学校编码：10384
学号：X2007221018
分类号__密级__
UDC___

厦门大学

工 程 硕 士 学 位 论 文

题　目：云环境下本地大规模数据处理的体系结构研究

A new Architecture for In-situ Data Processing in Large-scale Cloud Environment

林之玮

指导教师姓名：李翠华教授
专　业　名　称：计算机技术
论文提交日期：
论文答辩时间：
学位授予日期：

答辩委员会主席：_____
评阅人：_____

年　　月　　日

厦门大学学位论文著作权使用声明

兹呈交的学位论文，是本人在导师指导下独立完成的研究成果。本人在论文写作中参考的其他个人或集体的研究成果，均在文中以明确方式标明。本人依法享有和承担由此论文而产生的权利和责任。

声明人（签名）：

年　　月　　日

厦门大学学位论文著作权使用声明

本人同意厦门大学根据《中华人民共和国学位条例暂行实施办法》等规定保留和使用此学位论文，并向主管部门或其指定机构送交学位论文（包括纸质版和电子版），允许学位论文进入厦门大学图书馆及其数据库被查阅、借阅。本人同意厦门大学将学位论文加入全国博士、硕士学位论文共建单位数据库进行检索，将学位论文的标题和摘要汇编出版，采用影印、缩印或者其它方式合理复制学位论文。

本学位论文属于：

（　　）1.经厦门大学保密委员会审查核定的保密学位论文，于　　年　　月　　日解密，解密后适用上述授权。

（　　）2.不保密，适用上述授权。

（请在以上相应括号内打“√”或填上相应内容。保密学位论文应是已经厦门大学保密委员会审定过的学位论文，未经厦门大学保密委员会审定的学位论文均为公开学位论文。此声明栏不填写的，默认为公开学位论文，均适用上述授权。）

声明人（签名）：

年　　月　　日

摘要

本文为解决云服务供应商所面临的基本数据管理挑战，对大规模云环境下计算基础设施所产生的半结构化日志数据的分析处理工作做了进一步深入研究。分析处理工作是云服务供应商业务的一个重要方面，通过挖掘用户的行为模式和确保有效地利用资源都能让云服务供应商处于竞争优势地位。然而，云计算环境中产生的数据量正在以比摩尔定律更快的速度迅速增加，海量的数据使分析处理工作显得更加困难。为解决大数据的分析处理工作，目前的大部分方法是将数据转移到中心位置，然后再分析，这种方式将产生显著的额外成本和时延。随着云计算环境规模的增加，仅数据迁移的时间和成本问题，都会使这些方法不可行。

本文提出了持续的 MapReduce（Continuous MapReduce, CMR），一种本地处理大型数据的体系结构。为了更好地处理成千上万数据中心服务器的大量日志，CMR 被设计成可扩展的，响应迅速的和高度有效的高性能体系结构。CMR 扩展了 MapReduce 编程模型，在分布式流处理概念的基础上，允许在这些大数据流中持续查询。CMR 突出的结构特征分别包括本地处理的方法，带滑动块窗口的增量处理，以及放宽的一致性模型。

本文使用 Mortar 操作建立了一个原型 CMR 的体系结构，一个分布式流处理器，并通过与当前批处理系统进行对比，对其进行评估。实验结果表明，对于批查询或持续查询，这种分布式流处理器方法可以减少 30%的结果延迟。此外，CMR 的一致性模型，使其在节点部分失效情况下能够快速返回处理结果，并且仍然能保持较高的结果精确性。通过以上实验结果表明，持续 MapReduce 有望使云提供商解决下一代数据管理的挑战，是一个很有前途的大规模分布式数据处理方法。

关键字：大规模云环境，持续的 MapReduce 体系结构，本地处理，增量处理，放宽的一致性模型

Abstract

In order to resolve the fundamental data management challenges faced by the cloud service provider, this paper does a further research about analytical processing of semi-structured log data produced by calculation infrastructure for large-scale cloud environment. The analytical processing work is an important aspect of the cloud service provider business, which can allow cloud service providers to be at a competitive advantage by mining the user behavior patterns and ensuring efficient use of resources. However, the amount of data generated in a cloud computing environment is rapidly increasing in faster than Moore's Law speed, which makes the analytical processing work even more difficult. In order to solve the analytical processing work of large data, most of the current methods are to transfer the data to a central location and analyses them, which generate significantly additional cost and delay. With the increase in the scale of cloud computing environments, only the time and cost of data migration will make those methods unfeasible.

