



IP Mobility to Support Live Migration of Virtual Machines Across Subnets

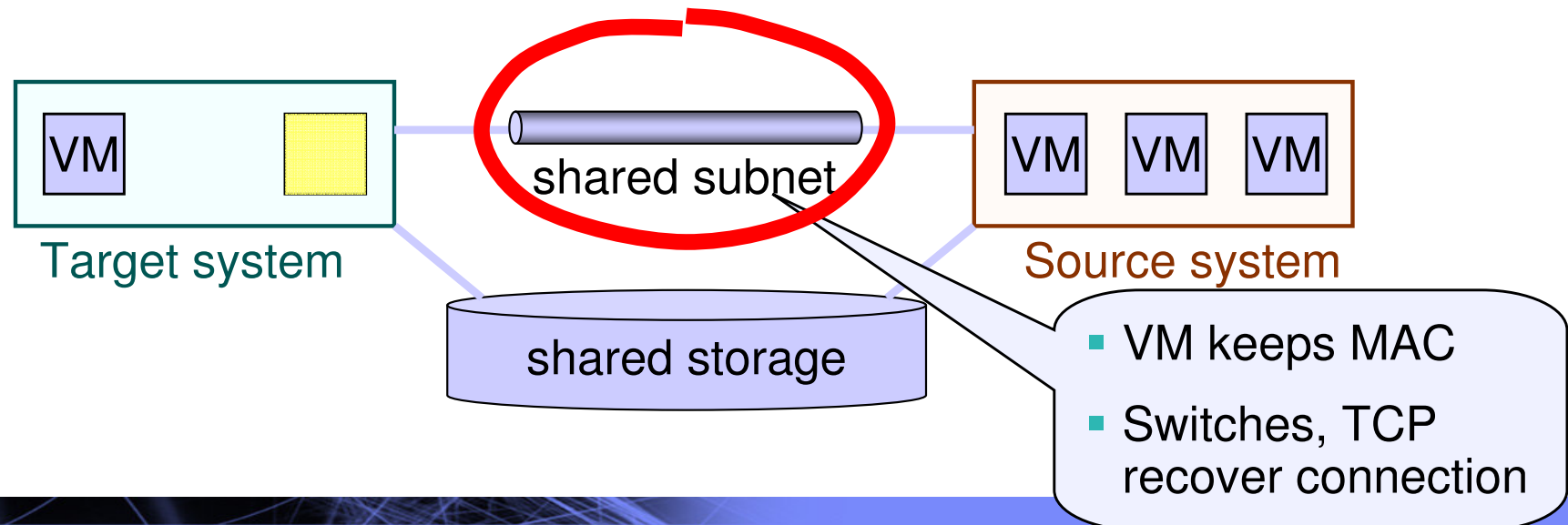
Dean Lorenz

Ezra Slivera, Gilad Sharaby, Inbar Shapira

IBM Haifa Research Labs / System Technologies & Services

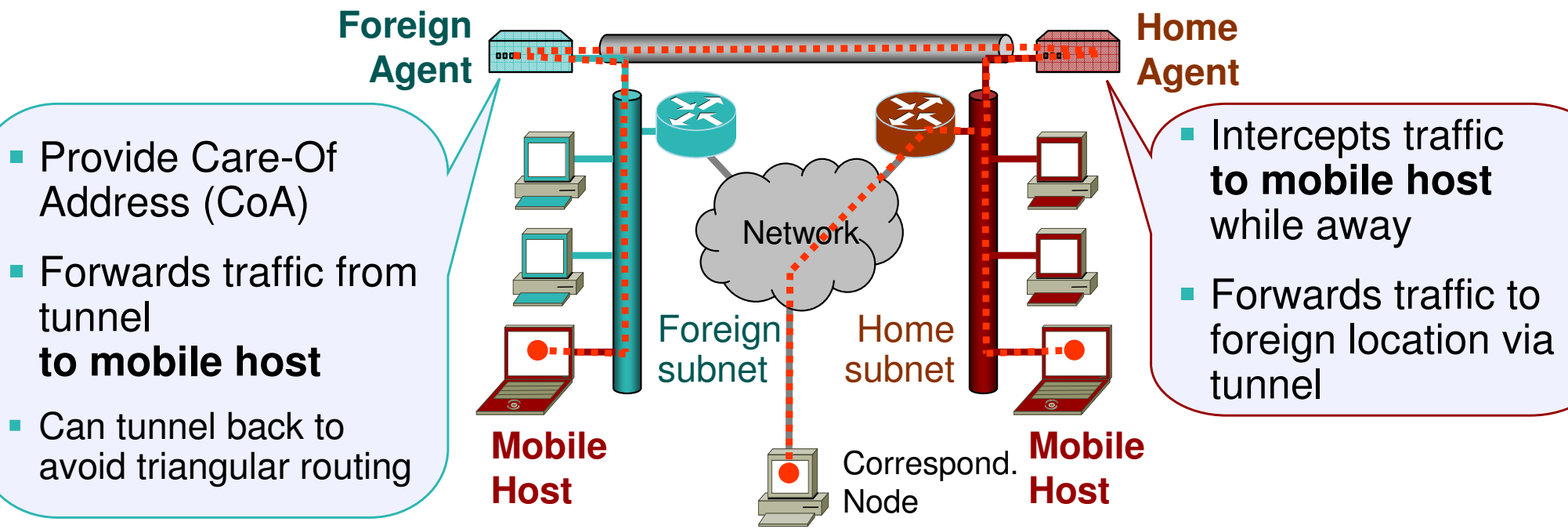
IP Mobility to Support Live Migration of VMs Across Subnets

