

InstantLab – The Cloud as Operating System Teaching Platform

Alexander Schmidt, Andreas Polze
Operating Systems and Middleware Group

Cloud Futures 2011

Operating Systems and Middleware



Prof. Dr. rer. nat. habil. Andreas Polze
Dipl.-Inf. Alexander Schmidt
Hasso-Plattner-Institute for Software Engineering
at University Potsdam
Prof.-Dr.-Helmert-Str. 2-3
14482 Potsdam, Germany



Agenda

1. Operating System Experiments – the Windows Case
2. InstantLab
3. Demo
4. Research Questions
5. Conclusions

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Windows Operating Systems Internal Curriculum Resource Kit (CRK)

http://www.microsoft.com/resources/sharedsource/windowsacademic/curriculumresourcekit.mspx

Click Here to Install Silverlight

United States Change | All Microsoft Sites

Microsoft

Shared Source Initiative

Shared Source Home

Source Code Offerings:

- Product Source Programs
- Reference Source Service
- Community Source

Licensing:

- Product Source
- Reference Source
- Community Source

Source Code For:

- Faculty and Students
- Governments
- Enterprises
- OEMs
- Most Valuable Professionals (MVP)
- Systems Integrators
- Developers

Source Code Venues:

- Overview
- Code Center Premium
- Reference Source Code Center
- CodePlex
- MSDN Download and Code Center

FAQ:

- Frequently Asked Questions

Microsoft Windows Academic Program

Integrate core Windows technologies into your teaching and research.

Home Curriculum Resource Kit (CRK) Windows Research Kernel Project OZ Faculty Experiences

Windows Operating System Internals Curriculum Resource Kit

- Overview
- Details
- Use and Restrictions
- Eligibility Requirements
- Customer Commitment
- Your Questions

Download CRK

The Windows Operating System Internals Curriculum Resource Kit (CRK) materials are available via the [Academic Resource Center](#).

[» Download](#)

Overview

The CRK is a pool of materials and resources that explain operating system (OS) concepts based on the Windows XP and Windows Server 2003 operating system family. The CRK structure follows the IEEE-CS/ACM Operating System Body of Knowledge (BOK) as defined in the Computing Curriculum 2001 project by the joint IEEE-CS and ACM Task Force (CC2001).

The CRK is based on *Windows Internals*, 4th edition (Microsoft Press, 2005) by Mark Russinovich and David Solomon. The experiments, lab descriptions, quizzes, and assignments are an integral part of the course materials and were tested over a five-year period in an OS architecture class taught by Andreas Polze at Humboldt University of Berlin and Hasso Plattner Institute at University of Potsdam, Germany.

[↑ Back to top](#)

Details

15 available units: The CRK consists of 15 units — 10 core and seven elective units from the BOK and three

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Windows Research Kernel (WRK)

- Stripped down Windows Server 2003 sources
 - Only kernel itself, no drivers, GUI, user-mode components
 - Missing components: HAL, power management, plug-and-play
- Released in 2006
- Freely available to academic institutions
- Encouraged by license:
 - Modification
 - Publication (of excerpts)

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Structuring Experiments: The UMK Approach

- U-phase
 - Concentrate on OS concepts
 - Introduce OS interfaces
 - Systems programming
- M-phase
 - Observe concepts at run-time
 - Introduce monitoring tools
 - System measurements
- K-phase
 - Discuss kernel implementation
 - Introduce kernel source code (WRK/UNIX)
 - Kernel programming

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Kernel Programming Experiments

- Debugging/Instrumenting the WRK
 - Boot phase
 - Process creation
 - Single-step debugging the WRK in a virtual machine
- Creating a new system call
 - Hide/Show a specified process from the system
 - Memorize hidden processes
 - Implement a system service DLL
- Memory management

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Kernel Programming Experiments – Bottom Line

- Experiments comprise
 - Documentation
 - Source code
 - Workload generators
 - Measurement/visualization tools
- Experiment setup:
 - Install and configure test operating system
 - Build and deploy the sources
 - Configure kernel debugging infrastructure
- Virtualization helps, but
 - Variety of OS platforms, virtualization vendors among students
 - Hardware requirements

