Virtual Clusters & Resource Management

- What is a physical cluster?
- What is a virtual cluster?
- live migration of VMs, memory and file migrations, and dynamic deployment of virtual clusters.

Virtual Cluster

- As with traditional physical servers, virtual machines (**VMs**) can also be **clustered**. A **VM cluster** starts with two or more physical servers;
- we'll call them Server A and Server B.
- In simple deployments if Server A fails, its workloads restart on Server B

Virtual Cluster features

- **HA:** virtual machines can be restarted on another hosts if the host where the virtual machine running fails.
- **DRS** (Distributed Resource Scheduler): virtual machines can be load balanced so that none of the hosts is too overloaded or too much empty in the cluster.
- **Live migration:** of virtual machines from one host to other.

Virtual Clusters & Resource Management

- In a traditional VM initialization, the administrator **manually writes** configuration information/specify the configuration sources.
- With many VMs, an inefficient configuration always causes problems with overloading or underutilization.

Virtual Clusters & Resource Management

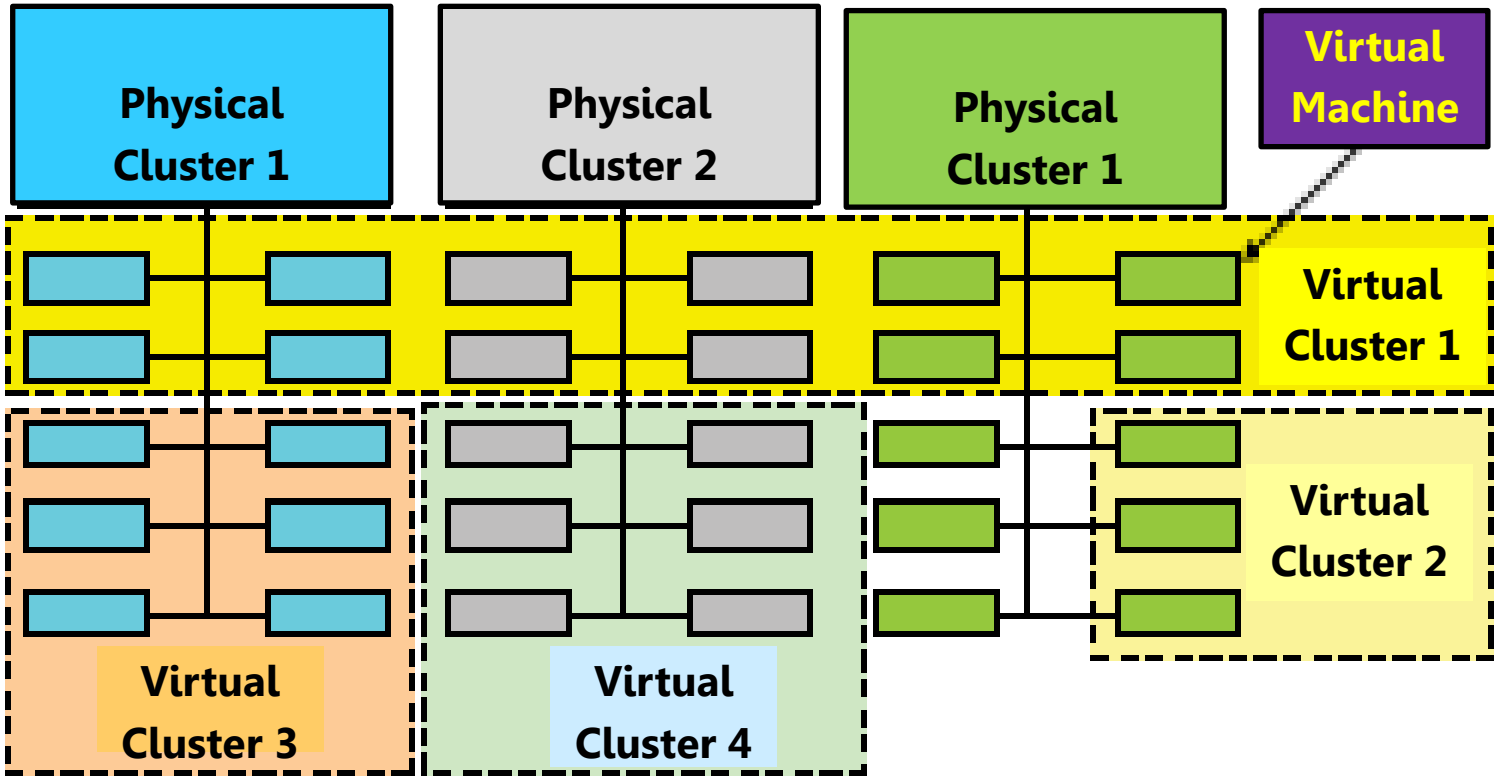
- **Amazon's EC2** provides elastic computing power in a cloud. EC2 permits customers to create VMs and to manage user accounts over the time of their use (resizable capacity).
- **XenServer** and **VMware ESXi** Server support a bridging mode which allows all domains to appear on the network as individual hosts.
- With this mode VMs can communicate with one another freely through the virtual network interface card and configure the network automatically.

