

Lecture #12

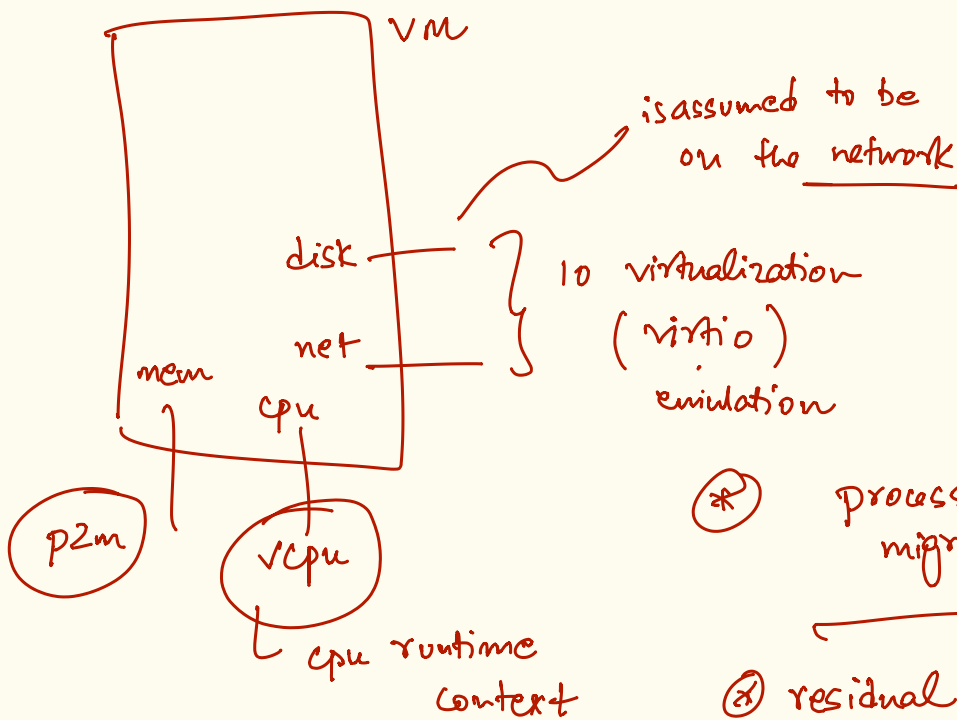
CS 695

mgmt. primitives/actions due to VMs.

- sharing, ballooning, migration, high availability, image introspection
content dedup

