多核操作系统发展综述

梁荣晓
（江南计算技术研究所　江苏无锡　214083）

【摘　要】多核处理器的核心迅速增长以及结构日益复杂，给未来操作系统的设计带来了很大的挑战。为适应多核处理器的发展，可以利用分布式设计思想，从结构和功能上对传统多核操作系统进行分布式处理优化，将多核硬件划分为不同的子系统，尽可能降低各子系统之间的耦合度，从而提高多核操作系统的可扩展性。本文概括当前多核操作系统研究的三种技术路线，力求宏观展现多核操作系统的发展趋势。

【关键词】多核 操作系统 功能分布 数据分布

Summary of Multicore Operating System Development

Liang Rong-xiao
(Jiangnan Institute of Computing Technology　JiangsuWuxi　214083)

【Abstract】The number of processor cores are growing rapidly and the structure of multi-core processor are being more and more complex, which has brought great challenges to the design of future operating systems. To adapt to the development of multi-core processors, we can take advantage of the distributed design ideas to optimize traditional multi-core operating system. Using the ideas, multi-core hardware is divided into different subsystems, and the degree of coupling is reduced as low as possible, so that the scalability of the multi-core operating system will be improved. In this paper, we summarize three technical design methods of multi-core operating system currently, striving to show the macro trend of multi-core operating system.

【Keywords】multi-core; operating system; functional distribution; data distribution

1　引言

多核处理器的出现大大提升了系统并行处理能力，使越来越多不同类型的应用可以同时在多核平台上进行高效的并行计算。现有成熟的操作系统经过长期的发展，对目前普通多核处理器大多能够提供较好的支持。但同时，多核处理器的核数迅速增长、结构日益复杂，也为未来多核操作系统的设计与优化带来了巨大的挑战。如何适应未来多核处理器的迅速发展，设计高可用、高并行、高可扩展的多核操作系统，是目前业界共同的奋斗目标。

2　现状与挑战

传统多核操作系统采用宏内核[(Macro Kernel，或称为大内核(Monolithic Kernel)]架构，其中以 Linux 与 Windows 操作系统为主要代表。宏内核相当于一个巨大的并发协同的进程组，主要使用单一数据结构，内核本身提供大多数系统服务。在多核处理器核数有限、结构并不复杂的情况下，传统宏内核操作系统基本能够充分利用多核处理器的并行处理能力，对外体现为一个紧耦合、高效的单一操作系统。

随着技术的进步，多核处理器在硬件性能和结构上达到了长足的发展。多核处理器的核心数持续增加，目前已有集成超过 100 个核心的芯片。同时，多核处理器的结构也越来越多样化，出现了异构多核与类 NUMA 多核。

多核处理器的核心迅速增长、结构日益多样化，为传统多核操作系统的设计带来了巨大的挑战。尽管操作系统已经针对类 SMP、类 NUMA 处理器结构对部分内核数据结构进行分布化，但它们本身与特定的同步模式以及数据布局紧密相关，其可扩展性受限于锁竞争、数据局部

性以及对共享内存的依赖等。传统多核操作系统难以适应多核处理器的发展趋势,具体表现在两个方面。

首先,传统多核操作系统难以适应多核处理器核数的飞速增长。传统操作系统往往通过锁来保护共享数据,随着 CPU 核数的增加,进入内核的线程也会随之增加,对锁的竞争将更为激烈,影响系统的整体性能。

另外,核数增加时,传统多核操作系统一般通过创建更细粒度的锁来增加内核的并发性,而调整锁粒度是一项异常复杂的工作。未来处理器核心数量指数增长的情况下,重新设计子操作系统的速度难以与之同步。

其次,类 NUMA 多核处理器以及异构多核处理器的出现给传统多核操作系统设计带来了新的困难。类 NUMA 微结构多核处理器的特点是,多个核在访问片上数据比如 L2Cache 的时延是不同的,各个核部分共享 L2Cache 或者私有 L2Cache。访问时延的不一致性使操作系统的设计更复杂,而且,当核数扩展时,为保证数据一致性所占用的操作系统开销将大大增加。异构多核处理器由一个或者多个主核以及其它从核组成,不同类型的核心给操作系统设计以及系统编程开发带来了很大的困难,其可扩展性也难以实现。

