LEIBNIZ INFORMATION CENTRE
FOR SCIENCE AND TECHNOLOGY
UNIVERSITY LIBRARY

TIB

TIB // P.O. Box 60 80 // 30060 Hannover, Germany

Orrick, Herrington & Sutcliffe LLP
Bibliothek, Lars Woehning

Heinrich-Heine-Allee 12
40213 Düsseldorf

Hannover, 26. April 2021

We hereby confirm that reference:

Proceedings of the 2nd Symposium on Networked Systems Design & Implementation (NSDI '05) held May 2 - 4, 2005, Boston, MA. USA, USENIX Association – PRINTED EDITION – ISBN 1-931971-30-7

with its information carrier (journal issue, conference proceedings, book, research report and such like in printed form or saved electronically)

was incorporated at our establishment on February 28, 2006, under the shelf mark RS 4601(2).

From the day of accession, the public, and in particular the professional world, shall in principle be able to access the reference.

pp.

Diana Friedrich

Diana Friedrich
TIB Customer Service

Technische
Informationsbibliothek (TIB)
Welfengarten 1 B
30167 Hannover, Germany
T +49 511 762-8989
F +49 511 762-8998
customerservice@tib.eu
www.tib.eu

VAT Reg No:
DE214931803
Tax.-No. 25/202/26441

TIB
Nord/LB Hannover
IBAN: DE47 2505
0000 0101 4287 53
BIC: NOLADE2HXXX

UB
Nord/LB Hannover
IBAN: DE57 2505
0000 0106 0276 34
BIC: NOLADE2HXXX

Die TIB is a member of

Leibniz
Leibniz Association

The TIB is the German National Library for all areas of engineering as well as architecture, chemistry, information technology, mathematics and physics. The TIB is a foundation of public law.

USENIX Association

Proceedings of the 2nd Symposium on Networked Systems Design & Implementation (NSDI '05)

May 2–4, 2005
Boston, MA, USA

Conference Organizers

Program Chairs

Amin Vahdat, *University of California, San Diego*
David Wetherall, *University of Washington*

Program Committee

Miguel Castro, *Microsoft Research*
Jon Crowcroft, *University of Cambridge*
David Culler, *University of California, Berkeley*
Michael Dahlin, *University of Texas at Austin*
Peter Druschel, *Rice University*
Paul Francis, *Cornell University*
Ramesh Govindan, *University of Southern California*
Joe Hellerstein, *Intel Research and University of California, Berkeley*
Dina Katabi, *Massachusetts Institute of Technology*
Eddie Kohler, *University of California, Los Angeles*
Ed Lazowska, *University of Washington*
Jeffrey Mogul, *HP Labs*
Vern Paxson, *ICSI and Lawrence Berkeley National Laboratory*
Larry Peterson, *Princeton University*
Stefan Savage, *University of California, San Diego*
Srinivasan Seshan, *Carnegie Mellon University*
Ellen Zegura, *Georgia Institute of Technology*

Steering Committee

Thomas Anderson, *University of Washington*
Peter Honeyman, *CITI, University of Michigan*
Mike Jones, *Microsoft Research*
Robert Morris, *Massachusetts Institute of Technology*
Mike Schroeder, *Microsoft Research*
Amin Vahdat, *University of California, San Diego*
Ellie Young, *USENIX Association*

Poster Session Chairs

Eddie Kohler, *University of California, Los Angeles*
Michael Dahlin, *University of Texas at Austin*

