- [8] Oinn T, Greenwood M. Taverna: Lessons in Creating a Workflow Environment for the Life Sciences [J]. Concurrency and Computation: Practice & Experience, 2006,18(10):1067-1100.
- [9] Deelman E, Singh G, Su M H, et al. Pegasus: A Framework for Mapping Complex Scientific Workflows onto Distributed Systems[J]. Science Programming, 2005,13:219-237.
- [10] Altintas I, Berkley C, Jaeger E, et al. Kepler: An Extensible System for Design and Execution of Scientific Workflows[C]//Proceedings of the 16th International Conference on Scientific and Statistical Database Management, Santorini Island, 2004:423-424.
- [11] Fahringer T, Prodan R, Duan R, et al. ASKALON: A Development and Grid Computing Environment for Scientific Workflows [C]//Workflows for e-Science, Springer, 2007:450-471.
- [12] Laszewski G, Gawor J, Lane P, et al. Features of the Java Commodity Grid Kit[J]. Concurrency and Computation: Practice & Experience, 2002,15(13-15).
- [13] Kacsuk P, Sipos G. Multi-Grid, Multi-User Workflows in the P-GRADE Grid Portal[J]. Journal of Grid Computing, 2006, 3(3-4): 221-238.
- [14] Guan Z J, Hernandez F, Bangalore P, et al. Grid-Flow: A Grid-Enabled Scientific Workflow System with a Petri Net-Based Interface [J]. Concurrency and Computation: Practice and Experience, 2005, 18(10): 1115-1140.
- [15] Bubak M, Nowakowski P, Unger S. K-WfGrid-Knowledge-Based Workflow System for Grid Applications [R]. Cracow Grid Workshop, 2006.
- [16] Taylor I, Deelman E, Gannon D, et al. Workflow for e-Science: Scientific Workflow for Grids[M]. Springer, 2007.
- [17] Marinescu D. Internet-Based Workflow Management: Toward a Semantic Web[M]. Wiley Interscience, 2002.
- [18] 罗海滨, 范玉顺, 吴澄. 工作流技术综述[J]. 软件学报, 2000,11(7):899-907.
- [19] Erwin D W. UNICORE-A Grid Computing Environment[J]. EuroPar 2001 Parallel Processing, Springer, 2001, 2150:825-834.
- [20] Couvares P, Kosar T, Roy A, et al. Workflow Management in Condor [C]//Workflow for e-Science, London, Springer, 2007: 357-375.
- [21] Goodale T. Expressing Workflow in the Cactus Framework [C]//Workflow for e-Science, London, Springer, 2007: 416-427.
- [22] Andrews T, Carbera F, Dholakia H, et al. Business Process Execution Language for Web Services Version 1.1[S]. 2002.
- [23] Oinn T, Addis M, Ferris J, et al. Delivering Web Service Coordination Capability to Users [C]//Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters, New York: ACM, 2004,438-439.
- [24] Krishnan S, Wagstrom P, Laszewski G V. GSFL: A Workflow Framework for Grid Services [R]. Technical Report Preprint ANL/MCS-P980-0802, Argonne National Laboratory, 2002.
- [25] Amin K, Laszewski G V, Hategan M, et al. GridAnt: A Client-Controllable Grid Workflow System [C]//Proceedings of the 37th Hawaii International Conference on System Sciences(HICSS'04), 2004:10-18.
- [26] Aalst W M P V D, Hofstede A H M t. YAWL: Yet Another Workflow Language[J]. Information Systems, 2005,30(4):245-275.
- [27] Hoheisel A. User Tools and Languages for Graph-based Grid Workflows [J]. Concurrency and Computation: Practice & Experience, 2006,18(10):1101-1113.
- [28] Alt M, Hoheisel A, Pohl H W, et al. A GridWorkflow Language Using High-Level Petri Nets [C]//the Proceeding of The Second Grid Resource Management Workshop (GRMW'2005), Springer, 2006:715-722.
- [29] Qin J, Fahringer T, Pllana S. UML Based Grid Workflow Modeling Under Askalon[J]. Distributed and Parallel Systems, Springer US, 2007:191-200.
- [30] Fahringer T, Pllana S, Villazon A. A-GWL: Abstract Grid Workflow Language [C]//ICCS 2004, Springer, 2004:42-49.
- [31] Lee E A, Neuendorffer S. MoML — A Modeling Markup Language in XML — Version 0.4[S]. Technical Memorandum ERL/UCB M 00/12, 2000.

- [32] Avery P, Foster I. GriPhyN Annual Report for 2003—2004[R]. GridPhyN Technical Report 2004 - 70, August, 2004.
- [33] XBay: A Graphical Workflow Composer for Web Services[R]. Technical Report 004, LEAD, 2006.
- [34] Wang Y, Hu C, Huai J. A New Grid Workflow Description Language [C]//Proceedings of 2005 IEEE International Conference on Services Computing, Beijing: IEEE CS, 2005,2:11 - 15.
- [35] Woodman S, Parastatidis S, Webber J. Protocol-Based Integration Using SSDL and Pi-Calculus[C]//Workflow for e-Science. 2007.
- [36] Huang C, Huang Q. SWFL 2.0 Specification Service Workflow Language [R]. United Kingdom: Cariff University, 2006: 1-95.
- [37] Malinova A, Gocheva-Ilieva S. Using the Business Process Execution Language for Managing Scientific Processes [J]. Information Technologies and Knowledge, 2008, 2(3):257 - 261.
- [38] Callahan S P, Freire J, Santos E, et al. Managing the Evolution of Dataflows with VisTrails [C]//Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW 2006), Washington, DC, IEEE Computer Society, 2006:71.
