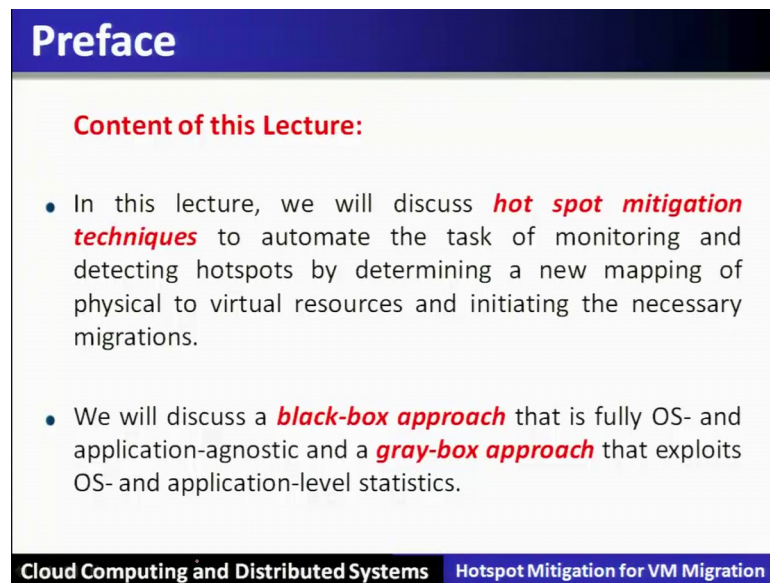


Cloud Computing and Distributed Systems
Dr. Rajiv Misra
Department of Computer Science and Engineering
Indian Institute of Technology, Patna

Lecture – 03
Hotspot Mitigation for Virtual Machine Migration

Hotspot Mitigation for Virtual Machine Migration; preface content of this lecture.

(Refer Slide Time: 00:24)



Preface

Content of this Lecture:

- In this lecture, we will discuss **hot spot mitigation techniques** to automate the task of monitoring and detecting hotspots by determining a new mapping of physical to virtual resources and initiating the necessary migrations.
- We will discuss a **black-box approach** that is fully OS- and application-agnostic and a **gray-box approach** that exploits OS- and application-level statistics.

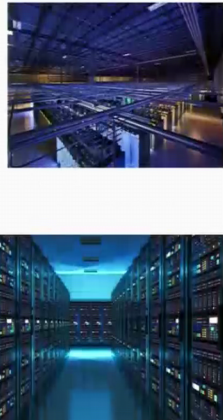
Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

We will discuss hotspot mitigation techniques and algorithms to automate the task of monitoring and detecting hotspots in the cloud system by determining new mapping of physical to the virtual resources and initiating the virtual machine migration. This all we will discuss in the topic hotspot mitigation techniques for virtualized data centers; here we will cover 2 different schemes one is called black box approach that is fully operating system and application agnostic. The other approach is called as a grey box approach that exploits operating system and application level statistics. We will also see the paper which has given these 2 approaches to solve the hotspot mitigation problem using virtual machine migration for virtualized data center.

(Refer Slide Time: 01:38)

Enterprise Data Centers

- **Data Centers are composed of:**
 - Large clusters of servers
 - Network attached storage devices
- **Multiple applications per server**
 - Shared hosting environment
 - Multi-tier, may span multiple servers
- **Allocates resources to meet Service Level Agreements (SLAs)**
- **Virtualization increasingly common**



Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Enterprise data centers are composed of large number of clusters of servers. So in a typical example you can see here 2 pictures, which shows the racks full of the servers and it comprises a large cluster and this all is packed in a room and this is called data center. This data center has along with the servers the storage devices are also there along with them and sufficient network devices are there to connect with the outside world these particular data centers are used to run many applications and many applications are running on a particular server or a particular application is running on a spanning multiple servers also.

(Refer Slide Time: 02:48)

Benefits of Virtualization

- **Run multiple applications on one server** *(multiplexing servers to host applications)*
 - Each application runs in its own virtual machine
- **Maintains isolation** *(Co-located Application - isolation)*
 - Provides security
- **Rapidly adjust resource allocations**
 - CPU priority, memory allocation
- **VM migration**
 - "Transparent" to application
 - No downtime, but incurs overhead

