Live Migration of Virtual Block Devices

Cristian Zamfir[1], Colin Perkins[2] and Peter Dickman[3]
Department of Computing Science, University of Glasgow, UK
{zamf[1], pd[3]}@dcs.gla.ac.uk, csp@csperkins.org[2]

Abstract

Today's computing environments do not provide support for seamless user mobility. We argue that the support for user mobility between various computing environments can be improved by using live migration of virtual machines and present our initial steps in this direction. We are addressing efficient live migration of persistent storage without leaving residual depencies between migration sites and without sacrificing I/O performance. Our initial implementation shows that migrating virtual block devices is feasable and can occur with unnoticeble downtime. The advantage of using live migration for user mobility that the work environment is never suspended and that network connections are not disturbed and batch jobs can be executed during user's commutes.

1. Introduction

User mobility

Support for user mobility between various computing environments can be improved by employing live virtual machine migration. We are addressing a scenario where a user can migrate his work environment without interruption across various computers.

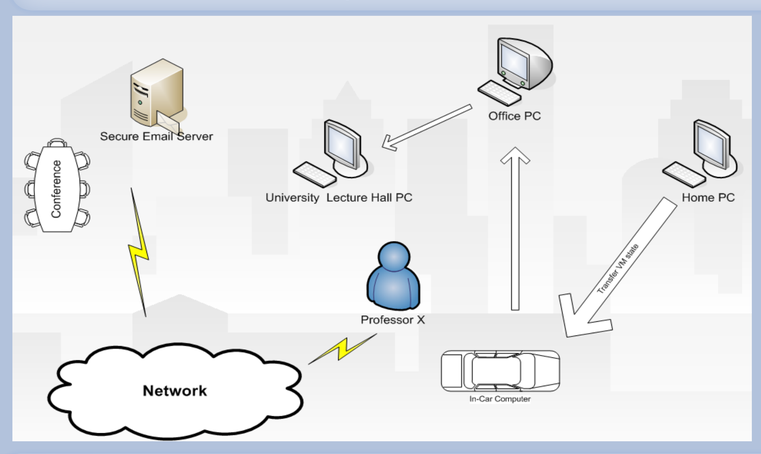

Figure 1: Scenario
Professor X's virtual environment is live migrated between various physical hosts around the world such as his office PC or his in-car PC. His system is not suspended during the commutes so that network activity such as attending a video conference is not affected.

Approach

- Pack the work environment as a virtual machine
- Use *live* migration to maintain network connectivity during commutes
- Allow batch jobs to be executed during user commutes
- Aim at running at near native performance
- Leave no residual dependencies to the originating site

Challenges

- Support for live migration of virtual machines without Network Attached Storage (NAS)
- Support trust relations between the migrating host and the migration destination
- Optimize memory migration for low bandwidth commodity links such as DSL
- Migrating outside the same subnet without relying on IP tunnels

2. Virtual Block Device Migration

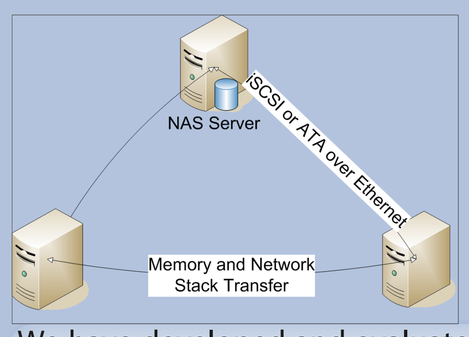

Figure 2: *Currently, Virtual Block Devices are migrated using Network Attached Storage.*

We have developed and evaluated a prototype for Xen live migration without NAS. Virtual block devices are live migrated with the aid of Distributed Replicated Block Devices (DRBD), a popular cluster disk mirroring software.