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

1. Operating System Experiments – the Windows Case
2. InstantLab
3. Demo
4. Research Questions
5. Conclusions

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

The InstantLab Idea

- Provision of “canned experiments”
 - Virtual machine images (VMI) as foundation
 - Self-contained, pre-configured experiment in one VMI
 - Instantaneous execution of a lab or experiment on Cloud resources



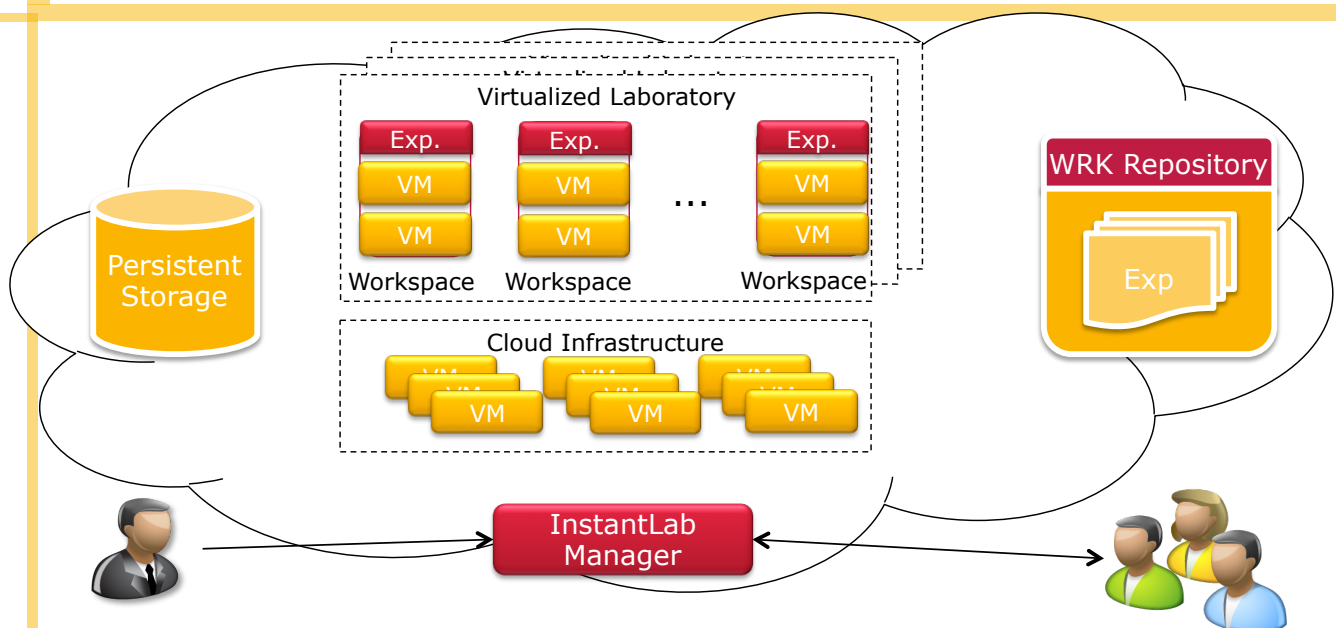
Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Embrace The Cloud

- Virtualize laboratory environment
 - No physical machines in university, no maintenance
 - Compute resources in the Cloud
- Migrate exercises and demos into the Cloud
 - Provision of VM template(s) for each exercise
 - Instantiation on demand
- Facilitate experiments through remote display session
 - Run experiments in Web browser
 - Support of various platforms and compute power

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

InstantLab - Architecture



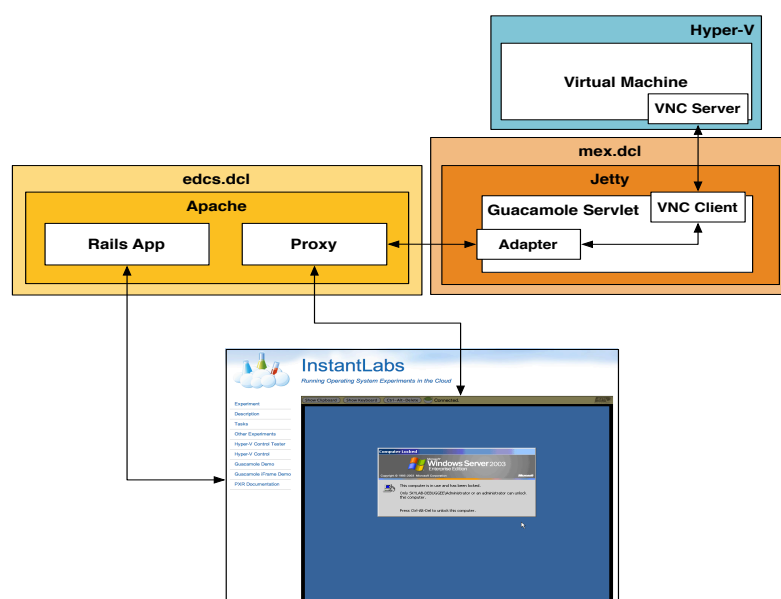
Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Agenda