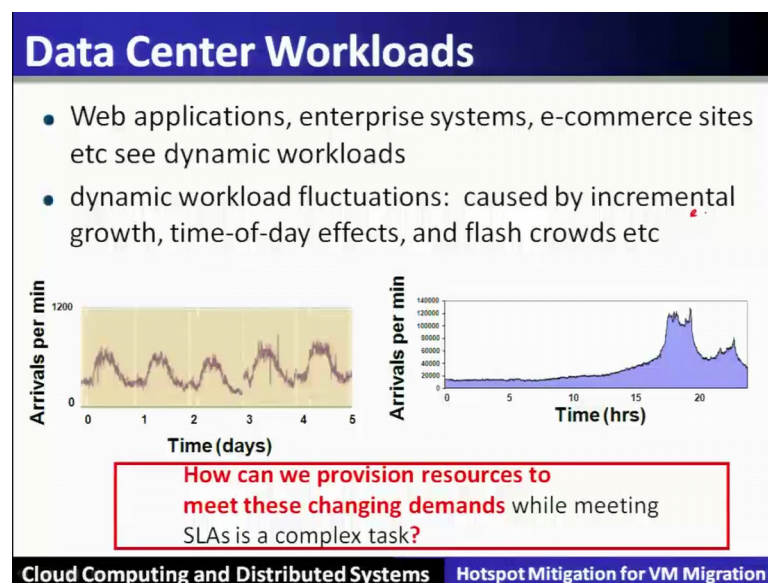
How can we use virtualization to more efficiently utilize data center resources?

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, this will provide a shared hosting environment for the applications to run in this environment, this becomes a multi tier and may span over multiple servers those applications. Now another important thing is the allocation of resources to these applications has to meet to the service level agreements, which is a contract between the customer and application provider and that application in turn will require these resources from the data center to be allocated to the application.

Hence the allocation of the resources has to meet the service level agreements and in this environment of sharing the data center of resources across many applications is done only with the help of the technique which is called virtualization. Let us see the benefits of virtualization, it will enable to run multiple applications on one server; that means, the server will be multiplexed across different applications. Second benefit of virtualization is that it will maintain the isolation across different application, which together in a particular machine. So, the CO located applications are provided with insufficient isolation and the security provisions.

(Refer Slide Time: 05:13)



So, that they may not interfere with each other and yet they will be multiplexing and sharing the resources of the work of the physical machines or the servers. Third benefit here is that it will rapidly adjust the resource allocation to the application for example, the application may require additional CPU on a priority basis or it will require more memory to be allocated to the application all these are possible in the scenario of

virtualized data center. Fourth benefit of virtualization is the virtual machine migration that will be transparent to the application.

So, the application will not know about where this particular application is being used for running that particular application and it is all transparent. Virtual machine migration will support the cause of adjusting the resource allocation, if let us say more resources are required and that server which is currently running the application does not have that many number of resources. So, it will be migrated this is called virtual machine migration and in this migration there will not be any downtime for the applications and also it will incur the less overhead this is all possible with the help of virtualization techniques.

So, how can we use virtualization to more efficiently utilize the data center resource becomes a very important task for the researchers to ponder over. Let us see the data center workloads these data centers are very popular to host wide variety of applications, such as web applications enterprise system applications e commerce websites which are running for the customers. So, these applications require the data center to be used and they will generate the dynamic workloads of different workloads. This dynamic workload will fluctuate due to the incremental growth of the workloads and also during the time of the day affects.

For example, there are some peak hours when many customers are accessing these e commerce website or doing the navigation during the office hours may be the peak time. So, the time of the day also affects this dynamic workload fluctuations and also due to the flash crowds, sometimes we will generate lot of workload and this will guarantee or this will generate the dynamic workload fluctuations.

This all fluctuation has to be absorbed by these data center who are running those applications. So, here how can we provision resources to meet these challenging demands of the workload or a dynamic workloads fluctuations, which are basically due to the incremental growth or during the day time of the day effects or during or due to the flash crowds. So, how can we provision these resources? To meet these challenging demands while meeting the service level agreements is not a trivial task.

(Refer Slide Time: 09:12)

Provisioning Methods

- **Hotspots** form if resource demand exceeds provisioned capacity
- **Over-provisioning**
 - Allocate for peak load
 - Wastes resources
 - Not suitable for dynamic workloads
 - Difficult to predict peak resource requirements
- **Dynamic provisioning**
 - Adjust based on workload
 - Often done manually — *agility*
 - Becoming easier with virtualization

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Now, we will see the provisioning methods which will allocate the resources as per the workload or a dynamic workload demand and what are the issues involved we will call it as the provisioning methods. Now if the resource demands exceed the provisioned capacity, then we say that the hotspots will form. Over provisioning of the resources means that the resources are allocated as per their peak load calculations and they have to be ensured as per the service level agreement slash.