- [39] XBay: A Graphical Workflow Composer for Web Services [R]. Technical Report 004, LEAD, 2006.
- [40] 杜晓丽, 蒋昌俊, 徐国荣, 等. 一种基于模糊聚类的网格 DAG 任务图调度算法[J]. 软件学报, 2006, 17(11):2277 - 2288.
- [41] 林伟伟, 齐德昱, 李拥军, 等. 树型网格计算环境下的独立任务调度[J]. 软件学报, 2006, 17(11):2352 - 2361.
- [42] Yong W, He C M, Zong X D. QoS-Aware Grid Workflow Schedule[J]. Journal of Software, 2006, 17(11):2341 - 2351.
- [43] Wiczorek M, Prodan R, Fahringer T. Comparison of Workflow Scheduling Strategies on the Grid [C]//Proceeding of Parallel Processing and Applied Mathematics (PPAM'05), Springer, 2006:792 - 800.
- [44] Yu J, Buyya R, Ramamohanarao K. Workflow Scheduling Algorithms for Grid Computing [J]. Metaheuristics for Scheduling in Distributed Computing Environments, LNCS 146, 2008:173 - 214.
- [45] Simmhan Y L, Plale B, Gannon D. Karma2: Provenance Management for Data Driven Workflows[J]. International Journal of Web Services Research, 2007, 5(2):1 - 22.
- [46] Plankensteiner K, Prodan R, Fahringer T. Fault-tolerant Behavior in State-of-the-art Grid Workflow Management Systems [R]. Technical Report, School of Electronics and Computer Science, University of Westminster. 2007, <http://www.coregrid.net/mambo/images/stories/TechnicalReports/tr-0091.pdf>
- [47] Laszewski G V, Foster I, Gawor J. Java CoG Kit Workflow Concepts for Scientific Experiments [R]. Argonne National Laboratory, Argonne, USA, 2005.
- [48] Roue D D, Goble C. Six Principles of Software Design to Empower Scientists[C]//WWW Conferences Archive, United Kingdom, 2007, <http://eprints.ecs.soton.ac.uk/15032/1/myExpSoftware.pdf>
- [49] Turuncoglu U, Murphy S, Technical Summary and Progress Report for a Kepler-based Modeling Workflow System [C]//2009. http://www.earthsystemcurator.org/publications/report_0906_workflow.pdf
- [50] McPhillips T, Bowers S, Zinn D, et al. Scientific Workflow Design for Mere Mortals[J]. Future Generation Computer Systems, 2009, 25(541 - 551).
- [51] Shawn B, Timothy M P, Sean R, et al. Kepler/pPOD: Scientific Workflow and Provenance Support for Assembling the Tree of Life[J]. Provenance and Annotation of Data and Processes, Springer, 2008, 5272:70 - 77.
- [52] Keyvn C T, Jannie C. Information Retrieval for Language Tutoring: An Overview of the REAP Project [C]//Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2004:544 - 545.
- [53] Liu M, He H, Sun X, et al. Scientific Workflow Approach[C]//2009 Second International Conference on Intelligent Computation Technology and Automation. 2009.
- [54] Barga R S, Jackson J, Araujo N, et al. Trident: Scientific Workflow Workbench for Oceanography [C]//Proceedings of the 2008 IEEE Congress on Services, 2008:465 - 466.
- [55] Lin C, Lu S, Fei X, et al. A Reference Architecture for Scientific Workflow Management Systems and the VIEW SOA Solution[J]. IEEE Transaction on Service Computing, 2009, 2(1):79 - 92.
- [56] Tan W, Missier P, Madduri R, et al. Building Scientific Workflow with Taverna and BPEL: A Comparative Study in caGrid [C]//Proceedings of the 4th International Workshop on Engineering Service-oriented Applications. IEEE, 2009: 118 - 129.
- [57] Mandal N, Deelman E, Mehta G, et al. Integrating Existing Scientific Workflow Systems: The Kepler/Pegasus Example[C]//Proceedings of the 2nd Workshop on Workflows in Support of Large-scale Science, Monterey, 2007:21 - 28.
- [58] Lin W, Peltier S T, Grethe J S, et al. Case Studied on the Use of Workflow Technologies for Scientific Analysis: The Biomedical Informatics Research Network and the Telescience project [C]//Workflow for e-Science: Scientific Workflow for Grids, 2007: 109 - 125.
- [59] Abel W L, Khim U. The Telescience Project: Application to Middleware Interaction Components[C]//Proceedings of the 18th IEEE Symposium on Computer-Based Medical Systems, 2005: 543 - 548.
- [60] Lin A W, Peltier S T, Grethe J S, et al. Case Studied on the Use of Workflow Technologies for Scientific Analysis: The Biomedical Informatics Research Network and the Telescience Project [C]//Workflow for e-Science, London, Springer, 2007:109 - 125.
- [61] Zeng J, Du Z, Hu C, et al. CROWN FlowEngine: A GPEL-Based Grid Workflow Engine [C]//Proceedings of High Performance Computing and Communications (HPCC'07), 2007: 4782:249 - 259.
- [62] Liu C, Zhang W, Luo Z, et al. Managing Large-Scale Scientific Computing in Ensemble Prediction Using BPEL [C]//IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA'09). Chengdu, 2009:94 - 101.
- [63] 王黎维, 黄泽谦, 罗敏, 等. 集成对象代理数据库的 SWF 服务框架中的数据跟踪[J]. 计算机学报, 2008, 31(5):721 - 732.