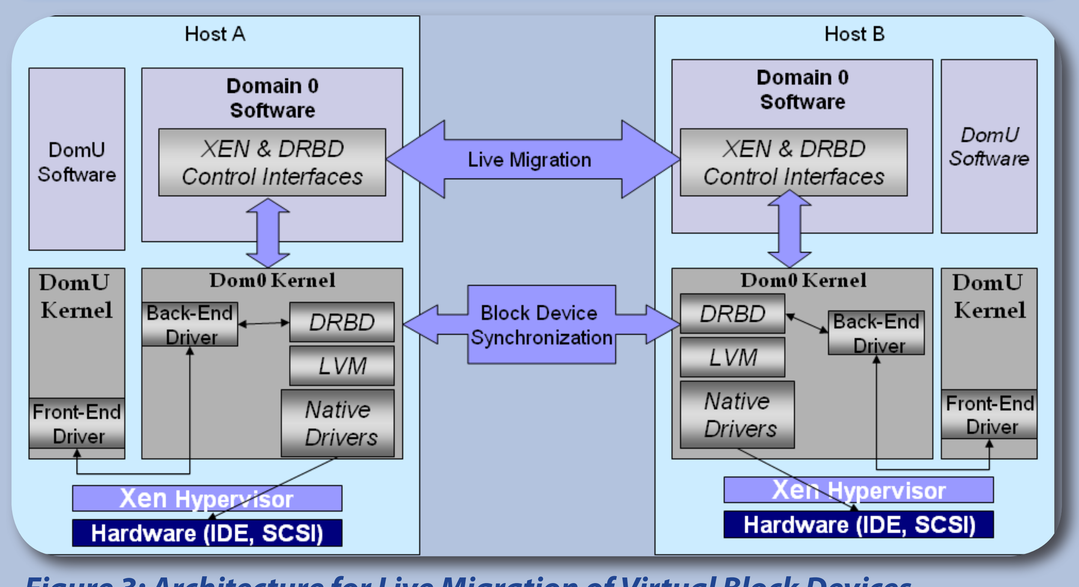

Figure 3: Architecture for Live Migration of Virtual Block Devices
An XML-RPC server plugged in Xen's userspace control interface is managing the block device synchronization and synchronizes with the migration of the memory. During migration, disk reads are done locally but disk writes are done synchronously on both block devices.

3. Results

Network Performance During Live Migration

Live migration without NAS over 100Mbps network links occurs with sub-second downtime.

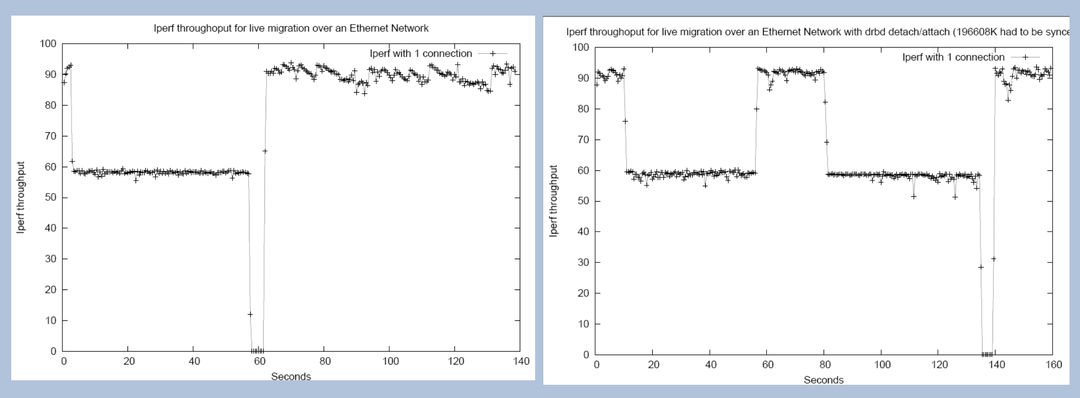

Figure 4: a) *Network throughput of one TCP connection from the migrating virtual machine to an external server. Storage is synchronized during migration* ***b)*** *The effect on network performance of a Split-Brain occurrence in DRBD causing some of the data to be synchronized.*

I/O Performance During Live Migration

Write access performance during migration depends on the network bandwidth available because disk writes are mirrored synchronously.