1. Operating System Experiments – the Windows Case
2. InstantLab
3. Demo
4. Research Questions
5. Conclusions

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Facilitating Remote Access



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

InstantLab Demo – Working Set Replacement Experiment

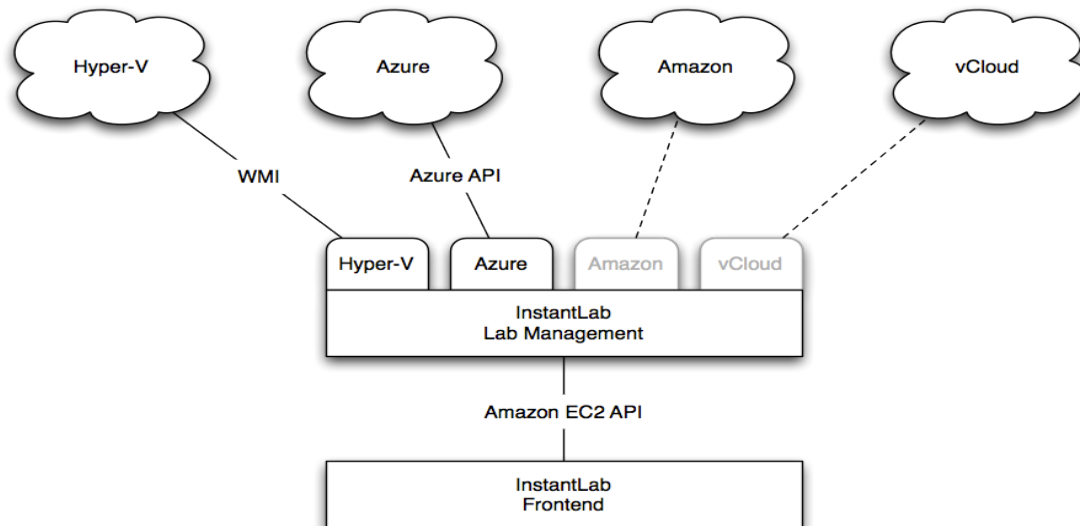
Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

InstantLab Demo – Working Set Replacement Experiment



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Lab Management – Architecture

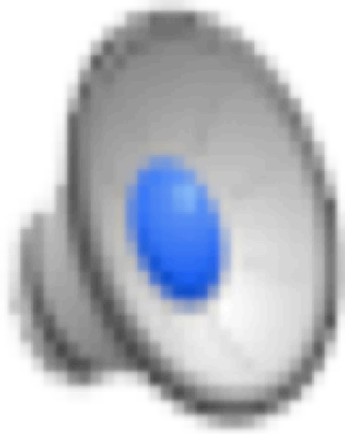


Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

InstantLab Demo – Lab Management

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

InstantLab Demo – Lab Management



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Agenda

1. Operating System Experiments – the Windows Case
2. InstantLab
3. Demo
4. Research Questions – Cloud Reliability
5. Conclusions

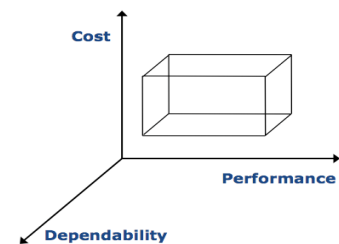
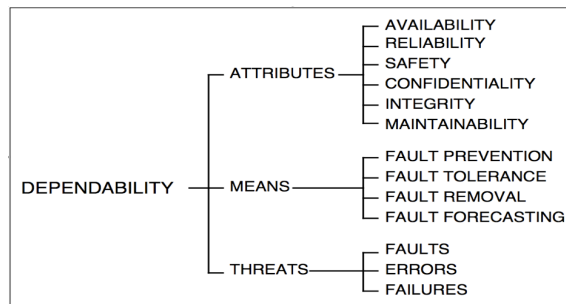
Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Dependability – does it matter for Cloud?