Live Migration with Virtual Machines

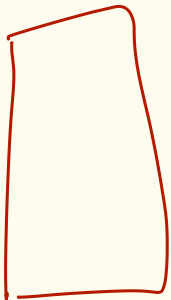
usenix
ATC
2005



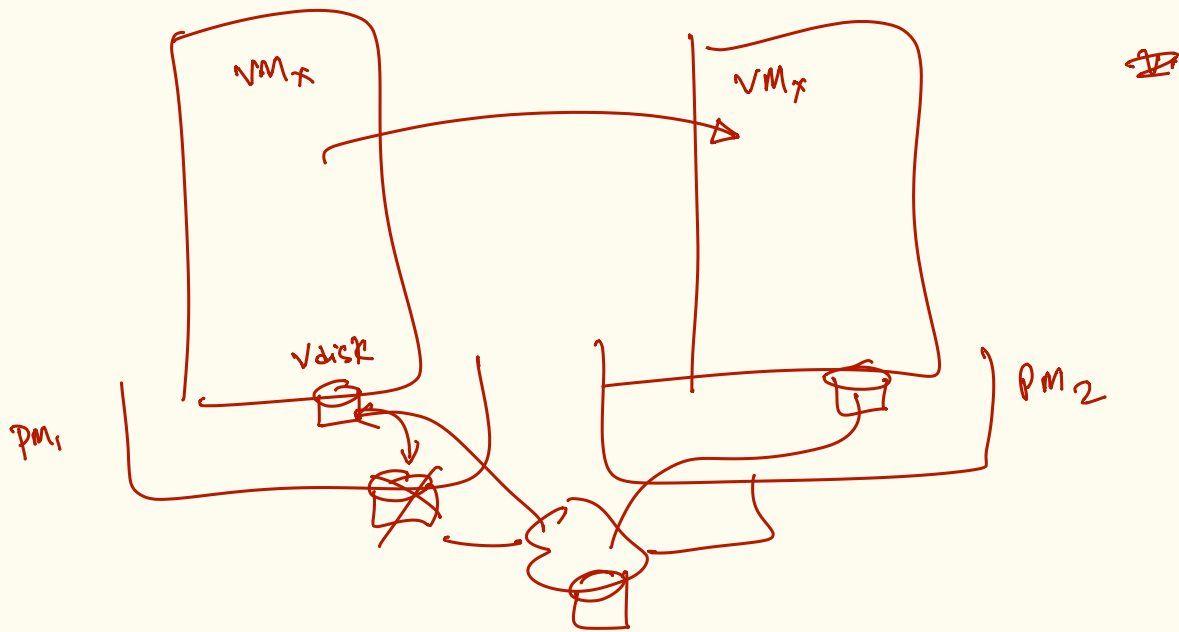
(*) process migration

(x) residual dependency
(challenge)

vm migration



- IPC primitives
 - ↳ signals, shared memory
- parent-child dependency
- Pid namespace
- open file descriptors



VM migration

benefits

~ move with "all" residual dependencies

~ primitive for balancing resources
managing

~ maintenance ~ H/W & S/W

esp. for
all OS
state &
abstractions

Techniques for VM migration

rtasc

stop-and-copy

iterative pre-copy
+
stop & copy

post-copy

~ move all state of VM from PM₁ to PM₂

+ CPUs ~ regs. vmcs

+ memory of VM (p2m)

- stop VM

- copy ^{all} memory
state of VM

- copy CPU state

- reattach devices

- start VM on new PM

* metrics of migration : (correctness)

- migration time, down time ~ service / VM not available for interaction

~ resources consumed ~ network resources
due to migration (source & destination)

~ interference w/ services / applications.

iterative
Pre-copy migration - (live)

- copy pages of VM to destⁿ while VM executes

~ in the time taken to T_x pages
VM memory is changing — dirty set

~ identify dirty set

~ re T_x dirty set in next iteration.

~ each iteration keeps pushing dirty pages.

~ at some point stop iterations & perform
stop-and-copy

dirty is small

dirty not changing
pages

~ max # iterations

~ when all at destination
restart VM.