- Motivation - migrate VMs between physical hosts
 - Good for load-balancing, hardware maintenance, etc.
 - **No (a.k.a. minimal) down time**
- How it works
 - Create a VM on target
 - Mirror source VM state (memory)
 - Suspend source VM, **synchronize**, resume VM on target system



IP Mobility to Support Live Migration of VMs Across Subnets

- Motivation:
 - Support mobile hosts when moving across IP subnets
- The problem:
 - IP address defines **location** not **identity**



Goal

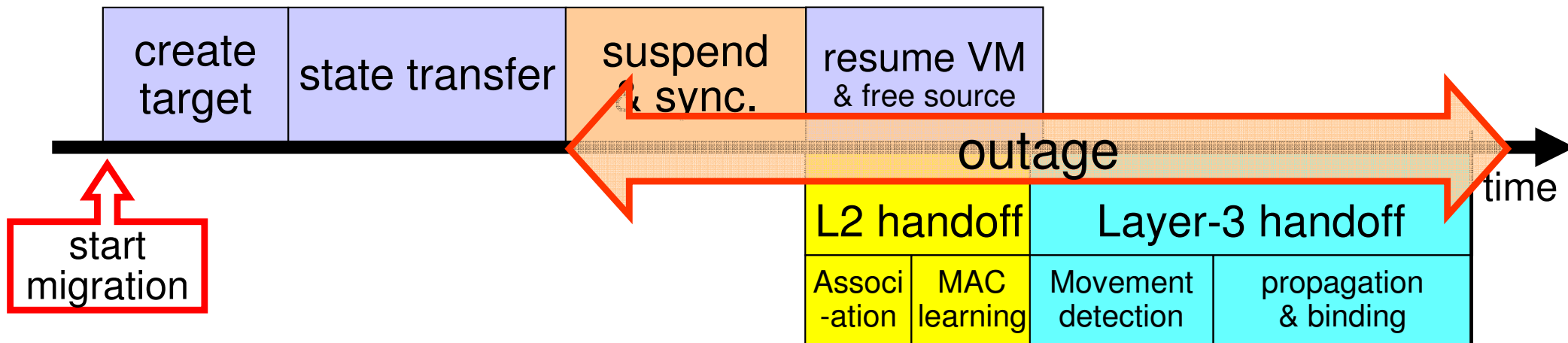
Enable Live Virtual Machine Migration Across Subnets

- Adapt solutions to the IP Mobility problem
 - Treat a migrating VM as a mobile device
- Are these the same problem?
 - What are the unique properties of VM migration?
 - Can we exploit them to do better than std. IP mobility?

