一种基于云计算架构的业务流程优化研究

赵 曦
(上海金融学院 信息管理学院,上海 201209)

摘 要: 本文提出了一种将业务流程进行优化分解成为可以进行独立并行处理任务的方法,可以在云计算环境下分组处理具有共同特征的计算和操作任务,实现优化资源调配.通过 Hadoop MapReduce 并行计算架构进行模拟验证,实验结果表明了该方法在业务处理效率、资源使用和灵活性方面的优势,在大量在线和批量业务流程处理领域具有一定的应用和深入研究价值.

关键词: 业务流程优化;MapReduce 计算框架;云计算
中图分类号: TP396 **文献标识码:** A **文章编号:** 1000-7180(2013)03-0160-05

Research of Business Process Optimization Based on Cloud Computing

ZHAO Xi
(School of Information Management, Shanghai Finance University, Shanghai 201209, China)

Abstract: A BPO method is presented, which consists in creating a dynamic model and dividing process into parallel and independent tasks to whom are allocated adequate computing resources, tasks with similar operation features are grouped and could be processed in cloud computing architecture. A Hadoop MapReduce cloud computing environment is set up to simulate the performance of the proposed model and experimental results show some inspiring advantages which deserve further research and application work.

Key words: Business Process Optimization; hadoop MapReduce; cloud computing

1 引言

以信息应用和计算资源服务虚拟化为核心的云计算架构和技术得到了广泛的认同,正在成为 IT 产业和信息化应用系统架构的发展趋势,实践表明,基于云计算架构的系统在资源利用、服务效率、运行成本及能源消耗方面具有明显的优势.由 Google 提出的 MapReduce 并行处理模型有效简化了在计算资源集群上的并行数据处理架构[1], Hadoop[2] 云计算平台在存储资源虚拟化 HDFS 的基础上实现了该并行计算架构和接口,较好地满足了并行数据处理的需求,越来越多地在实际得到应用.

针对行业业务管理和信息处理集中化后对流程处理效率、系统扩展性和数据存储可靠性的更高要求,云计算架构和应用的优势凸显.本文提出一种应对大量业务流程处理的实时优化方法,将业务流程的任务节点进行分解和分类,形成流程队列和若干可以进行并行处理分类任务队列,使用 Hadoop MapReduce 并行计算框架进行并行处理,MapReduce 提供的资源调度和容错机制能够有效提高业务处理系统的可扩展性和稳定性.模拟实验表明,流程任务分解优化方法(TBM)比通常的以流程为处理单元的方式(PBM)具有一定的优势,可以在云计算环境下分组处理具有共同特征的计算和操作任务,实现优化资源调配[3],提高大量在线和批量流程事务处理的效率,为进一步的深入研究和应用奠定了基础.

2 动态业务流程分析

业务流程是一组将输入转化为输出的相互关联

收稿日期: 2012-06-19; **修回日期:** 2012-07-25

或相互作用的活动,活动之间不仅有严格的先后顺序限定,而且活动的内容、方式、责任等也都必须有明确的安排和界定,以使不同活动在不同角色之间进行交接成为可能,活动与活动之间在时间和空间上的转移可以有较大的跨度. 业务流程集合和规范了企业一系列创造价值的活动,包括研发、生产、营销、服务和内部管理,信息管理系统能够有效地固化和处理流程执行信息,有效提高了流程执行的刚性和可追溯性,以工作流(Workflow)为基础的业务系统是流程化管理的典型应用,大部分业务流程都可以通过某种形式的工作流来描述和定义[4],在本文所研究问题的范围内,业务流程(business process)和工作流具有相同的含义.

在以业务信息处理为核心的金融、证券、电子商务领域 ,需要同时处理处理较大规模的业务流程实例及相关信息,需要配置足够的计算和存储资源来支撑业务处理,需要优化业务流程、提高资源使用效率和根据业务量的变化灵活配置资源.

图 1 为批量流程处理的原理. 有 n 个业务流程(实例)F 等待处理,每个实例对应一个已经定义的流程,每个流程由一系列具有顺序关系的任务节点组成,任务节点进行相应的信息计算和数据存储. 根据计算特征和对资源需求情况,将任务分成 K 个类型. 在一个固定时刻(T1,T2),每个流程对应一个或多个等待处理的活动节点,其它节点处于完成或等待状态.