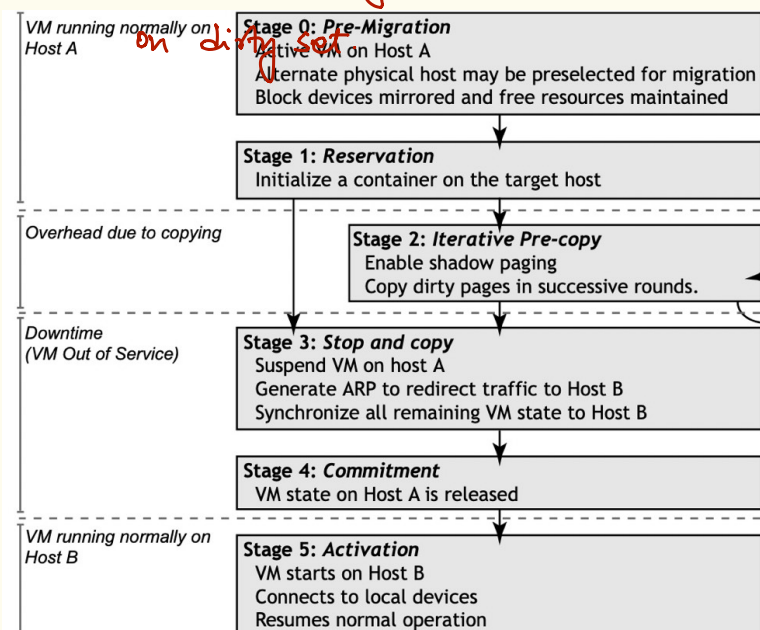
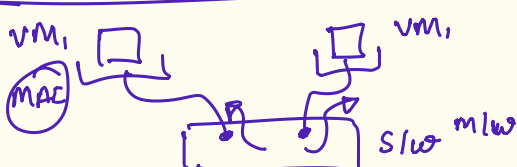
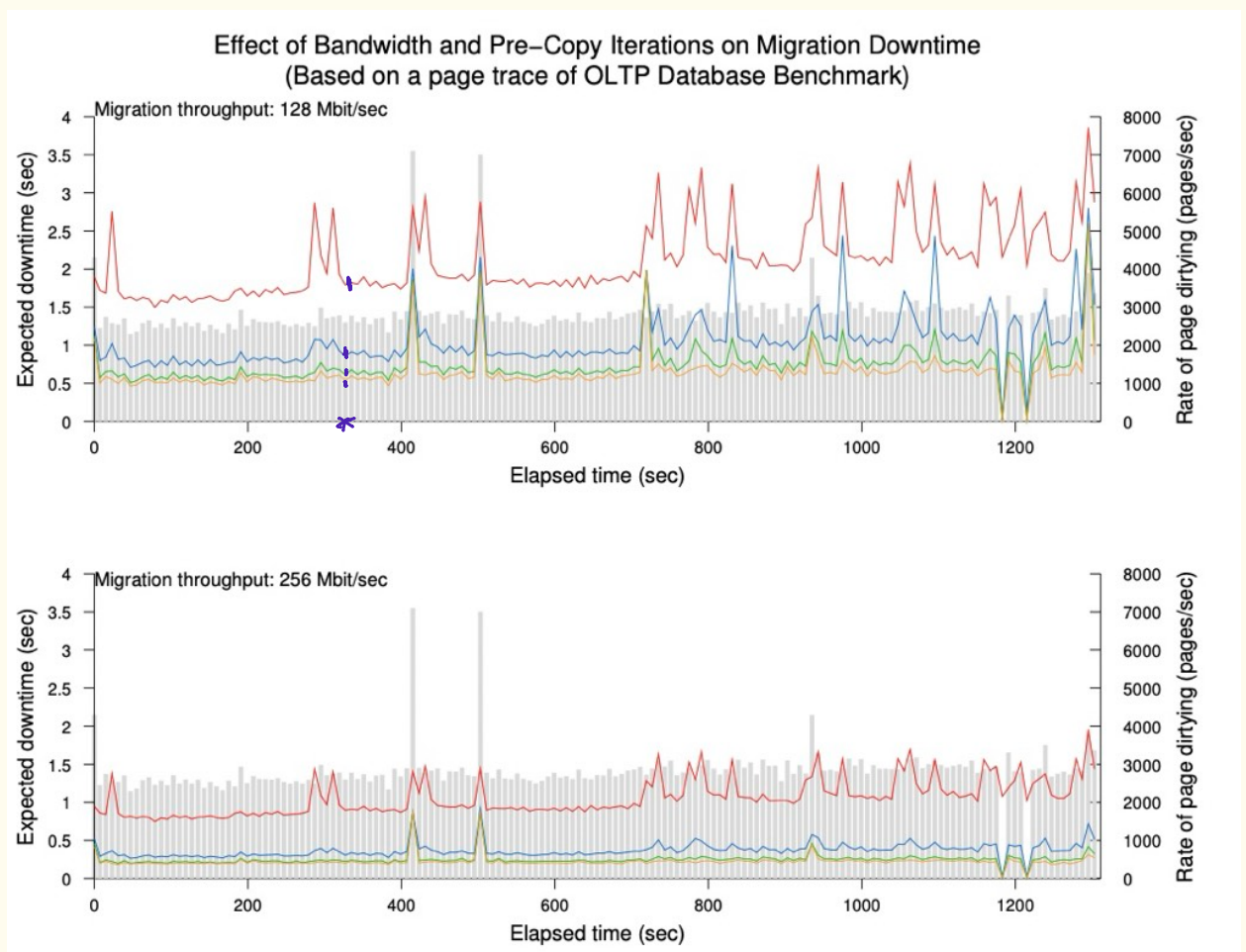
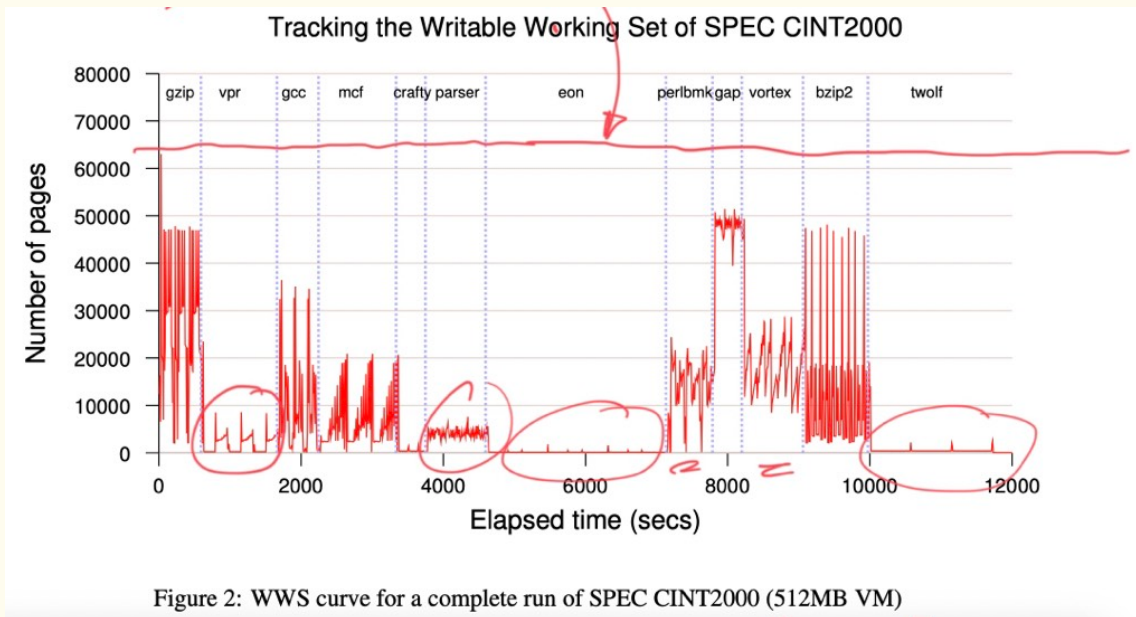