Outline

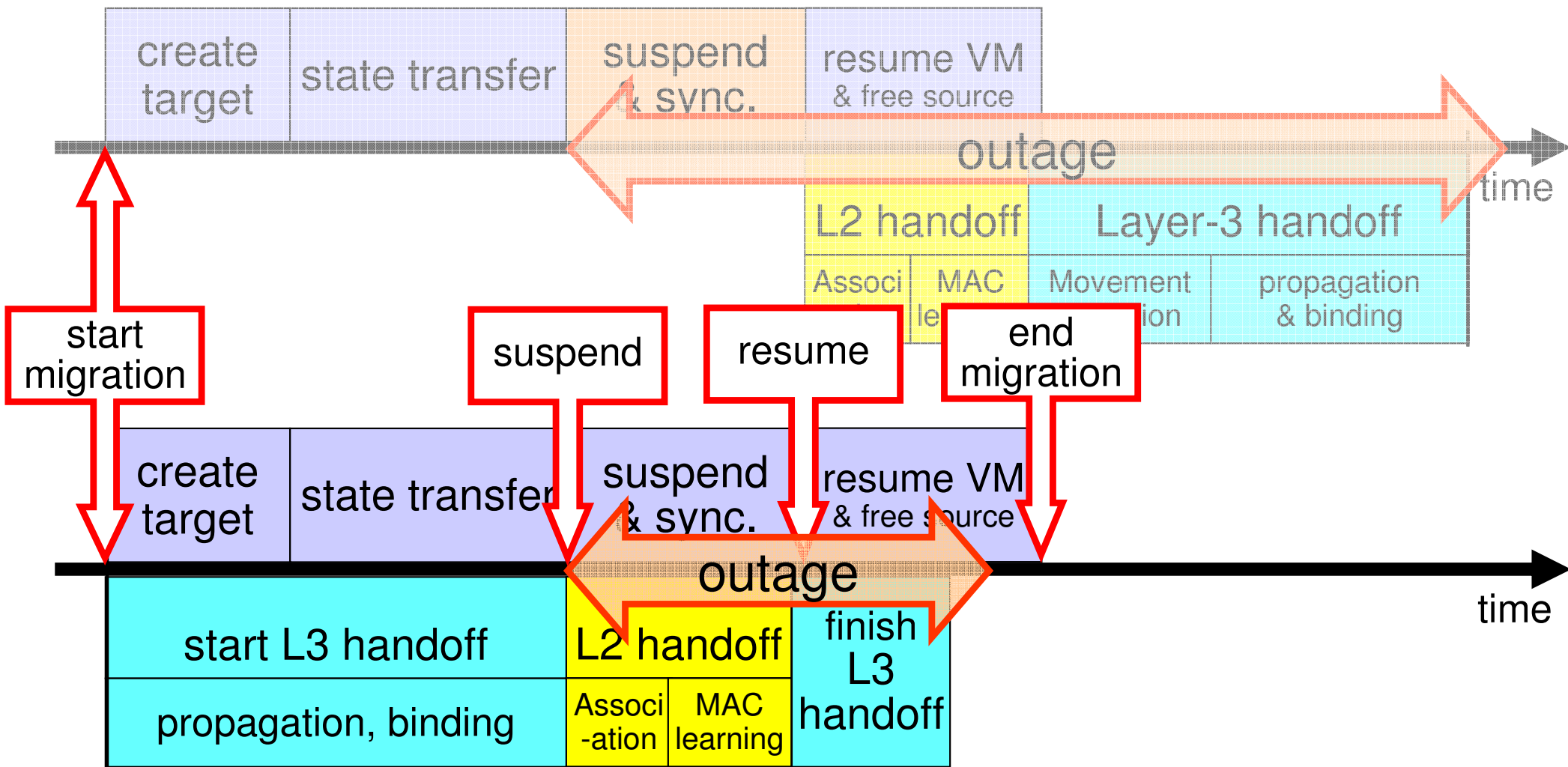
- Background & Goals
- IP Mobility Optimization for Live VM Migration
- Implementation
- Conclusions

Application Outage During Migration



- Total application outage is (VM down time) + (Network down time)
 - Network handoff starts **only after** VM is resumed on target
 - Layer-3 handoff added for cross-subnet migration
 - Application resumes only after higher levels recover (TCP, etc.)