Umbrella term for operational requirements on a system

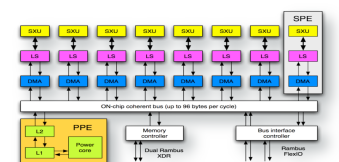
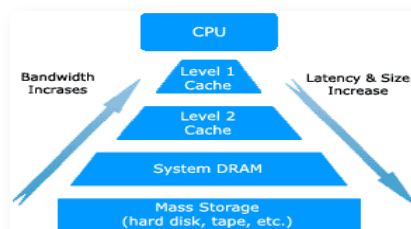
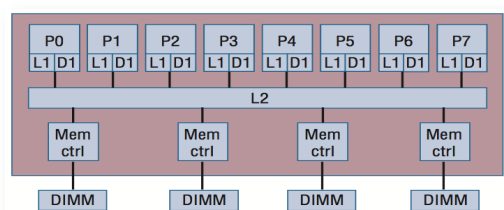
- „Trustworthiness of a computer system such that reliance can be placed on the service it delivers to the user“ [Laprie]

General question: How to deal with unexpected events ?



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

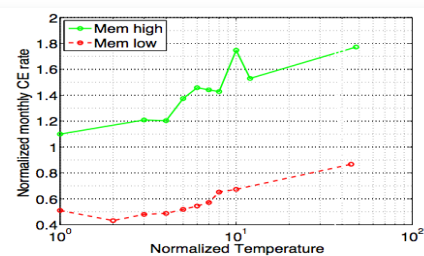
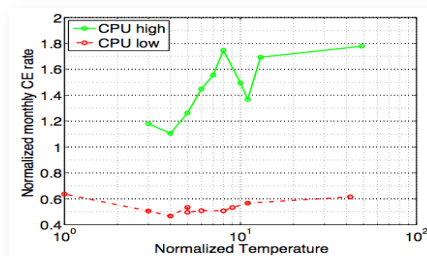
Hardware Revolution in the x86 World



Classical Reliability Wisdoms Get Replaced



- Dramatic shift in single machine reliability aspects
 - SMP becomes heterogeneous tiled on-chip network
 - Decreasing structural sizes + dynamic frequency and voltage
 - Massive memory increase
- **More fault classes, less error containment !**
- Few research results from HPC perspective
 - Type and intensity of workload significantly influences life time
 - Failure rates depend on processor count, not hardware type



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Research in the FutureSOC Lab



HPI FutureSOC Lab

- Collaboration with industry for software research on next-generation x86 hardware (32-65 cores, 1-2 TB RAM)

Our research @ FutureSOC Lab

- Failure prediction based on cross-level monitoring data analysis
- Pro-active virtual machine migration
- Fault injection based on UEFI firmware technology

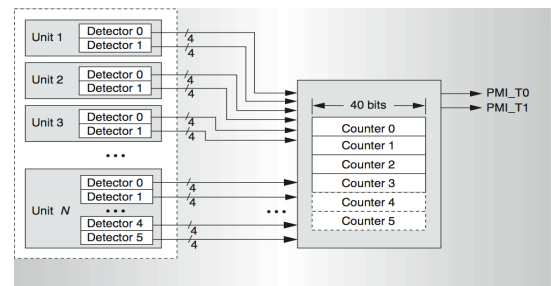
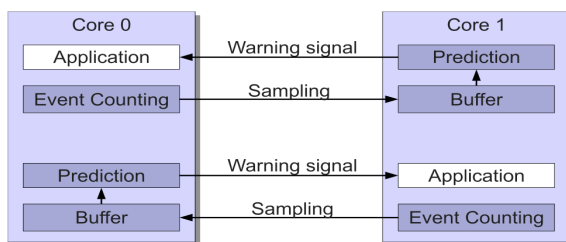


Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

CPU Level: Online Hardware Failure Prediction

Using X86 hardware performance events

- Instruction retirement, cache miss, branch miss-prediction, ...
 - Limited number of hardware counter units -> exploit event correlations
 - Threshold-triggered, time-triggered
- Applicable to major cellular multiprocessing platforms (Intel, AMD, SPARC, IBM Power)