Virtual Clusters

- Virtual clusters are built with VMs installed at distributed servers from one or more physical clusters.

- The VMs in a virtual cluster are **interconnected logically by a virtual network** across several physical networks

Virtual Clusters



Courtesy of Fan Zhang, Tsinghua University

Provisioning of VMs in Virtual Clusters

- *The provisioning of VMs to a virtual cluster is done dynamically* to have some interesting properties:

Provisioning of VMs in Virtual Clusters Conti…

1. The virtual cluster nodes can be either physical or virtual machines. Multiple VMs running with different OSes can be deployed on the same physical node.
2. A VM runs with a guest OS, which is often different from the host OS.
3. The purpose of using VMs is to *consolidate* multiple functionalities on the same server. This will greatly *enhance server utilization and application flexibility*.

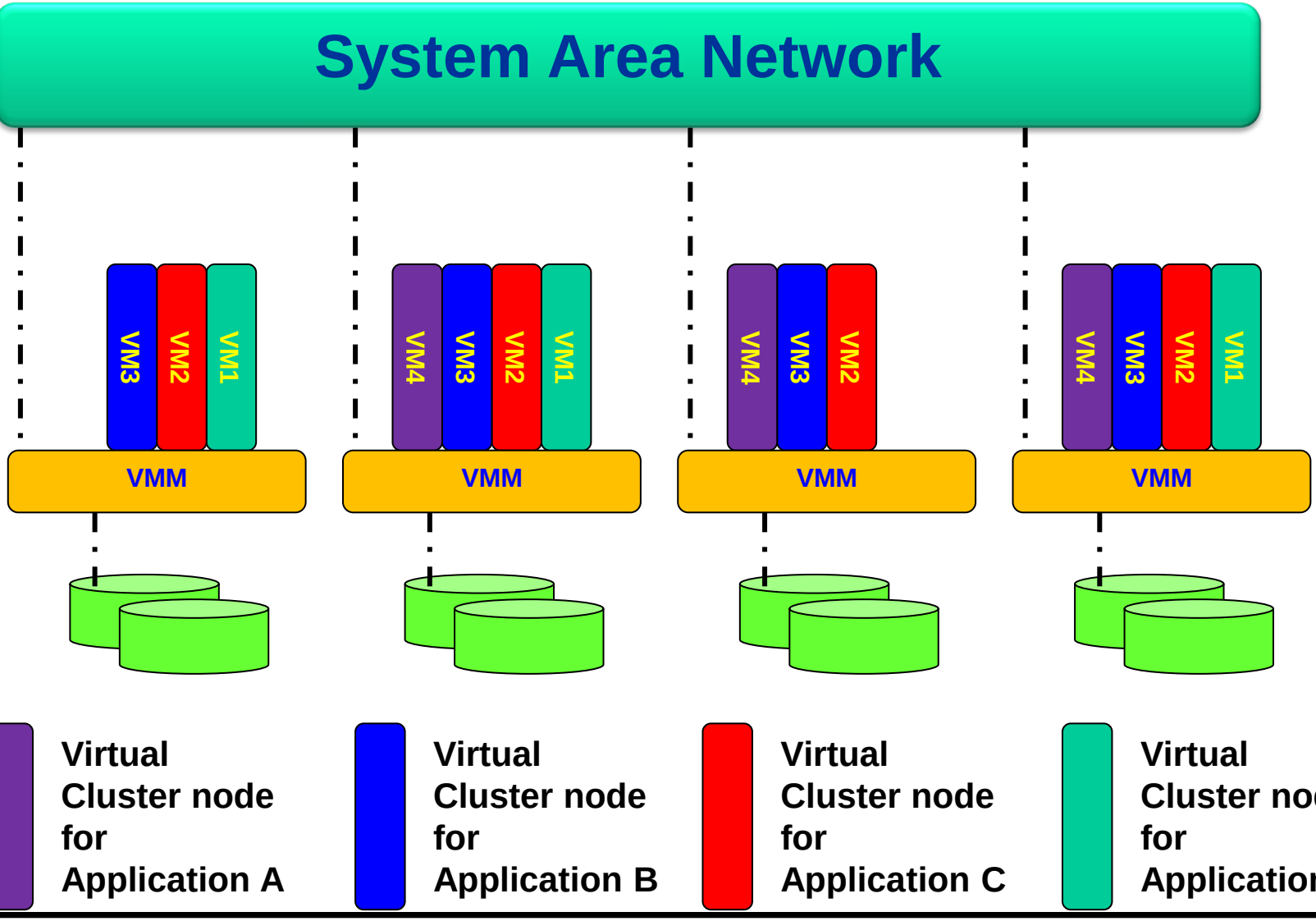
Provisioning of VMs in Virtual Clusters Conti…