Figure 1: Migration timeline





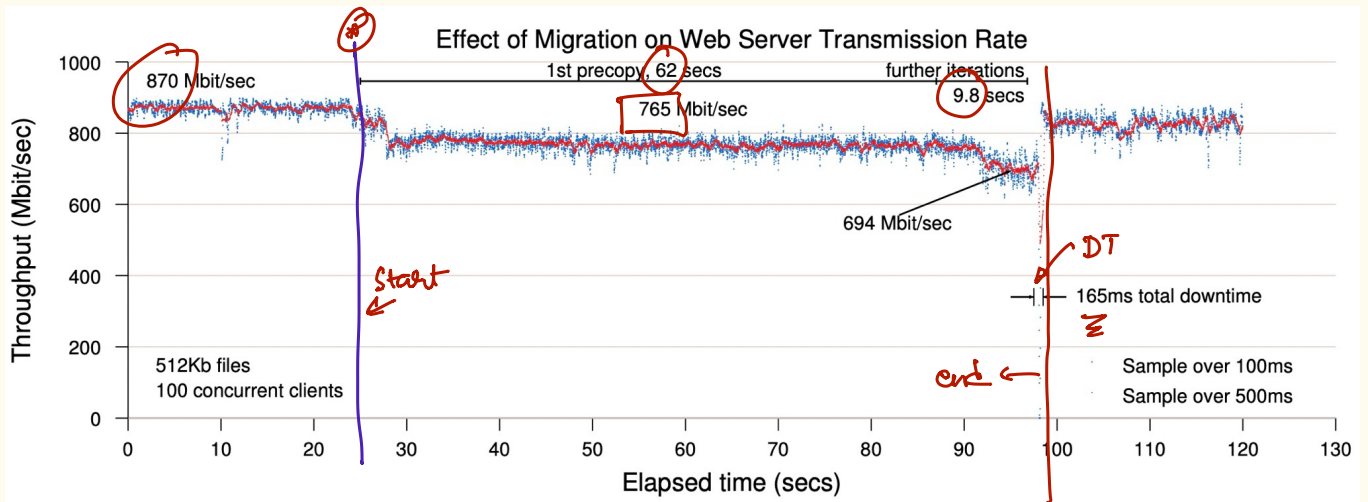


Figure 8: Results of migrating a running web server VM.

←----- MT -----→

* post-copy

- copy CPU state of VM } & some memory
- start VM }
- handle page faults on the n/w.