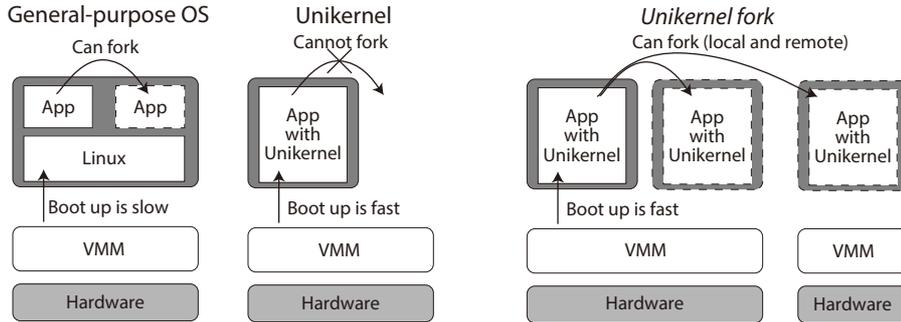


Running Multi-Process Applications on Unikernel-based VMs

Minoru Kanatsu (Student)[†], and Hiroshi Yamada[†]
[†]TUAT

Motivation

- The performance of VM launch is an important factor
 - Affect to service quality in cloud
- Unikernel is a lightweight VM
 - Allows apps. to run on hypervisor directly



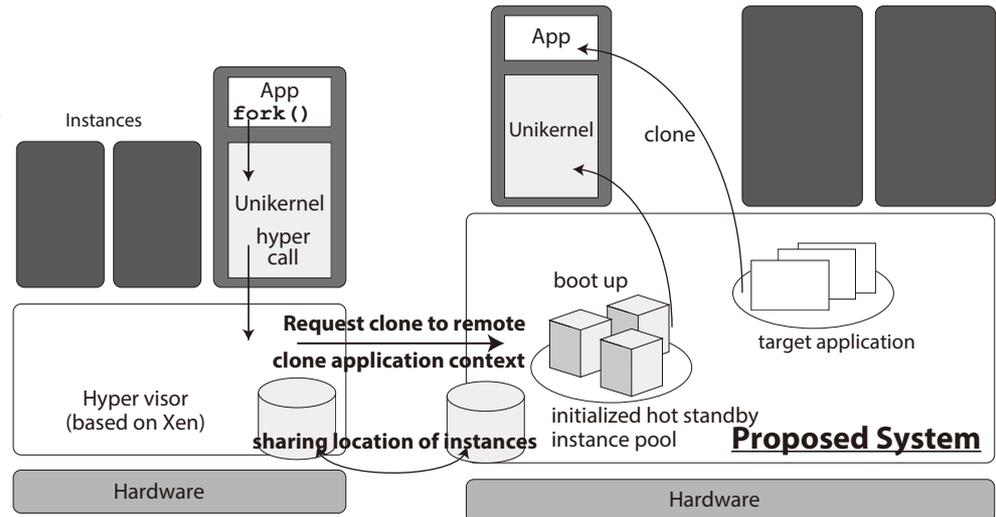
We present **Unikernel fork** which is a new mechanism for Unikernel-based VM on their cloud

Proposal and Design

1. Hot standby instance pool system
2. Real fork system on remote VMM
3. Location sharing system of instances

Target

- Server apps. whose child process are long-lived
- e.g.) Apache
 - Each connection is long-lived on streaming server
- e.g.) PostgreSQL
 - OLAP workloads are heavy

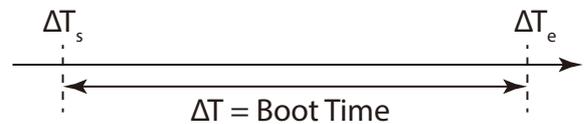


Preliminary Experiment

- Measured boot time of Mini-OS, Rumprun unikernel [Kantee and Cormack, 2014] and Alpine Linux
- 10 instances boot up at same time and measured each of instance

Definition of boot time

- ΔT_s ... start instance with $\times 1$
- ΔT_e ... print TSC by app. code on VM



Configuration

- Host
- Linux 4.9.23 (CentOS 7), Xen 4.7.0
 - Intel Xeon E5-2640v3
 - Memory 256 GB
- Guest
- Mini-OS 4.8.0
 - Rumprun commit ad23d14
 - Alpine Linux 3.4.3

Result

	Rumprun	Mini-OS	Alpine Linux
Boot up time[s]	1.00	5.40	19.5

5x
20x

Next Plans

- Implementation and evaluation real fork & fork system to Xen and Rumprun unikernel
 - To make running real world application such as Apache HTTP server
- Make evaluation system for new system architecture
- Optimize our fork time to get close to Rumprun boot time