XOS: An Application Defined Operating System in User-space designed for Datacenter

Chen Zheng (zhengchen@ict.ac.cn), Jianfeng Zhan, and Lixin Zhang
Institute of Computing Technology, Chinese Academy of Sciences

Motivation

Enhance OS performance utilizing virtualization hardware.

Key issue of monolithic kernel (Linux):

1. Applications cannot touch hardware,
2. Resource competitions,
3. Poor scalability.

VT-x: breaks current CPU modes into tow new modes, while allowing applications touch privileged hardware features.
SR-IOV: multiplex a device into several virtual functions, each of which has independent space.

XOS Prototype

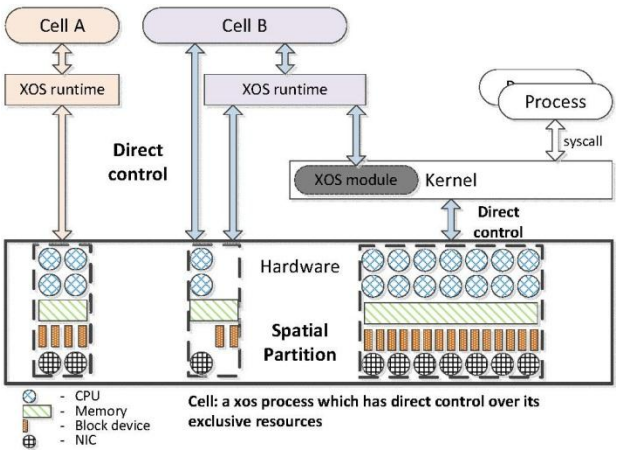
Goals:

1. Bypass OS kernel,
2. Reduce interference,
3. Scales well.

Design Principles:

1. Applications define their own kernel subsystem in user-space,
2. Spatial partition.

User-space application-defined OS model

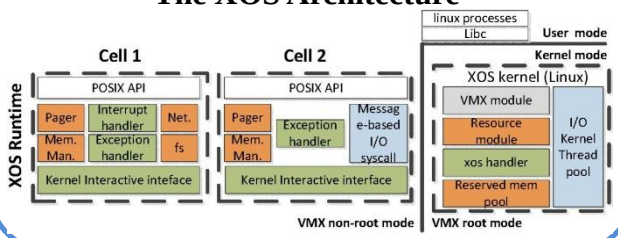


http://prof.ict.ac.cn/chenzheng/

Implementation

- Built on Linux
- Leverage VT-x to control hardware in user-space,
- Application-defined XOS runtime:
  - ✓ User-space memory and device management,
  - ✓ User-space interrupt/exception handler (pagefault handler, etc.),
  - ✓ Message-based I/O system call.

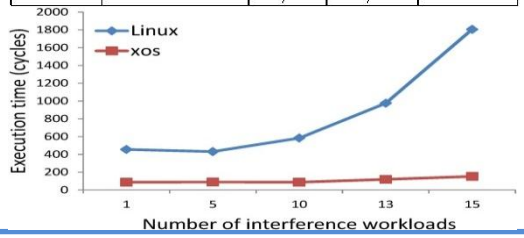
The XOS Architecture



Preliminary Results

OS	null syscall	lgdt	gettsc	malloc
XOS	42	114	45	126
Linux	174	N/A	N/A	312

2.4x



This work is supported by the National High Technology Research and Development Program of China (Grant No. 2015AA015308), Major Program of National Natural Science Foundation of China (Grant No.61432006), and National Natural Science Foundation of China (No. 61303054)