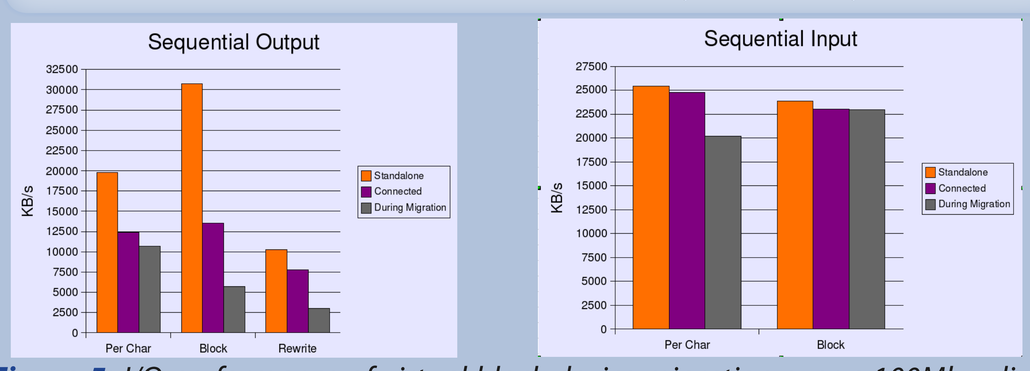

Figure 5: *I/O performance of virtual block device migration over a 100Mbps link- measured with the Bonnie++ benchmark.*

Peer-To-Peer Virtual Block Device Migration

Users typically work from several locations distributed over a large geographical area. Incomplete replicas of the persistent state of their virtual machine environment can be exploited to speed up the synchronization process. We are working on a Peer-To-Peer based approach for the live migration of virtual block devices.

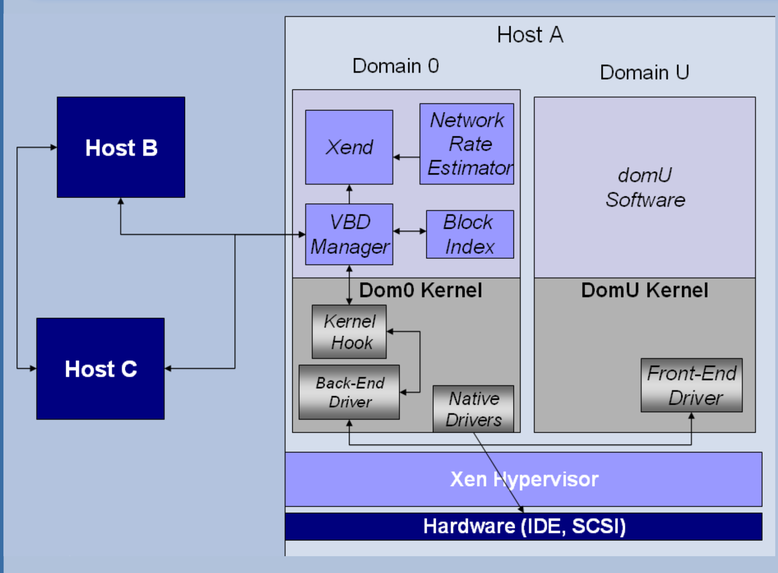

Figure 5: *Architecture of a P2P approach for live migration of virtual block devices. A kernel hook managed from userspace allows distribution of disk blocks to incomplete replicas on remote hosts.*

4. Conclusions

- We implemented a prototype for live migration of virtual block devices without NAS
- The current prototype uses DRBD and supports only two peers.
- Consistency of replication during live migration is done using synchronous disk writes, at the expense of potentially poor performance of disk write throughput in WAN environments.

Future work:

- Add support for multiple peers and asynchronous writes during migration
- Investigate the use of Intel Trusted Technology to establish a mutual trust relation between the hypervisor and the migrating VM

http://www.dcs.gla.ac.uk/~zamf
zamf@dcs.gla.ac.uk