4. VMs can be *colonized (replicated)* in multiple servers for the purpose of promoting distributed parallelism, fault tolerance, disaster recovery.
5. The size of a virtual cluster *can grow or shrink dynamically.*
6. The failure of any physical nodes may disable some VMs installed on the failing nodes. But the failure of VMs will not pull down the host system.

Virtual Clusters Management

- It is necessary to effectively manage VMs running on virtual clusters and consequently build a high-performance virtualized computing environment
- This involves
  - virtual cluster deployment,
  - monitoring and management over large-scale clusters, resource scheduling, load balancing,
  - server consolidation, fault tolerance, and other techniques

Virtual cluster based on application partitioning



Virtual Clusters Management Conti…

- Since large number of VM images might be present, the most important thing is to determine *how to store those images in the system efficiently*
- Apart from it there are common installations for most users or applications, such as OS or user-level programming libraries.
- These software packages can be preinstalled as templates (called template VMs).

Deployment

- There are *four steps* to deploy a group of VMs onto a target cluster:
  - *preparing the disk image,*
  - *configuring the VMs,*
  - *choosing the destination nodes,* and
  - *executing the VM deployment command on every host.*

Deployment Conti…

- Many systems use **templates** to simplify the disk image preparation process.
- A **template** is a disk image that includes a preinstalled operating system with or without certain application software.

Deployment Conti…

- Users choose a proper template according to their requirements and make a duplicate of it as their own disk image.
- Templates could implement the *COW (Copy on Write) format*. A new COW backup file is very small and easy to create and transfer.
- Therefore, it definitely reduces disk space consumption.

Copy-on-write

- An optimization strategy in which if multiple callers ask for resources which are initially indistinguishable, give them pointers to the same resource.
- This function can be maintained until a **caller tries to modify** its "copy" of the resource, at which point a true private copy is created to prevent the changes becoming visible to everyone else.

Copy-on-write

- All of this happens transparently to the callers.
- The primary advantage is that if a caller never makes any modifications, no private copy need ever be created.
- All changes are recorded in a separate file preserving the original image. Several COW files can point to the same image to test several configurations simultaneously without jeopardizing the basic system.

Copy-on-write

- Unlike the snapshot, the copy-on-write uses multiple files and allows to simultaneously run multiple instances of the basic machine.