3 技术路线

为适应多核处理器的发展,可以利用分布式设计思想,从结构和功能上对传统多核操作系统进行分布式处理优化,将多核硬件划分为不同的子系统,尽可能降低各子系统之间的耦合度,从而提高多核操作系统的可扩展性。

目前,面向可扩展多核操作系统的研究主要可分为三种技术路线:1)改进传统宏内核架构,以适应多核体系结构,这是目前最广泛的研究方法;2)基于功能分布思想,将不同的核(或者核组)划分为不同的功能,不同功能之间通过共享内存或消息传递通信,开发功能分布式多核操作系统;3)借鉴分布式系统的数据分布思想以及消息通信机制,创新设计数据分布式多核操作系统。

3.1 改进传统宏内核架构

目前商业上应用最广泛的多核操作系统仍然是 Linux、Windows 等老牌操作系统。为改善系统的可扩展性,linux 等传统操作系统一直没有停止过对多核处理器的优化支持。Linux 针对 NUMA 结构处理器修改了内存分配策略,CPU 会优先选择当前节点的物理内存,不够时才寻找附近节点请求物理内存分配。微软的 Windows7 移除了 dispatcher 锁,改动涉及 50 多个文件、6000 多行代码。但限制可扩展性的根本因素——锁与共享内存等,依然是传统操作系统的主要运作元素,因此,对于多核的优化,他们还有较大的改进提升空间。

Corey 操作系统是 MIT 等组织在 Linux 基础上修改操作系统接口实现的,其设计目标是针对当前主流的 Cache 一致性 SMP 多核处理器。其设计思想是“应用程序控制数据的共享”,即通过应用程序对内核间共享资源的控制,减少多核之间不必要的资源传递与更新,以达到更高效利用多内核的目的。

Corey 在 Linux 中增加了三个新接口:1)地址范围,允许应用程序编程时决定私有地址与共享地址的范围;2)核心,允许应用程序制定特定的核心执行;3)共享对象,允许应用程序决定哪个对象对其它核心可见。Corey 系统相对 Linux 系统性能提升明显,基于某 AMD16 核处理器的实验表明,Corey 的 Map Reduce 性能较 Linux 提高了 25%。但是,Corey 改变了操作系统接口,普通应用程序需要经过修改才能在其上运行,其兼容性存在一定问题

3.2 功能分布式多核操作系统

传统多核操作系统的不同核心使用相同的宏内核,主要基于数据并行扩展多核性能,锁机制成为限制系统可扩展性的主要因素。功能分布式多核操作系统是一类将多核按照功能划分的操作系统,不同核心(Core)所使用的内核(Kernel)可以是宏内核或微内核。该类操作系统开辟了新的多核性能扩展路线,从原有的数据并行到新的功能分布,由于功能分布对数据的耦合度大大低于数据并行,因此可扩展性显著高于传统多核操作系统。

FOS 是 MIT 开发的一种面向多核与云计算的操作系统,其设计宗旨是可扩展性以及自适应性。FOS 的设计原则主要是:1) 空间复用取代时间复用,FOS 是在命名空间中进行调度,调度的资源是分布的多个核;2)操作系统分解成特定的服务,各操作系统服务分布在各服

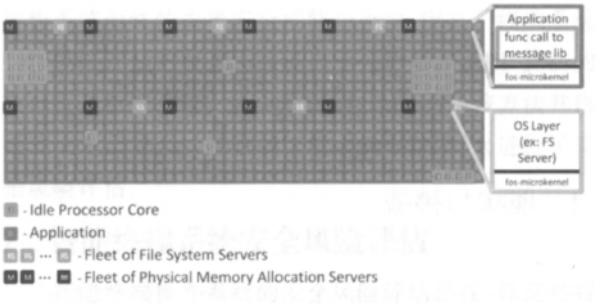

图 1 FOS 系统架构

务器中,各服务器互相协作,彼此通过消息传递进行通信;3)错误自动检测与处理等。图 1 为 FOS 的微内核架构,其中每个处理器核运行一个不同的微内核,分别提供不同的操作系统服务或者运行应用程序。应用程序进程通过高效的消息传递获得操作系统服务,不同的服务或进程间的通信也通过消息机制进行。FOS 对外提供单一系统映像(SSI,Single System Image),适用于传统操作系统的应用程序无需经过特定修改即可直接在 FOS 上运行。FOS 系统具有良好的兼容性以及可扩展性。