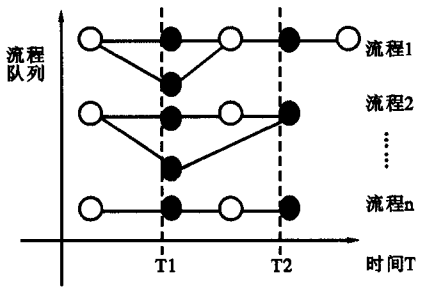

图 1　在线和批量处理流程原理

流程类型 $WT=\{WT_1,WT_2,\cdots,WT_i,WT_k\}$,K 为流程总数.

流程实例 $WF=\{WF_1,WF_2,\cdots,WF_j,WF_n\}$,n 为实时流程实例数.

每个流程包括若干个任务节点 jb、节点之间的关系和节点环境描述,在时刻 T 形成一个由活动节点构成的任务队列 JB.

任务类型 $JT=\{JT_1,JT_2,\cdots,JT_m\}$, M 节点任务类型数.

任务队列 $JB_t=\sum JB_{ij}$,i 为流程编号,j 为任务编号,t 为时刻.

任务环境 JE ={关联,会话,约束,状态,计算}.

根据上述描述,获得了一个实时流程队列与任务队列,并根据任务的类型构建若干个子队列,设想一下在某些领域和企业,当每天有数十万甚至百万或更多的流程实例需要处理,就会形成大量实时或批量处理流程事务和流程任务请求,随着业务处理和数据集中的进一步发展,将会对计算和存储资源提出更高的要求.

3　动态流程任务优化原理

根据业务流程和事务处理的原理和现有信息处理架构,多数系统在处理过程中以流程(实例)为处理单元,根据业务类型分配到指定的处理模块或系统,可以通过负载均衡机制来调度计算资源,实现一定程度上的并行处理[1],图 2 为处理架构(PBM). 其中 Fi 为系统需要处理的所有类型流程队列,Ftj 为按照流程的业务类型 T 进行分组的流程子队列集合,Mtx 为类型 T 的流程处理资源(模块、系统、服务)的组合,当某个类型流程队列长度变化时,可以调整 Mtx 来满足流程处理的性能指标,Fio 为流程处理完毕的事务输出信息.

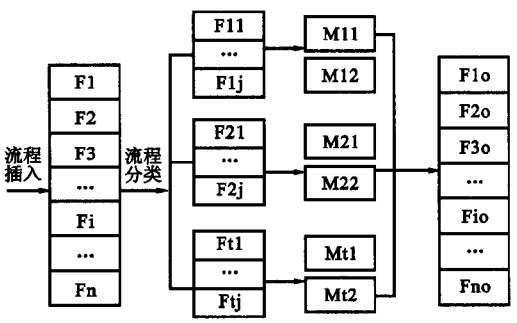

图 2　以流程为单位的处理流程原理(PBM)

目前大部分应用信息系统采用以上述流程为单位的处理模式,比较适应当前的以服务器为计算资源单位(cpu,ram,disk,os)和流程处理程序运行载体的架构,通过业务流程分类来配置服务器或集群,将流程处理提交给对应的服务器群.

上述处理架构在资源使用效率方面存在不足,一方面因为组成流程的节点在资源需求和处理时间跨度上会存在较大差异,难以提高资源使用效率,另一方面,所分配资源需要具备能够处理所有节点工作的能力,提高了对资源的要求.

为了提高计算资源的使用效率及大批量业务流程处理的性能，需要对流程进行分解和优化，即按照流程任务节点(Job)来进行处理队列调度、业务计算和资源分配. 根据(1)中描述的业务流程任务分解模型，获得了一个实时活动任务队列和若干个按照任务类型的子队列，队列中任务是独立的处理，并可以进行并行计算. 流程优化原理模型如图 3 所示. 其中 Jobij 为任务类型为 i 的流程任务 j.