So, over provisioning though there is lot of resources and also is not very suitable for dynamic workloads and also very difficult to predict the peak resource requirements. So, here we will see this over provisioning methods, which will basically how we will ensure it according to the service level agreements. Another thing is called dynamic provisions; that means, due to the workload fluctuations a dynamic provisioning also is required to be in place.

So, this will adjust the resource allocation based on the dynamic workload fluctuation, often it is done manually but it lacks with the agility. If it is done through the in a manual way we will understand this way that why manually will lack the agility; therefore, dynamic provisioning has to be done in a automatic manner.

(Refer Slide Time: 10:15)

Problem Statement

How can we automatically (i) monitoring for resource usage,(ii) hotspot detection, and (iii) mitigation i.e. determining a new mapping and initiating the necessary migrations (i.e. **detect and mitigate Hotspots**) in virtualized data centers?

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Now let us understand the problem for this particular discussion, so given this particular scenario how can we automatically monitor for the resource usage and how can we detect the hotspots and then meeting at the hotspots after finding them, that is how to determine the new mapping and initiate the necessary virtual machines migration. Therefore the problem for this particular discussion is to detect and mitigate the hotspots in a virtualized data center environment. So, again we will recall this word hotspot means that if the resource demands exceeds the provision capacity.

(Refer Slide Time: 12:10)

Hotspot Mitigation Problem

- Once a hotspot has been detected and new allocations have been determined for overloaded VMs, the migration manager invokes its hotspot mitigation algorithm.
- This algorithm determines which virtual servers to migrate and where in order to dissipate the hotspot.
- **Determining a new mapping of VMs to physical servers that avoids threshold violations is NP-hard**—the multidimensional bin packing problem can be reduced to this problem, where each physical server is a bin with dimensions corresponding to its resource constraints and each VM is an object that needs to be packed with size equal to its resource requirements.
- Even the problem of determining if a valid packing exists is NP-hard.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, hotspot mitigation problem is that once we have to detect the hotspot that also is not trivial, we will see some of the methods how we can detect the hotspots and having detected the hotspots then we have to involve a new allocation strategy to deal with the overloaded virtual machines and maybe sometimes requires the virtual machine migration that is all contained in the hotspot mitigation algorithms.

So, mitigation of the hotspot algorithms will determine which virtual servers have that many that sufficient resources required by over provisioned virtual machines are required by the virtual machines, therefore it has to be migrated in order to mitigate the hotspots. So, determining a new mapping of virtual machine to the physical machine that avoids the threshold violations, specified as per the service level agreement is an NP hard problem. That means, there exist an NP complete problem that is called multidimensional bin packing problem, which can be reduced to the hotspot mitigation problem that we have just described.

So, if it is reduced that means multiple multidimensional bin packing problem can be reduced to the hotspot mitigation problem, where each server is a bin with the multiple dimension corresponding to the resource constraints and each virtual machine is an object that need to be packed with the equal with the size equal to it is resource requirements. Even the problem of determining if a valid packing of multidimensional bin exist to determinate itself is a hard problem

(Refer Slide Time: 14:35)

Research Challenges

- **Hotspot Mitigation:** automatically detect and mitigate hotspots through virtual machine migration
- **When** to migrate?
- **Where** to move to?
- **How much** of each resource to allocate?



Sandpiper
A migratory bird

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, this will serve the research challenges for this particular discussion we will see some of the intricacies and we will also see what are the solutions available in the literature in this part of the discussion. So, the research challenges for hotspot mitigation is to automatically detect and mitigate hotspot through the virtual machine migration.

To do this we have to decide when to migrate where to migrate and how much of these resources will be needed to allocate after the migration, this all migration problem is now is published in a paper which has proposed the method which is called sandpiper, which is inspired from that bird called sandpiper which is a migratory bird.

(Refer Slide Time: 15:38)

