

用户空间协议栈在通用多核 服务器上的并行化和性能优化

指导老师 华蓓

SA13011047 别体伟

Outline

- 选题背景
- 研究内容
- 预期成果
- 技术路线
- 可行性分析
- 进度安排

选题背景

- 利用通用多核服务器（配置了万兆网卡）构建高速网络系统已成为主流
- 万兆网卡拥有一些重要的新特性，比如：
 - 多队列支持：网卡可按照包头中的指定域把包发送到固定队列，与多核处理器配合使用很容易实现基于流亲和性的流量负载分配。
- 已有一些利用万兆网卡新特性提供高速raw packet I/O的软件，比如：
 - Intel DPDK、Netmap、IOEngine

基于万兆网卡的协议栈

- 目前基于万兆网卡的网络系统多为流量监视类系统，不具备完整的协议栈（不终结TCP连接）
- Mtcp是第一个基于万兆网卡开发的具有TCP终结能力的协议栈，但它是自行开发的一个演示性的协议栈，很多重要的协议功能（如IP重组，IP地址类型检查，IPsec支持等）没有实现
- 目前尚没有能够充分利用万兆网卡新特性的完整的TCP/IP协议栈实现

论文工作

- 以Libuinet为蓝本，对其进行并行化和性能优化：
 - 使其充分利用万兆网卡的多队列特性和多核处理器的并行计算能力
- Libuinet：
 - 移植自FreeBSD 9.1-RELEASE内核的TCP/IP协议栈
 - 具有完整的协议栈功能
 - 目前的性能不如内核中的原始版本高

研究内容

- 在Intel多核平台上并行化Libuinet，并进行性能优化
- 移植一到两个应用程序到协议栈上，验证协议栈的实用性和性能

预期成果

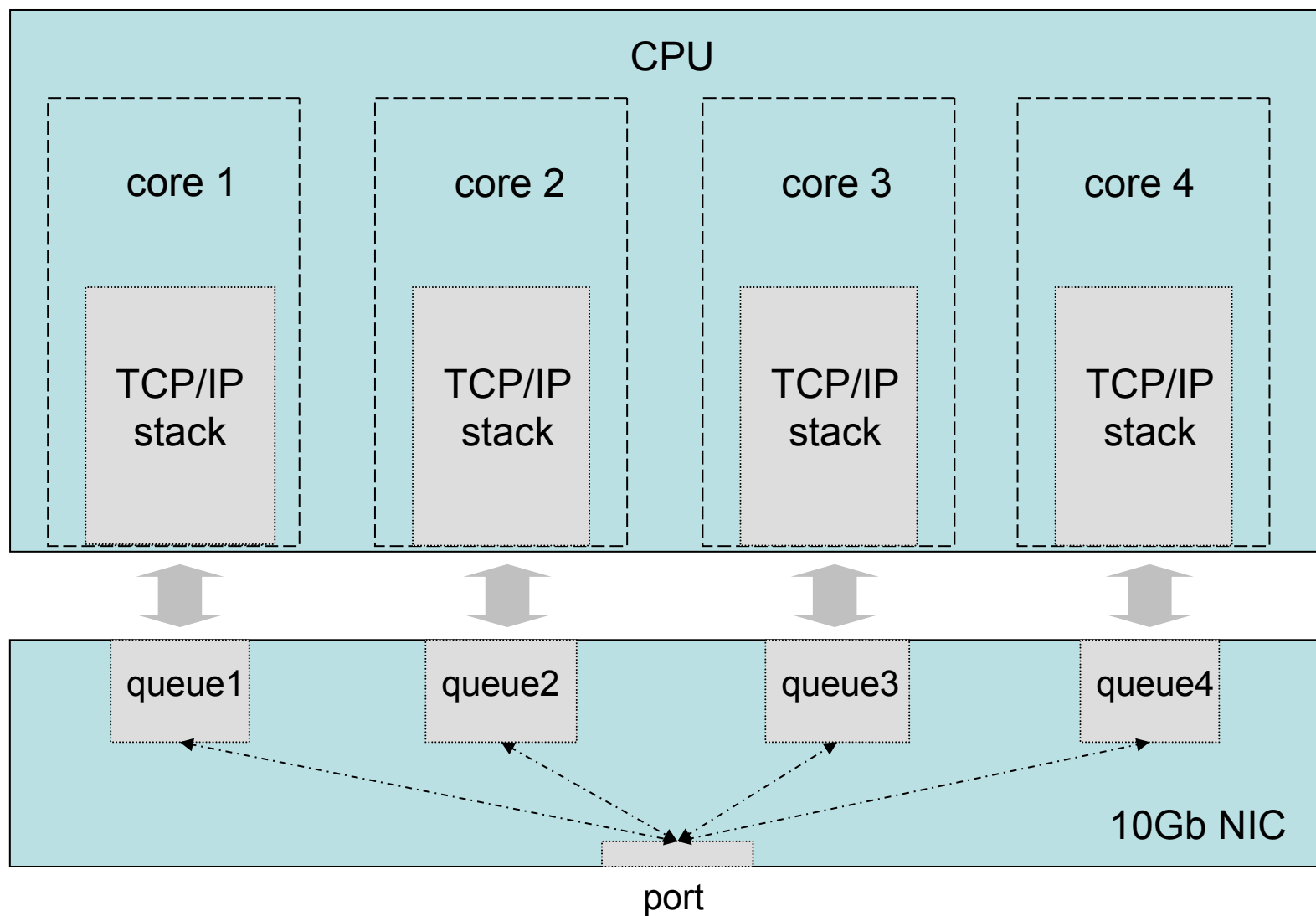
- 以开源软件的形式提供一个可在多核服务器上运行的高性能用户空间协议栈及应用案例，并发表学术论文。

