



Internet Suspend/Resume

Michael Kozuch, M.Satyanarayanan



Big Picture

- Identify a '*new Capability*' for mobile computing that mimics opening and closing of laptop without physical transport
- Problem: how do we store and restore computing state ?
- Proposed Solution: By Layering Virtual machine technology on distributed file system.



Other Solutions

- Distributed File System (Coda/AFS)
 - Only Persistent state saved
 - Heavy weight
- Process migration (Sprite)
 - Too complex
 - External references ?? (forwarding)
 - Matching OS
- Thin Client Technology
 - Requires constant connectivity



VMM

- Layer of Software that sits directly on raw hardware and exports a '*virtual machine abstraction*'
- *Properties*
 - Encapsulates all Machine State
 - Same device interface irrespective of underlying hardware
 - File System can be used as backup



VM & DFS Solution

- VMM encapsulates all state in files
- Move files to new host with 'similar hardware' architecture
 - Use distributed file system as transport mechanism
 - Explicit copy-in/copy-out files from file system
- Is the idea *new /novel* ??



VM solution

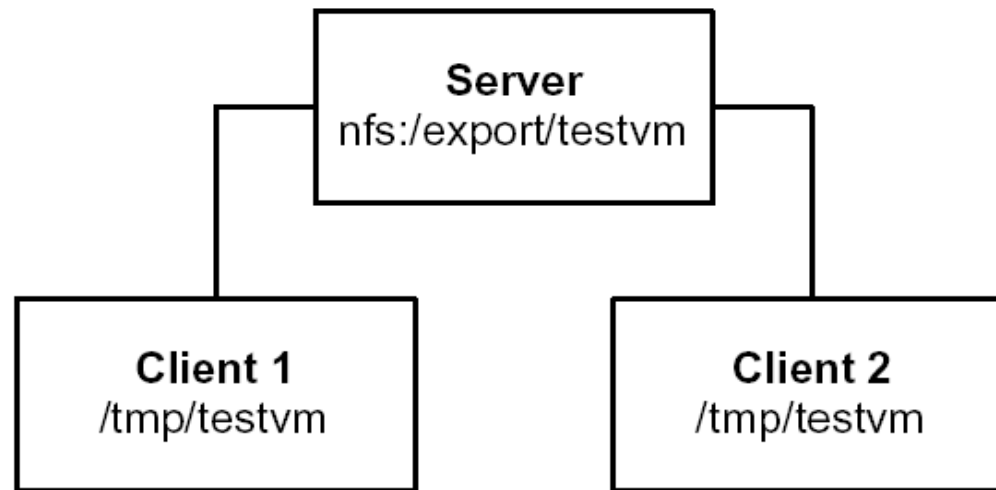
- Advantages

- *Easier* than process migration
- Machine level migration so no resource migration problem
- Migrating network connection
- Tolerant to greater disparity between systems
- Why copy-in/copy-out ?

- Disadvantage

- Amount of state to be transferred

Proof Of Concept



All machines are 1.7 GHz Pentium 4, single processor computers with 512 MB DRAM running Red Hat Linux 7.2 and NFS v3. The client machines are also running VMware Workstation 3.0, and the network connections are 100 Mb/s Ethernet.



Test System

Filename	Size	Function
testvm.cfg	1KB	Configuration information specifying number and type of virtual hardware components
testvm.vmdk	1GB	Data on virtual disk drive (Configured to be 2GB, but only 1GB is currently being used.)
testvm.nvram	8KB	State of non-volatile system memory (CMOS RAM)
testvm.vmsx	134MB	<i>Only present if VM is suspended.</i> Stores the current system state of the VM (DRAM, processor, and devices).
testvm.log	30KB	Log file used for debugging.

These are the files used by VMware Workstation to represent the suspended state of *testvm*, the VM used in our test system. As mentioned in Section 3.1, *testvm* is configured with 128MB of main memory and a 2GB disk. The guest OS is Windows XP.



Performance

Event	No Compression	With Compression
(Cold) Resume	125 (0.2)	73 (4.3)
Warm Suspend	114 (5.1)	158 (1.4)
Cold Suspend	146 (19.6)	158 (0.9)

This table shows the average time, in seconds, of suspend and resume operations under cold and warm file cache conditions. All experiments were repeated three times, and the observed standard deviations are shown in parentheses.

Figure 3. Suspend/Resume Times in *testvm*



Shortcomings

- Imperfect abstraction of host platform
 - KVM
 - is this the only abstraction problem ?
- File Location
 - Resume/suspend should be in same directory
- Lack of remote triggering of VM



Future Work

- Improved Security
 - Content security (encryption) time ??
 - Mutual validation (TCPA)
- Smart Coping
 - Migration between fixed locations
 - Temporal locality (Pure Demand fetching)
 - Synthesize state at resume site
(same OS/software files)



Future Work(contd)

- Proactive State transfer
 - Predict resume site
- Multiple Back Ends
 - Improve scalability/performance
 - Flexible on consistency
- Incremental Reconstruction
 - Demand reconstruction with proactivity



Future Work – Results!!!

Policy	SOP	SICC	RSD
Baseline	2060 (29)	2020 (82)	2034 (23)
Proactive	2.5 (0.15)	2.5 (0.14)	3 (0.02)
Demand	27 (1.9)	33 (5.1)	18 (1.0)

Average resume time in seconds and standard deviation for three iterations of the SOP, SICC, and RSD benchmarks.

* From '*Efficient State Transfer for Internet Suspend/Resume*'



Conclusion

- *'First'* to identify the concept
- Why ?
 - VM technology relatively new
 - Ignorance on part of research community
 - *Microsoft !!!*



Questions/Discussion

- *Contribution* of the paper ?
- Is there a need for such a technology ?
 - good to have one
- How well does it act in case of '*weakly connected devices*' ?
- Is latency acceptable ?