Application Outage During Migration



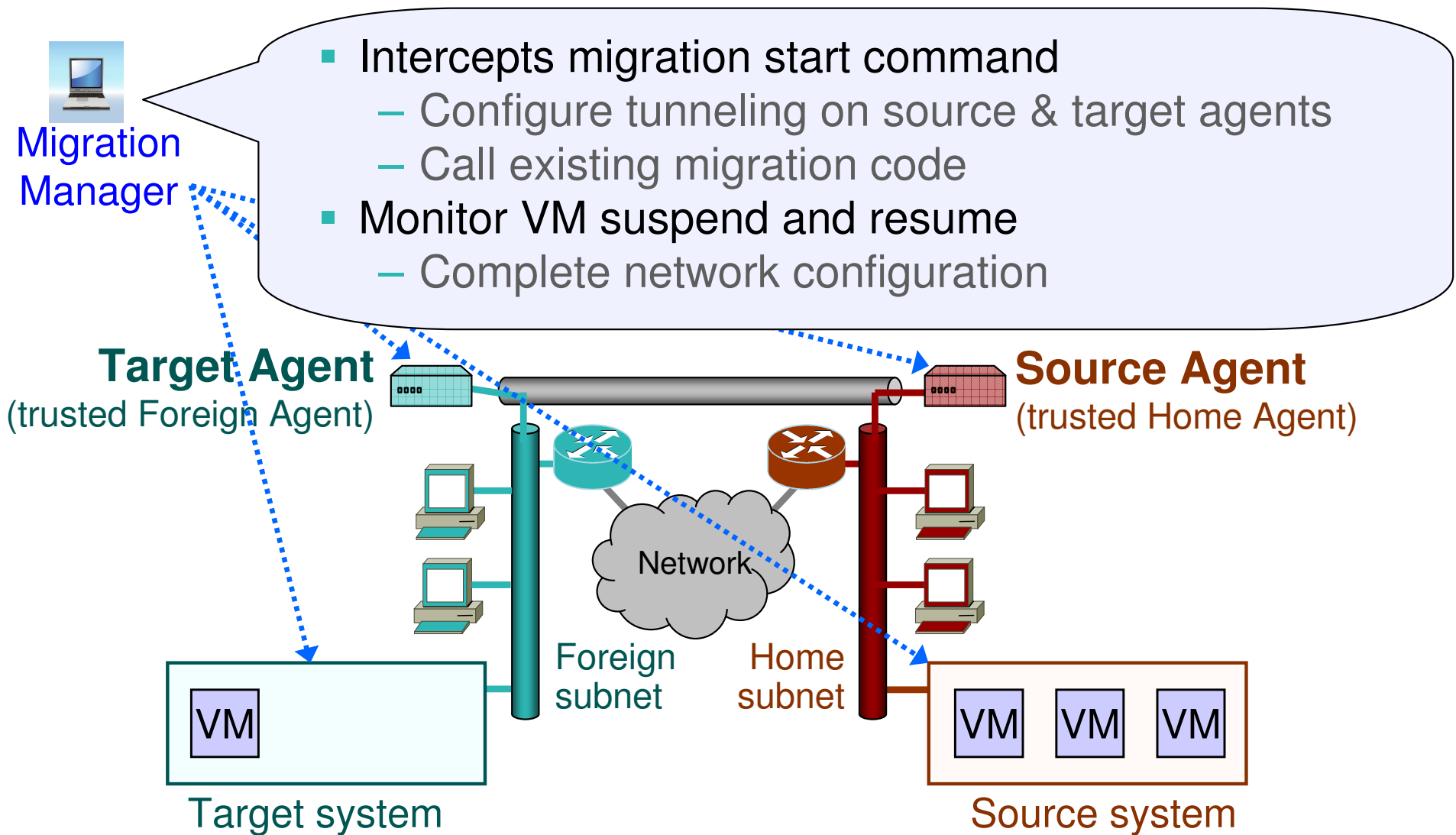
Properties of VM Migration

	IP Mobility	VM Migration
Network QoS	Typically wireless	Wired
Collaboration	Sometimes	Always
Trust	Un-trusted foreign network Authentication for home net	Hypervisors fully trusted Established prior to migration
Mobility agents	Require discovery	Always exist, More powerful
Transparent	Typically no	Yes
Synchronized	No host moves, network reacts	Yes (need events)

Outline

- Background & Goals
- IP Mobility Optimization for Live VM Migration
- Implementation
- Conclusions