In this paper, a continuous MapReduce (CMR) architecture is proposed, which is a local architecture to deal with large-scale data. In order to better deal with tens of thousands of data center server log, CMR is designed to be scalable, responsive, highly effective and high-performance architecture. CMR extends MapReduce programming model, allowing continuous query in these large data stream on the basis of the concept of distributed stream processing. CMR prominent structural features include the methods of in-situ data processing, incremental processing of a sliding block window and a relax consistency model respectively.

This paper uses the Mortar operator to set up a prototypical architecture of CMR, which is a distributed stream processors and evaluated by comparison with the current batch system. The experimental results show that this distributed stream processors can reduce 30% delay for the batch queries or the continuous queries. In addition, the CMR consistency model let it can quickly return processing results and still be able to maintain the higher accuracy of the results while partial nodes failure.

The experimental results indicate that the continuing MapReduce is promising in addressing the challenges of the next generation of data management with large-scale distributed data processing method to cloud providers.

Key words: large-scale cloud environment, continuous MapReduce architecture, in-situ processing, incremental processing, relaxed consistency model.

目 录

Content

绪 论

1.1 论文研究背景

本文主要探讨分布式大型数据中心之间的数据处理体系结构。良好的体系结构是运行大型云服务的一个关键组成部分。“云”是信息技术行业的一种当前模式，它使原来由内部专家定制的 IT 解决方案，转变成由第三方的云服务提供商提供的在互联网上的动态可扩展性服务。这些云服务提供商提供的服务覆盖了整个技术领域，包括可扩展的硬件基础设施，开发环境，部署平台，以及技术成品。无 IT 基础设施的企业可以使用这些云服务快速地开发和部署它们的应用程序。

这种“一切作为一种服务”的模式，创造了一个复杂的生态系统：以互联网为基础的网络服务。单一客户提出超过 50 种不同的服务请求，并且接触成千上万的机器[1]，这是很正常的情况。服务供应商每秒需要处理数十万客户的请求[2]，随着这些服务普及率的增加，所要处理的请求负载也随之增加。

在一个具有成本效益的环境下处理这些负载，云服务供应商使用一般性的廉价硬件来构建一个规模庞大的基础设施。一个单一的云服务提供商组织可以在世界各地部署几十上百个数据中心，并拥有数千台服务器的。微软最近将数据中心部署在爱尔兰的都柏林和美国的芝加哥，每一个数据中心拥有 30 万台服务器[3]。亚马逊、雅虎和 Facebook 在其整个基础设施，估计有超过 50,000 台的服务器[4]。谷歌已经表示，他们目前正在考虑设计一种核心服务，它的规模在 100 万至 1000 万台服务器之间[1]。

为了确保有效地利用基础设施，并获得竞争优势，可扩展的日志处理是一个重要的环节。服务供应商持续多方面多角度地监控他们使用半结构化日志数据的系统。通过分析用户点击流日志的行为模式，提供有针对性的广告，可以增加相应的利润收益。电子商务和信用卡公司通过分析销售点的交易日志，以检测欺诈。基础设施供应商使用日志数据，检测到硬件的错误配置，来提高在大型数据中心之间的负载均衡，并收集关于基础设施使用模式的统计信息[5-7]。

这种半结构化的日志数据在整个分布式基础设施中以极快的速度积累，成了云环境中一个巨大的数据管理挑战。Facebook 每天产生超过 25 TB 的日志数据，用来分析用户的行为和产生有针对性的广告[6]。在一个大型云服务提供商的个人通信方面，每台服务器的日志增长率可超过 10 MB /秒。当前爆炸数据的收集以比摩尔定律更快的速度增加，这表明了数据管理和分析在未来中将变得更加困难[8]。