3.3 数据分布式多核操作系统

异构以及类 NUMA 多核处理器的与传统多核处理器有明显的区别,即核间耦合度大大降低,主要表现在核间共享内存与 cache 的开销增加以及效率下降。传统紧耦合操作系统抑或 Linux 类 NUMA 操作系统,难以很好的发挥新型处理器的特点。考虑到新型处理器的硬件分布式特点,借鉴分布式系统的数据分布思想,创新设计松散耦合的类分布式多宏内核操作系统,对于提高多核操作系统的可扩展性,无疑是另辟蹊径。

Barrelfish 系统基于 Multikernel 体系结构,是由剑桥微软研究院与瑞士苏黎世联邦理工学院联合开发的新型操作系统,其设计目标是高效管理使用异构的硬件资源,适应多核处理器的发展,如图 2 所示。该系统中每个内核都运行自己的操作系统,很好的支持了内核的异构性。同时它继承了分布式系统的思想,将各内核作为独立的单元,单元通过总线上的消息传递进行通信。这种模型可以带来更好的模块化性能,并使得分布式算法可以直接应用于多内核系统中。

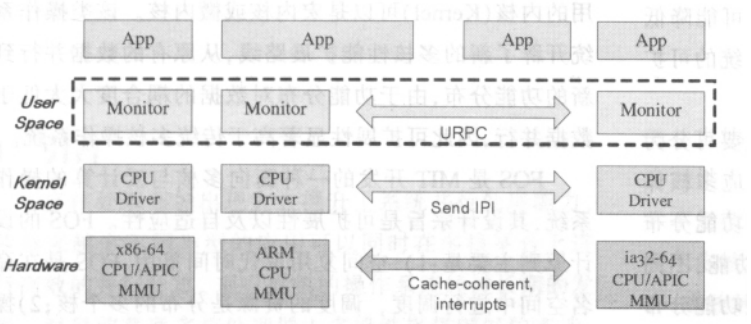

图 2 Barrelfish 架构

4 结束语

多核处理器的核心迅速增长以及结构日益复杂,给未来操作系统的设计带来了很大的挑战。传统多核操作系统的可扩展性受限于锁竞争与 Cache 缺失,因而难以适应多核处理器的发展趋势。

目前,面向可扩展多核操作系统的研究主要可分为三种技术路线,分别是改进传统宏内核架构、开发功能分布式多核操作系统以及开发数据分布式多核操作系统。后两者通过利用分布式设计思想,从结构和功能上对传统多核操作系统进行分布式处理优化,将多核硬件划分为不同的子系统,尽可能降低各子系统之间的耦合度,从而提高多核操作系统的可扩展性。因此,功能分布和数据分布是未来多核操作系统的发展趋势。

参考文献

[1] D. Wentzlaff and A. Agarwal. Factored operating systems (fos): the case for a scalable operating system for multicores. SIGOPS Oper. Syst. Rev., 43(2):76 ⊨85, 2009.

[2] T. C. Rajkumar Buyya, "Single system image (ssi)," the International Journal of High Performance Computing Applications, vol. 15, pp. 124 ⊨135, 2001.

[3] S. Boyd-Wickizer, H. Chen, et al. Corey: Anoperating system for many cores. Proc of 8thUSENIX Symposium on Operating Systems Design and Implementation, pp.43-57, 2008.

作者简介:

梁荣晓(1989-),男,山东日照人,江南计算技术研究所在读研究生;研究方向:操作系统。