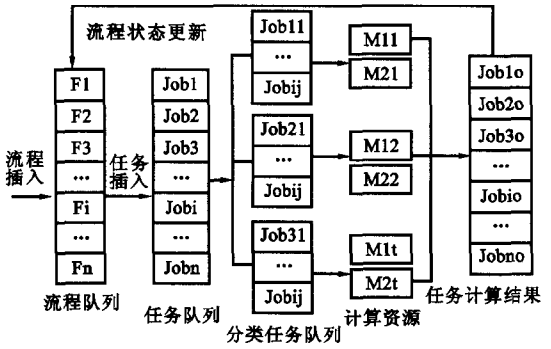

图 3 流程任务分解优化原理(TBM)

上述流程分解优化模型(TBM)的核心为将流程任务进行分解和分组，以任务为计算和资源分配单元，这样可以根据任务类型分配最合适的资源，有效发挥资源的特点优势，提高资源使用效率和成本，如为需要进行复杂计算的任务配置高性能 CPU 处理器和内存量大的计算设备，而为需要存储信息和网络通讯的任务配备 I/O 性能强和硬盘量大的资源. 由于流程任务的计算和资源需求可以精确估算，可以根据任务队列的规律配置计算资源以满足业务处理的性能需求和提高资源使用效率.

优化分解后的任务与其它任务的关联被暂时解除，可以单独进行处理，这一特征为任务并行计算提供了可能，以此为基础的计算和信息处理系统将可以充分利用网格技术、并行计算器和云计算平台的优势.

4 基于 MapReduce 架构的流程优化计算

Hadoop 云计算平台的核心由 HDFS 分布存储和映射机制和 MapReduce 并行计算架构组成，具有开放性、稳定性和扩展性方面的优势，成为了云计算研究和应用的重要平台之一[5]，其架构与提出的业务流程优化模型(TBM)吻合程度高，我们通过该平台来对上述提出的业务流程优化模型进行验证.

MapReduce 通过两个函数 Map 和 Reduce 提供并行计算框架[6]，将计算任务(Job)分解为可以进行独立和并行计算操作集合(Tasks)，提交给 Map 函数处理，而 Reduce 函数收集、整理、排序 Map 函数的计算结果. 其基本功能是按一定的映射规则将输入的 (k1,v1)键值对转换成另一个或一批 list(k2,v2)对输出，而 Reduce 将一个或多个 Map 输出的 list(k2,v2)转换为新的键值对 list(k3,v3)，作为任务计算的输出[7]. 如图 4 所示.

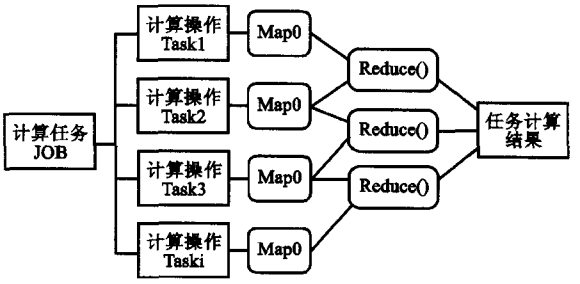

图 4 MapReduce 并行计算框架原理

HDFS 提供了分布计算的存储和映射机制，其中 NameNode 作为计算任务调度和分布文件的管理节点，DataNode 是存储和计算的基本单元，HDFS 通过 NameNode 完成整个计算资源集群的扩展、复制和容错. 如图 5 所示.

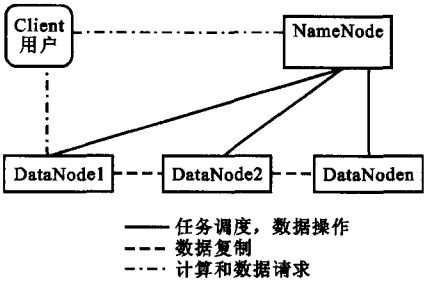

图 5 HDFS 存储和调度架构原理

Hadoop MapReduce 提供基于 JAVA 的并行计算框架类，同时提供 pipe 机制供其他编程语言的函数实现[8]，本文采用 GNU C++实现流程节点处理的 MAP 和 REDUCE 函数.

流程任务输入到 Map 函数及输出的键值为流程和任务编号<Job,Task>，Map 函数的 C++实现执行流程任务需要的 I/O 和数据读写操作，Reduce 函数进行状态管理和整合 Map 函数同类型输出键值表<编号，list(task)>，任务状态的改变动态更新 Job 流程队列及对应的任务队列，从而引发新的激活任务和 MapReduce 处理[9].