为了能够从大量数据中发现知识并对此加以利用，使人们最终做出决策，

就必须对大量数据进行深入分析处理，这些复杂的分析处理依赖于复杂的分析模型构造，很难直接进行表达，这称为深度分析(deep analysis)。人们通过数据分析不仅需要知道现在发生什么，更想利用数据预测将要发生什么，这样就可以在事情发生前做一些提前准备[9]。特别对实时性要求严格的应用，高效的数据处理速率和精确的分析结果是处理这些事物的基本要求，特别是对数据处理速率的要求上。传统处理大数据的方法通常是使用采样分类技术，通过采集样本，可以规模化地把数据变小，然后利用现有的技术方法进行数据分析和管理。然而在某些关键领域，采样将使一些重要信息发生丢失，比如在 DNA 分析中，缩小数据规模将使数据分析结果往往与事实背道而驰，然后精准的分析结果需要建立在庞大的数据处理和分析基础上。这意味着需要分析的数据量将急剧膨胀和增长，所以，高效精准的数据处理和分析技术是当今面临的一大难题，也是人们亟待解决的问题之一。

1.2 研究现状

谷歌、微软、雅虎和 Facebook 等公司使用批量处理体系结构，如 MapReduce[10,11]和 Dryad[12]，来执行其分布式的日志数据的特定分析。这些体系结构抽象掉了编写分布式应用程序的复杂性，如并行化、容错、数据分布、负载均衡。这样的抽象化使企业能够利用廉价的硬件，利用成千上万的商品机进行大量的数据任务处理。它们是面向本地区域网络（local-area network, LAN）集群，面向批处理的工作负载和优化吞吐量。

目前，占主导地位的日志处理体系结构，在传统的存储先查询后（store-first-query-later,SFQL）的模型[13]下使用这些批量处理体系结构。许多公司从源节点迁移日志数据到只能附加的分布式文件系统，如谷歌文件系统(Google File System, GFS) [14]和 Hadoop 分布式文件系统(Hadoop Distribute File System, HDFS) [10]。这些分布式文件系统为了日志数据的可用性和容错性而复制它们。一旦这些数据被放置在文件系统中，用户可以在这些分布式文件系统中使用批量处理体系结构执行查询并检索结果。图 1.1 说明了这种工作模式。

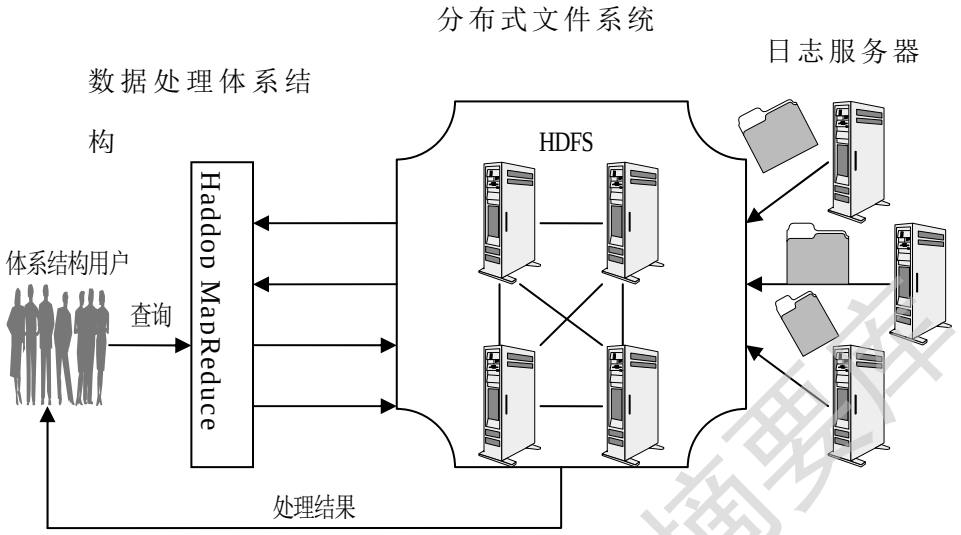

图 1.1 先存储后查询模式下的日志处理

目前的这些方法在分布式环境中表现出一些主要限制。迁移到分布式文件系统的日志数据将有一定的时间限额，即使是时间限制放宽的面向批处理分析。一个简单回复包络分析揭示了这个问题。假定 1 万台服务器（Facebook 目前拥有 30000 台）产生日志数据的速度为 10 MB /秒。此群集每天产生 8PB 的日志数据。在我们的测量中，服务器级的机器可以处理的数据速率为 30 MB /秒。这将需要 3314 个专用的 HDFS[10]节点去处理每一天的日志数据量。这些机器是完全 I/O 限制的，并且无法同时进行超大数据量处理。因此，事实上他们的 CPU 几乎是被闲置的。服务器是在数据中心里不断贬值的最大固定资产之一，其绝大部分的利用价值是在数据迁移上。