Deployment Conti…

- In addition, VM deployment time is much shorter than that of copying the whole raw image file.
- *VM is configured with a name, disk image, network setting, and allocated CPU and memory.*
- One needs to record each VM configuration into a file. However, this method is inefficient when managing a large group of VMs

Deployment Conti…

- VMs with the same configurations could use *pre-edited profiles to simplify the process*. In this scenario, the system configures the VMs according to the chosen profile.
- Most configuration items use the same settings, while other items, such as UUID, VM name, and IP address, are assigned with automatically calculated values

Live VM Migration Steps and Performance Effects Conti…

- When a VM fails, its role could be replaced by another VM on a different node, as long as they both run with the same guest OS
- a VM must stop playing its role if its residing host node fails
- This problem can be mitigated with *VM live migration*
- The migration copies the *VM state file* from the storage area to the host machine.

Live VM Migration Steps and Performance Effects Conti…

- There are four ways to manage a virtual cluster

  First way is to use a *guest-based manager*, by which the cluster manager resides on a guest system. In this case, multiple VMs form a virtual cluster

  - Ex. openMosix is an open source Linux cluster running different guest systems on top of the Xen hypervisor

Live VM Migration Steps and Performance Effects Conti…

- There are four ways to manage a virtual cluster
  - Second way is we can build a *cluster manager on the host systems*. The host-based manager supervises the guest systems and can restart the guest system on another physical machine.
    - Ex. A good example is the VMware HA system that can restart a guest system after failure.

Live VM Migration Steps and Performance Effects Conti…

- There are four ways to manage a virtual cluster
  - Third way to manage a virtual cluster is to use an *independent cluster manager* on both the host and guest systems. This will make infrastructure management more complex,

Live VM Migration Steps and Performance Effects Conti…

- There are four ways to manage a virtual cluster
  - Finally can use *an integrated cluster Manager on the guest and host systems*. This means the manager must be designed to distinguish between virtualized resources and physical resources.
  - Various cluster management schemes can be greatly enhanced when VM life migration is enabled with minimal overhead.

Live VM Migration Steps and Performance Effects Conti…

- *Virtual clustering plays a key role in cloud computing.*
- VMs can be live-migrated from one physical machine to another; *in case of failure*
- When a VM runs a live service, it is necessary to make a trade-off to ensure that the migration occurs in a manner that minimizes all three metrics.

Live VM Migration Steps and Performance Effects Conti…

- The motivation is to design a live VM migration scheme with
  - negligible downtime,
  - the lowest network bandwidth consumption possible, and
  - a reasonable total migration time

Live VM Migration Steps and Performance Effects Conti…

- A VM can be in one of the following four states.
  - An *inactive state* is defined by the virtualization platform, under which the VM is not enabled.
  - An *active state* refers to a VM that has been instantiated at the virtualization platform to perform a real task.
  - A *paused state* corresponds to a VM that has been instantiated but disabled to process a task or paused in a waiting state.
  - A VM enters the *suspended state* if its *machine file and virtual resources* are stored back to the disk.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Steps 0 and 1: Start migration. This step makes preparations for the migration, including determining the migrating VM and the destination host.
    - Although users could manually make a VM migrate to an appointed host, in most circumstances, the migration is automatically started by strategies such as *load balancing and server consolidation*.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Steps 2: Transfer memory.
  - Since the whole execution state of the VM is stored in memory, sending the VM's memory to the destination node ensures continuity of the service provided by the VM.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Steps 2: Transfer memory.
  - All of the memory data is transferred in the first round, and then the *migration controller recopies the memory data which is changed in the last round*.
  - These steps keep iterating until the dirty portion of the memory is small enough to handle the final copy.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Step 3: Suspend the VM and copy the last portion of the data.
  - The migrating VM's execution is suspended when the last round's memory data is transferred. Other non-memory data such as CPU and network states should be sent as well.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Step 3: Suspend the VM and copy the last portion of the data.
  - Here the VM is stopped and its applications will no longer run. This *"service unavailable"* time is called the *"downtime"* of migration, which should be as short as possible so that it can be negligible to users.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Steps 4 and 5: Commit and activate the new host.
  - After all the needed data is copied, on the destination host, the *VM reloads the states* and recovers the execution of programs in it, and the service provided by this VM continues.

Live VM Migration Steps and Performance Effects Conti…

- Live Migration of a VM consists of the following six steps:
  - Steps 4 and 5: Commit and activate the new host.
  - Then the network connection is redirected to the new VM and the *dependency to the source host is cleared*.
  - The whole migration process finishes by removing the original VM from the source host.

Live migration process of a VM from one host to another