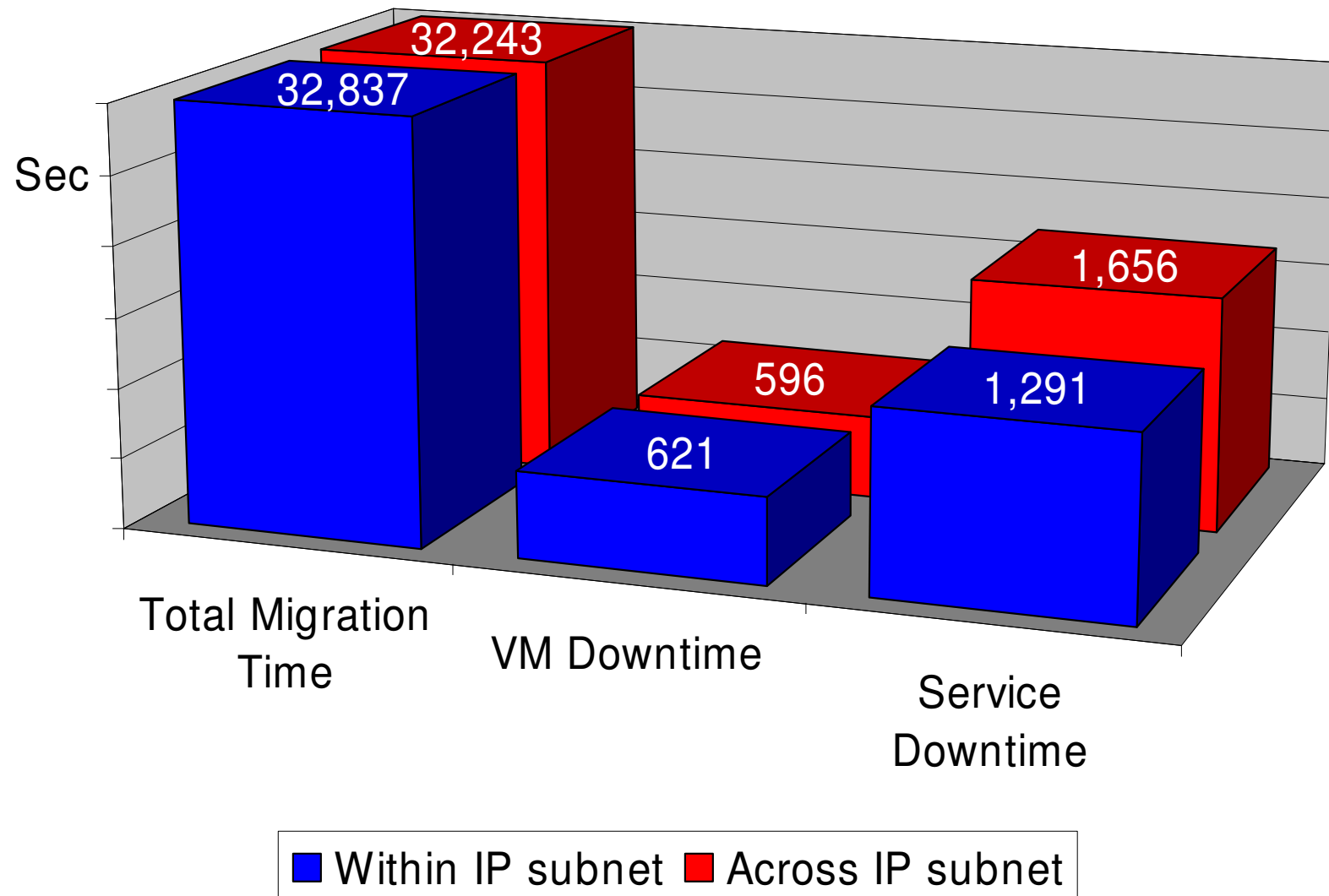
Implementation Architecture



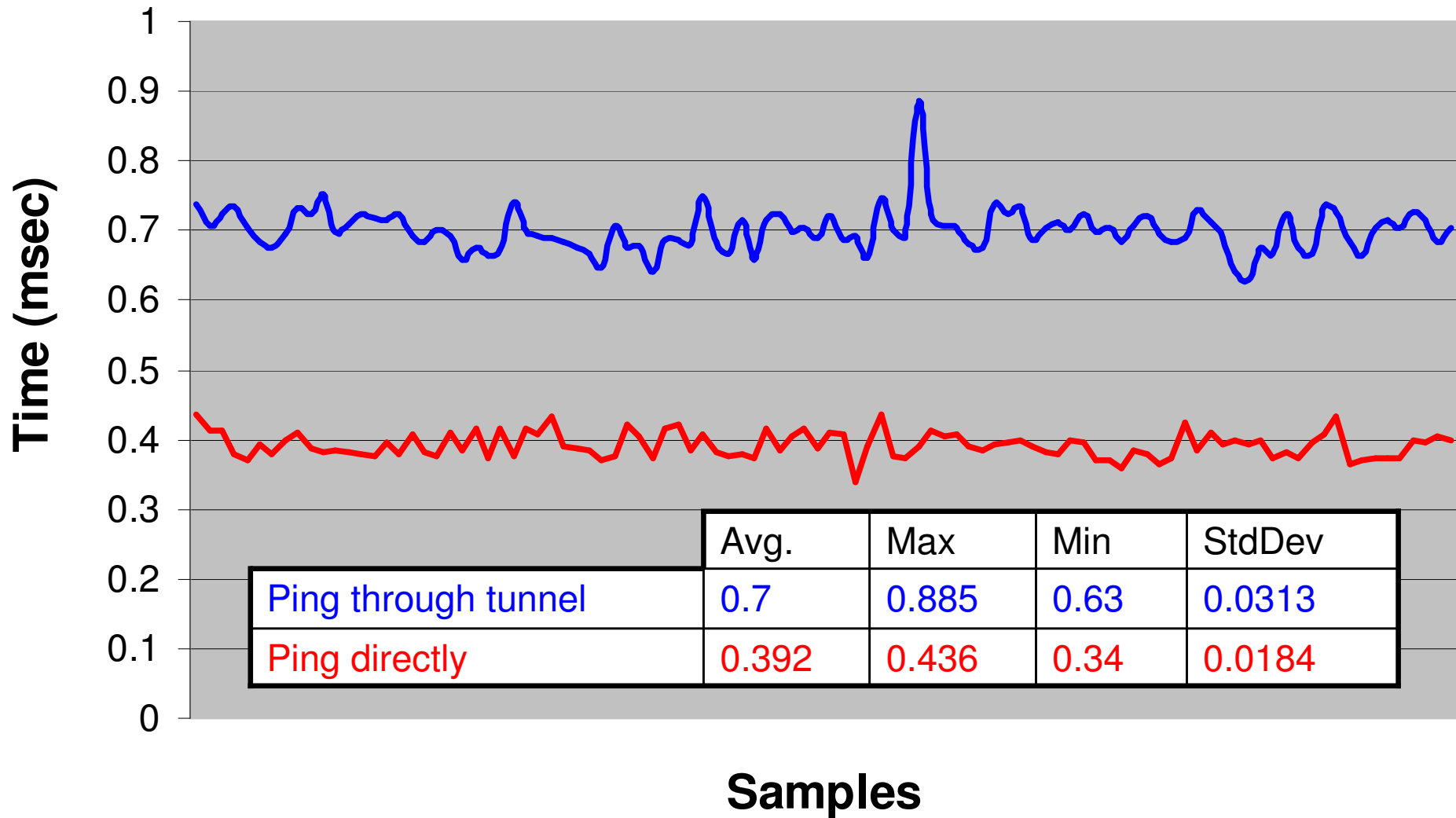
Evaluation I: It works!

- We enabled live VM migration across subnets
 - Working implementations for Xen, pHype
- Transparent
 - Outside the hypervisor
 - Can work on any virtualization technology (e.g., Xen, VMWare)
 - No changes to VMs
- Simple agents for forwarding & tunneling
 - No special code is installed on agents

Evaluation II: Downtime



Evaluation III: Steady State Latency



Outline

- Background & Goals
- IP Mobility Optimization for Live VM Migration
- Implementation & Evaluation
- Conclusions

Conclusions

- We extend VM migration across subnets
 - Can adapt existing technology
 - Similar, but different
- Promote open standards migration events
 - Also useful for other optimizations (storage, application)
- Future / alternatives
 - Forward events into existing IP mobility
 - No special agents, allow network-level optimization
 - Integrate solution into hypervisor
 - No extra agents, better isolation, scalability
 - Further optimizations
 - E.g., shortcut routing, for better steady-state performance



The End

Contact: dean@il.ibm.com

Solutions to IP Mobility

- Mobile IP
- Above network layer
 - Mobility aware protocols (TCP-MH, MSOCS, SIP)
 - Mobility aware applications (can handle IP change)
- Below network layer
 - Layer 2 routing, forwarding, and tunneling
 - Ignore IP — just send stuff to the right place
- Overlay networks
 - Virtual network topology
 - “mobile wires”
 - Works best for all virtual environments