这基本限制在日志查询处理上，对数据的准确度和可用性产生广泛影响。许多公司通过缩减相应的数据分析工作，或者使用更多的资源以确保处理速率能跟上的数据的产生率。这两种选择都是昂贵的。在第一种情况下，缩减的数据分析工作将牺牲领先的竞争优势，直接导致收入的损失；在第二种情况下，提高对资源的需求，将使成本提高，特别当加入的资源是不能使用时，这种情况尤为明显。如果我们扩大我们的计算量，假定 10 万台计算机每天产生 80 千兆字节日志数据，它将需要 33140 台专用服务器，才能使处理速度跟上的数据的生产速度。因此，使用此方法缩放相应资源所花费的成本会变得过高。

复杂事件处理（Complex Event Processing, CEP）[15-18]在针对业务流程管理（Business Process Management, BPM）和运筹学（Operational Research, OR）领域[19,20]的数据处理分析问题，已被证明是一个可行的解决方案，它能够在连续的事件流中提取有意义的可操作信息：典型应用是 CEP 引擎[21,22]，它能够处理的数据速率达到每秒 1000 至 10 万条。CEP 是一个为构建简明精确的数据视图，

专门为应用程序提供一个灵活的可伸缩机制。虽然它有高效的处理数据处理速率，但它依赖于许多相关的复杂技术，其中涉及到事件模式检测，事件抽象和事件分层等，操作起来繁琐困难。而且，大多数 CEP 引擎需要手动定义处理和检测模式，其特定的功能取决于所采用的 CEP 引擎，这进一步加大了操作的困难性。

Obweger 等人，在文献[23]中基于 SARI 数据处理框架，提出处理方案模板并解决了数据处理和分析问题，但其处理效率仍相对较低。Jeffery 等人[24]利用 CEP 技术，对基础设施应用的数据进行处理。在他们提出的可扩展性流数据处理框架中，是根据数据的时间和空间特性对其进行处理。主要缺陷在于加工组件被定义为 CQL[25]的查询，这对于没有深入了解流数据处理语言知识的人是很难撰写和部署的，并且难以重用和优化。

另外，传送日志数据表现出耗费精力和低下的资源利用率。最近的一项研究[26]表明，这些批量处理系统更多地选择去处理一个由重复数据驱动查询组成的工作任务。大多数查询都是每天重复一次，有的是每周重复一次，然后每个月重复一次的基本上占少数。查询是更新驱动的，运行后不久，新的日志数据将变得可用，并且只能存取最新的信息。查询所使用的高选择性滤波器，在一些情况下可使总输入数据减少至 17%，并得到最后的输出。同样，Facebook 也表示，他们的查询能过滤掉累计日志数据的 80%无用数据[6]。图书馆提供的许多相同查询过滤和子程序，也表明彼此的相似性。

最后，对数据迁移的障碍应用所造成的延迟进行分析，它需要及时可靠的结果。前面提到的所有应用程序都可以受益于较小的结果延迟。服务器日志，用户点击流和销售点交易是高度动态的内容。在线分析这些数据将创建一个更灵活的系统，使企业能够对重要事件更迅速地作出反应，包括检测欺诈的信用卡活动，检测一组服务器是否正常运作，识别安全漏洞，检测服务内容的流行趋势。

1.3 主要研究内容

这些存储先查询后的基本模型的局限性，突出展示了需要用不同的方法来处理大规模云环境下批量任务负载。我们提出了持续的 MapReduce（Continuous MapReduce, CMR）体系结构，它是一种数据处理体系结构，用来处理及时性要求较高的分布式环境中大规模云环境下批量化的工作负载。

CMR 的体系结构有几个属性，是专门用来解决目前所采用方法所受到的限制。它在大型网络中具有可扩展性，使企业能够更好地应对下一代的数据管理的挑战。它能够及时响应的，并且更灵活的分析有价值的日志数据。它是高度