The USENIX Association Staff

External Reviewers

Mukesh Agrawal
Aditya Akella
Nazrul Alam
David Andersen
Thomas Anderson
Remzi Arpaci-Dusseau
Mary Baker
Magdalena Balazinska
Hitesh Ballani
Andy Bavier
Brian Bershad
Ashwin Bharambe
Ken Birman
Mike Burrows
John Byers
Yatin Chawathe
Allen Clement
Richard Cox
Constantinos Dovrolis
Richard Dunn
Jeremy Elson
Cristian Estan
Nick Feamster
Marc Fiuczynski
Sally Floyd
Paul Gauthier
Steven Gribble
Timothy Griffin
Saikat Guha
Krishna Gummadi
Andreas Haeberlen
Jeff Hoye
Mark Huang
Jaeyeon Jung
Flavio Junqueira
Frans Kaashoek
Shrikanth Kandula
Scott Karlin
Sachin Katti
Dejan Kostic
David Kotz
Boon Thau Loo
David Lowenthal
Ratul Mahajan
Petros Maniatis
Tudor Marian
David Mazières
Alan Mislove
Michael Mitzenmacher
Steve Muir
Aki Nakao
Suman Nath
David Oppenheimer
Vivek Pai
Jeff Pang
Ruoming Pang
Kurt Partridge
Adrian Perrig
Ansley Post
Kostas Psounis
Venugopalan Ramasubramanian
Charles Reis
Maya Rodrig
Dan Sandler
Stefan Saroiu
Marianne Shaw
Alan Shieh
Atul Singh
Manpreet Singh
Alex Snoeren
Yee Jiun Song
Tammo Spalink
Neil Spring
Ion Stoica
Michael Swift
George Varghese
Vidhyashankar Venkataraman
Arun Venkataramani
Vivek Vishnumurthy
Geoffrey Voelker
Kevin Walsh
Matt Walsh
Andrew Whitaker
Dan Williams
Bernard Wong
Kenneth Yocum
Haifeng Yu
John Zahorjan
Ming Zhang
Xinyang Zhang

Live Migration of Virtual Machines

Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hansen[†],
Eric Jul[†], Christian Limpach, Ian Pratt, Andrew Warfield

University of Cambridge Computer Laboratory
15 JJ Thomson Avenue, Cambridge, UK
`firstname.lastname@cl.cam.ac.uk`

[†] *Department of Computer Science*
University of Copenhagen, Denmark
`{jacobg,eric}@diku.dk`

Abstract

Migrating operating system instances across distinct physical hosts is a useful tool for administrators of data centers and clusters: It allows a clean separation between hardware and software, and facilitates fault management, load balancing, and low-level system maintenance.

By carrying out the majority of migration while OSes continue to run, we achieve impressive performance with minimal service downtimes; we demonstrate the migration of entire OS instances on a commodity cluster, recording service downtimes as low as $60ms$. We show that that our performance is sufficient to make live migration a practical tool even for servers running interactive loads.

In this paper we consider the design options for migrating OSes running services with liveness constraints, focusing on data center and cluster environments. We introduce and analyze the concept of *writable working set*, and present the design, implementation and evaluation of high-performance OS migration built on top of the Xen VMM.

1 Introduction

Operating system virtualization has attracted considerable interest in recent years, particularly from the data center and cluster computing communities. It has previously been shown [1] that paravirtualization allows many OS instances to run concurrently on a single physical machine with high performance, providing better use of physical resources and isolating individual OS instances.

In this paper we explore a further benefit allowed by virtualization: that of live OS migration. Migrating an entire OS and all of its applications as one unit allows us to avoid many of the difficulties faced by process-level migration approaches. In particular the narrow interface between a virtualized OS and the virtual machine monitor (VMM) makes it easy avoid the problem of 'residual dependencies' [2] in which the original host machine must remain available and network-accessible in order to service certain system calls or even memory accesses on behalf of migrated processes. With virtual machine migration, on the other hand, the original host may be decommissioned once migration has completed. This is particularly valuable when migration is occurring in order to allow maintenance of the original host.

Secondly, migrating at the level of an entire virtual machine means that in-memory state can be transferred in a consistent and (as will be shown) efficient fashion. This applies to kernel-internal state (e.g. the TCP control block for a currently active connection) as well as application-level state, even when this is shared between multiple cooperating processes. In practical terms, for example, this means that we can migrate an on-line game server or streaming media server without requiring clients to reconnect: something not possible with approaches which use application-level restart and layer 7 redirection.

Thirdly, live migration of virtual machines allows a separation of concerns between the users and operator of a data center or cluster. Users have 'carte blanche' regarding the software and services they run within their virtual machine, and need not provide the operator with any OS-level access at all (e.g. a root login to quiesce processes or I/O prior to migration). Similarly the operator need not be concerned with the details of what is occurring within the virtual machine; instead they can simply migrate the entire operating system and its attendant processes as a single unit.

Overall, live OS migration is a extremelely powerful tool for cluster administrators, allowing separation of hardware and software considerations, and consolidating clustered hardware into a single coherent management domain. If a physical machine needs to be removed from service an administrator may migrate OS instances including the applications that they are running to alternative machine(s), freeing the original machine for maintenance. Similarly, OS instances may be rearranged across machines in a cluster to relieve load on congested hosts. In these situations the combination of virtualization and migration significantly improves manageability.