流程优化模拟实验基于 5 个业务流程类型，每个业务流程包含 3-6 任务节点，4 个任务类型，如表 1 和表 2，以及图 6 示意了描述流程的节点任务组成

和节点顺序关系. 本次模拟验证以节点任务为单位的资源分配模型,节点顺序关系中的分支和聚合只考虑"与"逻辑关系,即一个节点执行完毕后,所有后续节点马上成为激活状态进入队列等待处理;只有当所有前置节点执行完成后才能处于激活状态,暂不考虑流程的优先级别,根据其进入队列的时间顺序进行处理.

表 1　流程及组成节点

流程/任务	T0	T1	T2	T3	T4	T5	T6	T7
F1	●	●		●	●			●
F2		●	●	●				
F3	●		●		●	●	●	●
F4						●	●	●
F5	●		●		●		●	

表 2　流程节点关系

流程/后任务	T0	T1	T2	T3	T4	T5	T6	T7
F1	T1	T3 T4		T7	T7			
F2		T2	T3	T4				
F3	T2		T3 T4	T5	T5	T6	T7	
F4						T7	T7	
F5	T2		T2		T6			

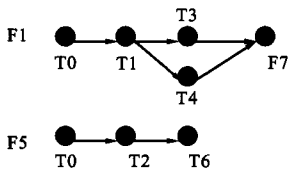

图 6　流程节点逻辑顺序示意例

节点任务执行可以包括如下操作:

(1)状态改变,包括'完成'、'等待'、'失败'、'退回'和'停止';

(2)节点操作包括 I/O,添加处理日志记录和处理结果表;

(3)网络操作,包括 TCP 和 HTTP 协议发送和接受数据.

一种基于 C++的脚本语言和解释器封装了实现上述功能的基本功能函数,用来描述任务的执行操作,如节点任务 T4 的执行脚本如下:

```
[;exp=
@log(1,task_id+" starting");
//update database for task status
@sqlupd("WFDB","r_task","task_id=?","task_status",taskid);
//send data on net for WF queue updates
@net("TCP://namenode:7003","&id="+task_id+"&st=end");
// idle for ressource occupation
@Sleep(1500);
//write execution log
@log(1,task_id+" succ");
]
```

流程实例队列(F queue)平均长度为 400,当一个流程的所有节点任务都执行完毕后从队列中移除. 当小于 300 时启动队列插入,达到 500 时停止,4 任务队列(task queues)平均长度为 100.

使用两个数组来实现表 1 和 2 描述的流程和节点关系,主程序分为流程和任务队列处理两个线程(threads),第一个线程维护流程队列长度和状态,向任务队列输送激活流程节点,另一个线程根据任务队列调用 Map 并行计算,并根据结果维护任务队列,如果完毕则在队列中移除该任务. 节点执行 Map 的键值对输入和输出都为<Wf_id, Task_id>,分别为流程标识和节点任务编号,Map 函数根据节点编号调用对应的任务执行脚本,Reduce 函数将根据任务编号汇集流程,格式为<Task_id, [Wf1, Wf2…]>,发送到流程队列处理线程进行状态维护.

硬件计算架构使用 4 台 RedHat Linux 服务器提供 Hadoop MapReduce 并行计算平台,1 个 NameNode 和 3 个 DataNode,网络环境为 100M 局域网,通过 SSH 实现服务器之间的连接、控制和 HDFS 数据复制,流程和任务队列管理程序运行在 NameNode 上,MAP 和 REDUCE 实现分布在 DateNode 上. 图 7 示意流程优化模拟架构.

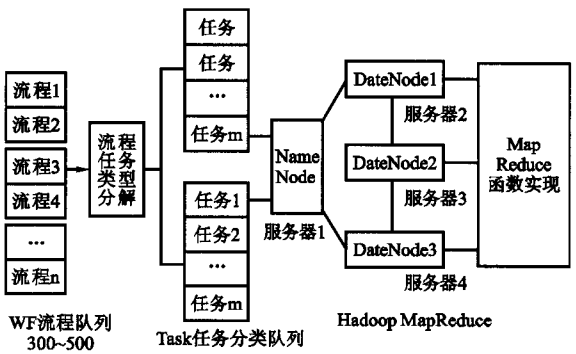

图 7　Haddop MapReduce 流程优化模拟系统架构

平台模拟了以业务流程为处理单元和进行任务分解优化两种架构,使用同样的模拟流程和数据,同时检测了当一个 DateNote 服务器宕机时的任务容

错机制和性能. 表 3 和表 4 分别示意了模拟运行的比较数据.