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Memory level: observations from our FutureSOC Lab

Date	Severity	Event	Source	Description
15-Jun-2010 13:47:12	Info	No	BIOS	System boot (POST complete)
15-Jun-2010 13:45:53	Major	No	[0x00:00]	POST - 'MEM4_DIMM-2D' memory training failed
15-Jun-2010 13:45:53	Major	No	[0x00:00]	POST - 'MEM4_DIMM-1D' memory training failed
15-Jun-2010 13:45:53	Major	No	[0x00:00]	POST - 'MEM4_DIMM-2B' memory training failed
15-Jun-2010 13:45:53	Major	No	[0x00:00]	POST - 'MEM4_DIMM-1B' memory training failed
15-Jun-2010 13:45:53	Critical	Yes	SMI	'MEM4_DIMM-1D' Memory: Uncorrectable error (ECC)
15-Jun-2010 13:45:53	Critical	Yes	SMI	'MEM4_DIMM-1C' Memory: Uncorrectable error (ECC)
15-Jun-2010 13:45:53	Critical	Yes	SMI	'MEM4_DIMM-1B' Memory: Uncorrectable error (ECC)
15-Jun-2010 13:45:53	Critical	Yes	SMI	'MEM4_DIMM-1A' Memory: Uncorrectable error (ECC)
15-Jun-2010 13:45:40	Critical	Yes	iRMC S2	'MEM4_DIMM-2D': Memory module failed (disabled)
15-Jun-2010 13:45:40	Critical	Yes	iRMC S2	'MEM4_DIMM-1D': Memory module failed (disabled)
15-Jun-2010 13:45:40	Critical	Yes	iRMC S2	'MEM4_DIMM-2B': Memory module failed (disabled)
15-Jun-2010 13:45:40	Critical	Yes	iRMC S2	'MEM4_DIMM-1B': Memory module failed (disabled)
15-Jun-2010 13:43:43	Info	No	BIOS	System boot (POST complete)
14-Jun-2010 17:41:47	Critical	Yes	iRMC S2	'MEM4_DIMM-1D': Memory module error
14-Jun-2010 17:26:17	Major	Yes	iRMC S2	'MEM4_DIMM-1D': Memory module failure predicted

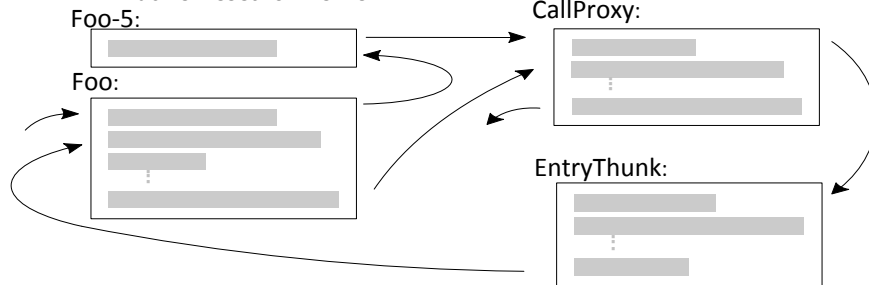
Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

OS level: our NTrace for Windows

- Compiler/linker switch
 - /hotpatch, /functionpadmin
 - Microsoft C compiler shipped with Windows Server 2003 SP1 and later

- Hotpatchable:

- Windows Server 2003 SP1, Vista, Server 2008, Windows 7
- Windows Research Kernel

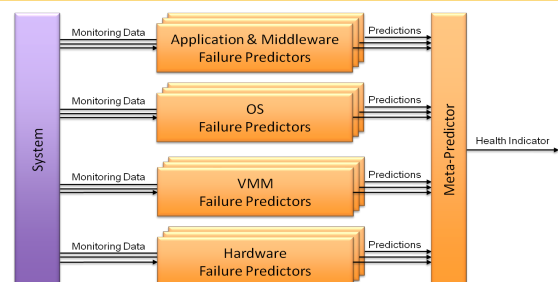


```

...
retn 10
nop
nop
nop
nop
nop
nop
NtfsPinMappedData:
mov     edi, edi
push    ebp
mov     ebp, esp
mov     ecx, [ebp+18h]
mov     edx, [ebp+0Ch]
...
    
```