技术路线（1）

协议栈并行化和性能优化：

- 并行编程模型的选择
- 全局协议栈实例转化为per-core的协议栈实例
- 协议栈接收路径和发送路径的加速

并行编程模型



per-core协议栈实例

- 利用网络虚拟化的相关机制，可以创建一个虚拟协议栈实例；
- 可以直接为每个核心建立一个虚拟协议栈实例，从而做到将原本所有CPU核共享的全局协议栈实例转化为每个CPU核独占的per-core的协议栈实例；

协议栈性能优化

- 对协议栈接收路径和发送路径进行加速：
 - 尽量避免从网卡的缓冲区拷贝每个数据包；
 - 加速协议栈的接收例程(ether_input(), ip_input(), tcp_input());
- 拟通过profiling工具寻找具体的系统性能瓶颈，然后进行优化。

技术路线（2）

应用案例开发：

- 将为华为的POF交换机开发的SDN控制器软件移植到协议栈上，采用业界标准的SDN benchmark工具cbench对系统性能进行测试
- 移植一个开源的应用代理服务器，并与基于内核协议栈运行的应用代理服务器进行性能比较

可行性分析

- 到目前为止，已经完成了：
 - 协议栈的并行化
 - 基于新的协议栈编写了简单的应用，并通过了运行测试
- 初步的研究工作表明，以上技术方案是可行的。

创新之处

- 提供首个可在通用多核服务器及万兆网卡上高效运行的完整的开源用户空间协议栈实现。

进度安排

- 2014.09--2015.1 已完成用户空间协议栈的并行化
- 2015.01--2015.02 搭建网络测试环境
- 2015.03--2015.07 进行初步的性能测试工作，并根据结果进行优化
- 2015.08--2015.11 移植SDN控制器，并进行性能测试和优化
- 2015.12--2016.01 移植应用代理服务器，并进行性能测试和优化
- 2016.02--2016.03 分析实验结果，得出结论
- 2016.04--2016.05 撰写毕业论文和准备答辩

Thanks