表 3 流程任务优化分解模型输入数据

方法	流程队列平均数量	并行计算方式	流程最短处理时间	流程任务分类数量	任务队列平均长度	模拟运行时间(分)
流程事务	400	流程	2.5 秒	—	—	120
流程分解	400	任务	2.5 秒	4	100	120

表 4 流程任务优化分解模拟运行结果指标

方法	流程平均处理时间	资源使用差异(%)	队列处理占用时间	容错处理切换时间	MapReduce 调度时间	处理流程数量
流程事务	7.8 秒	54%	11.4 秒	14.1 秒	15.7 秒	1473
流程分解	4.7 秒	21%	20.2 秒	2.7 秒	19.4 秒	2031

流程最短处理时间指流程任务按照关键逻辑路径执行需要的累计时间,当一个任务处理完成后才能激活后续节点任务,以流程为单位的队列处理在 3 个 DataNode 平均分配流程,每个 DataNode 处理的流程数量几乎均等,资源的使用率也相同,任务分解方法形成了 4 个任务队列,DateNode1 和 2 分别处理队列 1 和 2,DateNode3 处理队列 3 和 4.

流程平均处理时间指进入流程队列到所有任务处理完毕的时间,包括等待和任务处理时间,资源使用差异指 CPU 和 IO 在高度使用和过度空闲的比例关系.

根据实验平台记录的运行数据,在使用的业务流程和硬件架构条件下,使用业务流程分解优化的并行计算能够处理更多的流程(2031 对比 1473),资源的使用效率得到提高(54%对比 21%),即使考虑到流程分解和多个任务队列管理的额外开销,运行结果还是表明了所提出方法的优势,为进一步深入研究和完善提供了基础,同时也表明了 Hadoop MapReduce 并行计算架构在处理非搜索类应用和以及搭建云计算平台的可行性.

5 结束语

本文提出了一种信息化处理业务流程的分解优化方法,以适应和使用云计算架构带来的挑战和机遇. 业务流程分解优化方法通过对流程队列的任务分组,可以为实时处理大批量流程的应用领域(银行、证券、保险、电子商务)提高处理效率,MapReduce 原理为流程节点处理提供了并行计算框架,其调度和容错机制可以实现系统计算资源的高扩展性和稳定性.

模拟实验表明,本文介绍的基于并行计算的流程分解优化方法(TBM)比以整个流程事务为处理单元的方法(PBM)在效率、架构和灵活性具有一定的优势. 下一个阶段,拟将该方法进一步完善,在具有典型大批量在线业务流程实时处理的应用领域进行深化和拓展.

参考文献:

[1] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008, 51(1): 107-113.

[2] Jimmy Lin. Data-Intensive Text Processing With MapReduce[M]. Morgan and Claypool Publishers, 2010.

[3] 李建锋. 云计算环境下基于改进遗传算法的任务调度算法[J]. 计算机应用 2011(1):184-186.

[4] Chen Zhang. CloudWF: A computational workflow system for clouds based on hadoop[J]. Lecture Notes in Computer Science, 2009(5931):393-404.

[5] 李成华. MapReduce:新型的分布式并行计算编程模型[J]. 计算机工程与科学 2011,33(3):129-135.

[6] 杨志豪. 一种适应数据与计算密集型任务的私有云系统实现研究[J]. 计算机应用研究, 2011,28(2):621-624.

[7] 汪德帅. 支持多租约 SaaS 应用按需服务的负载均衡策略[J]. 东北大学学报:自然科学版,2011,32(11):352-355.

[8] 易小华. 面向 MapReduce 的数据处理流程开发方法[J]. 计算机科学与探索,2011, 5(2):161-174.

[9] Welcome to hadoop mapreduce[EB/OL]. [2012-04-20]. http://hadoop.apache.org/mapreduce/.

作者简介:

赵 曦 男,(1971—) 博士,副教授. 研究方向为互联网应用、数据结构、云计算技术、软件开发平台.