„Ablaufverfolgung in einem laufenden Computersystem“
Pat. pend. **DE-10 1009 038 177.5**

The Meta Predictor – Bringing it all together

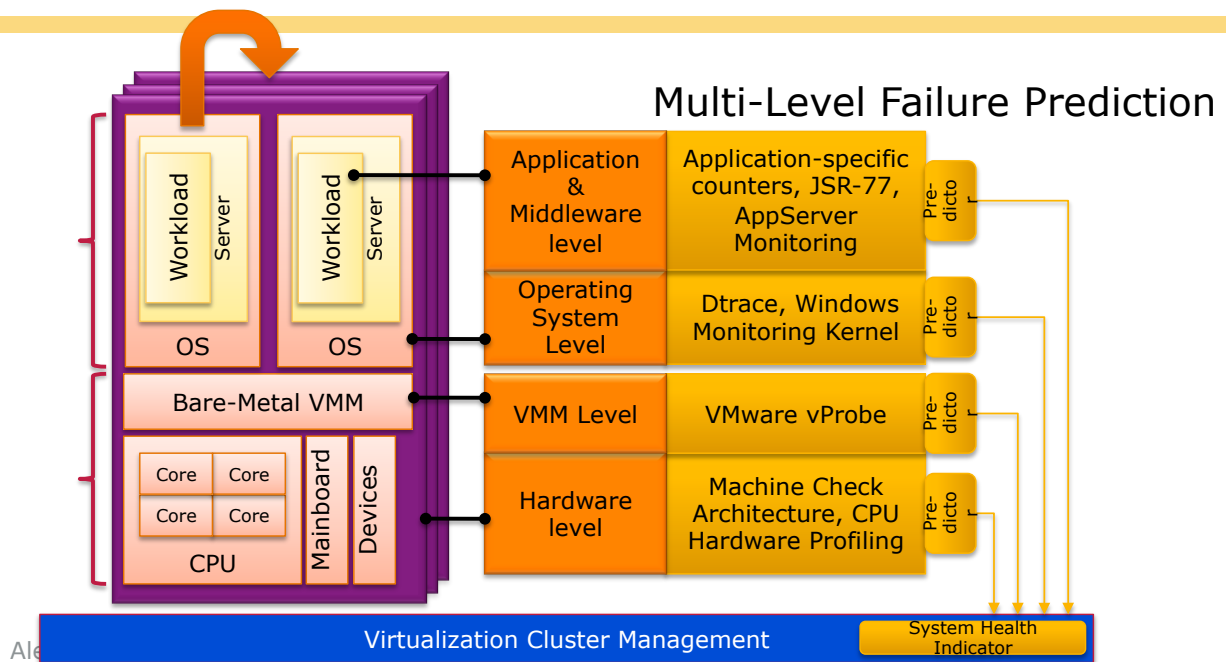


Ensemble learning:

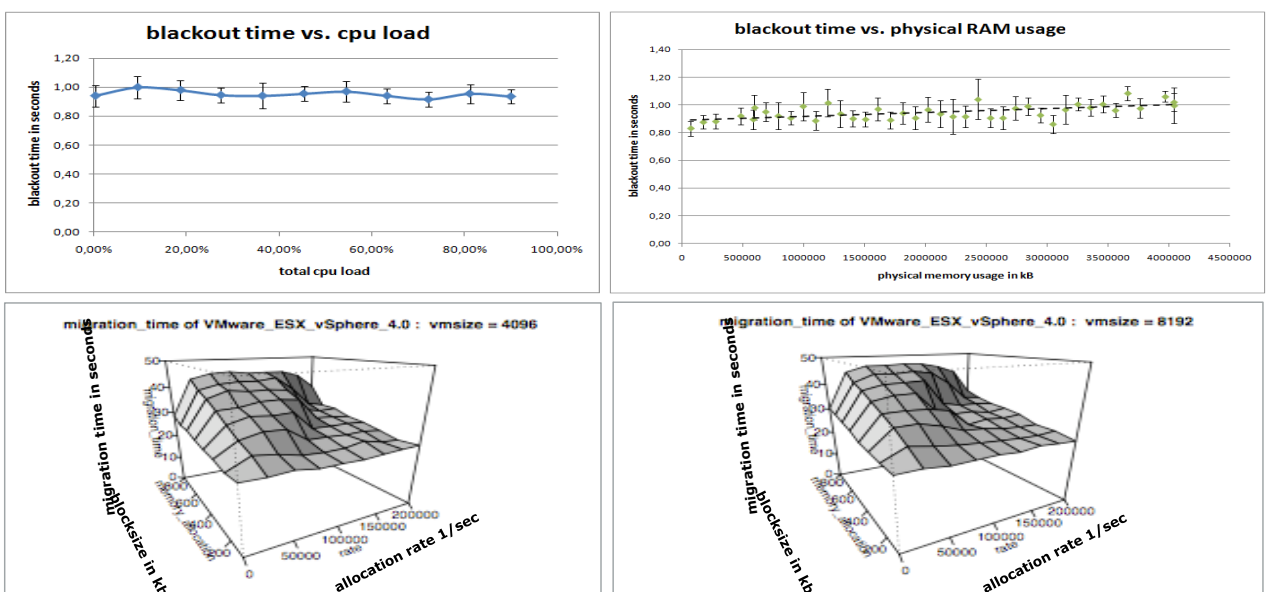
- Boosts accuracy – which failure-prone situations can best be identified by either hardware, OS, VMM failure predictors?
- Domain knowledge – operating system vendors know their system best and can provide the most advanced predictor on OS level
- Pluggable – domain predictors provided by an application vendor can easily be integrated into our anticipatory virtualization architecture
- Ensemble-learning can combine predictions across all system levels

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Our Idea: Global System Health Indicator



VM Migration – how long does it take? VMWare ESX 4



Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Agenda

1. Operating System Experiments – the Windows Case
2. InstantLab
3. Demo
4. Research Questions
5. Conclusions

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Applying it to the Cloud

- Servers have evolved – cloud will too
 - Ever growing number of CPU cores
 - Tremendous amounts of memory
- Reliability will become the most sought-after feature of future server systems
 - Higher density, integration levels in future CPUs will lead to multi-bit faults
 - Failure prediction and VM migration as promising concept
- Must have fault isolation boundaries (LPARs, blades)
- Cloud will embrace new programming and management models

Alexander Schmidt, Andreas Polze | Cloud Futures 2011 | June 2, 2011

Computer architecture drives changes in system software

Servers have evolved...

- New form factors
- Higher density
- Standard architectures
- Multicore/multithreaded

Advances in operating systems

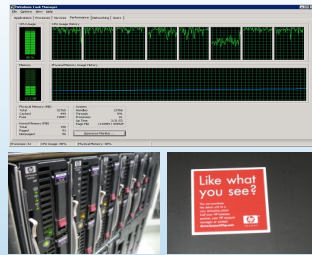
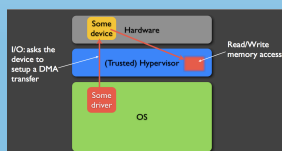
- Virtualization
- Thrustworthiness/security
- Clustering
- Need for new programming models, SW Architectures, Services

Virtualization problems

- Security: extended attack surface
- Virtualization-based malware
- Must trust hypervisor



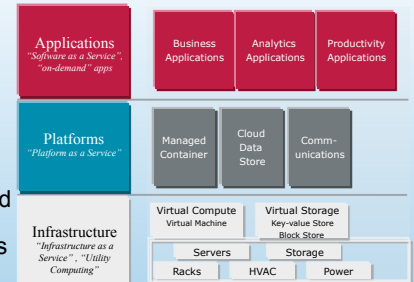
Intel VT-x, AMD Pacifica



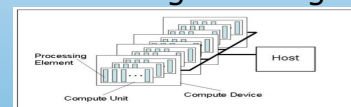
Cloud Computing – the three layers

Challenges:

- Has to abstract underlying hardware
- Be elastic in scaling to demand
- Pay per use basis



Hybrid Computing OpenCL: New Programming Models



One Host + one or more Compute Devices

- Each Compute Device is composed of one or more Compute Units
- Each Compute Unit is further divided into one or more Processing Elements

Alexander Schmidt, Andreas Polze / Cloud Futures 2011, June 21, 2011

Andreas Polze, Operating Systems and Middleware