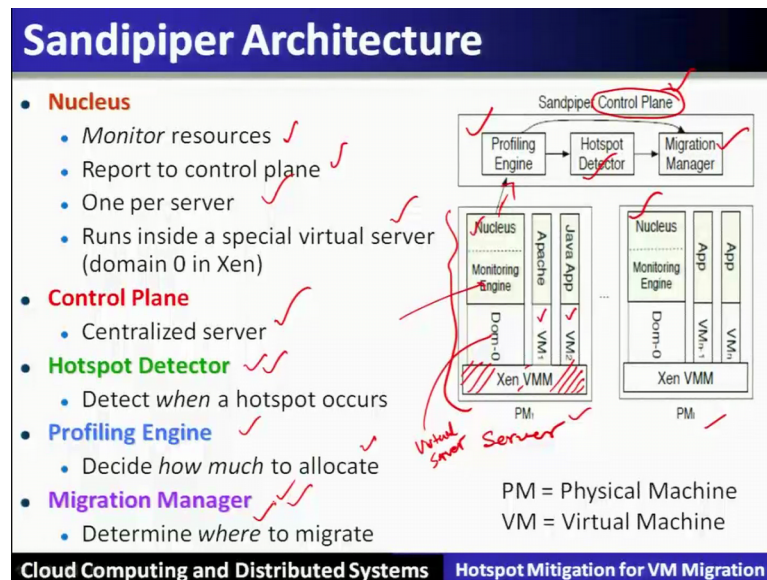
Background

- **Dynamic replication:**
 - Dynamic provisioning approaches are focused on dynamic replication, where the number of servers allocated to an application is varied.
- **Dynamic slicing:**
 - In dynamic slicing, the fraction of a server allocated to an application is varied.
- **Application migration:**
 - In the virtualization, VM migration is performed for dynamic provisioning.
 - Migration is transparent to applications executing within virtual machines.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

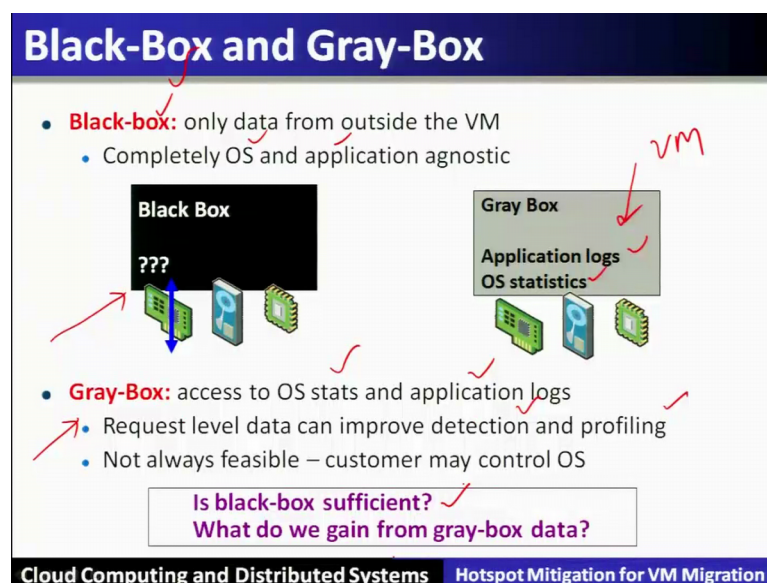
So, how much information is needed to make these particular decisions let us see some of the terminologies as a background. A dynamic replication means that a dynamic provisioning approaches which are focused on dynamic replication where the number of servers which are to be allocated to an application is being varied. A dynamic slicing in a dynamic slicing a fraction of the server is allocated to an application is varied, application migration in the virtualized scenario virtual machine migration is performed for dynamic provisioning and this migration is transparent to the applications executing within the virtual machine.

(Refer Slide Time: 16:37)



Let us understand the architecture of sandpiper scheme for hotspot mitigation problem. The entire architecture can be understood by looking at the physical machines that is nothing but the server, server is we assume that it is virtualized using Xen virtual machine monitor or an hypervisor; therefore, the entire server can run virtual machine on top of it which is shown over here as virtual machine 1 virtual machine 2. Now, there is another virtual machine or that is a control virtual machine which is called a nucleus, so nucleus will monitor the resources of that particular server this will report to the control plane this is the control plane that is a global control plane across the data center.

(Refer Slide Time: 17:44)



This particular nucleus will be per machine per server that we see that there are 2 different servers and every server is running a nucleus, this particular nucleus will run inside a special virtual server. So, this is a special virtual server which is called domain zero in the Xen hypervisor.

So, nucleus is nothing but a monitoring engine of that particular server or a physical machine. Next comes the control plane control plane is a global plane or you can say that it is a global intelligence which comprises of this particular data center. So, this intelligence will have the complete information about all the servers running in that particular data center and this information will be sent by special virtualized server which is called as a nucleus. Now, let us see the details of control plane which is the centralized server, this particular control plane has 3 different components the first 1 is called profiling engine.

This profiling engine will get the profile of all the resources, which are collected by the nucleus and then it will decide using that particular profile of the resources. How much of these resources are available and how much they are allocated the next one is called hotspot detector.

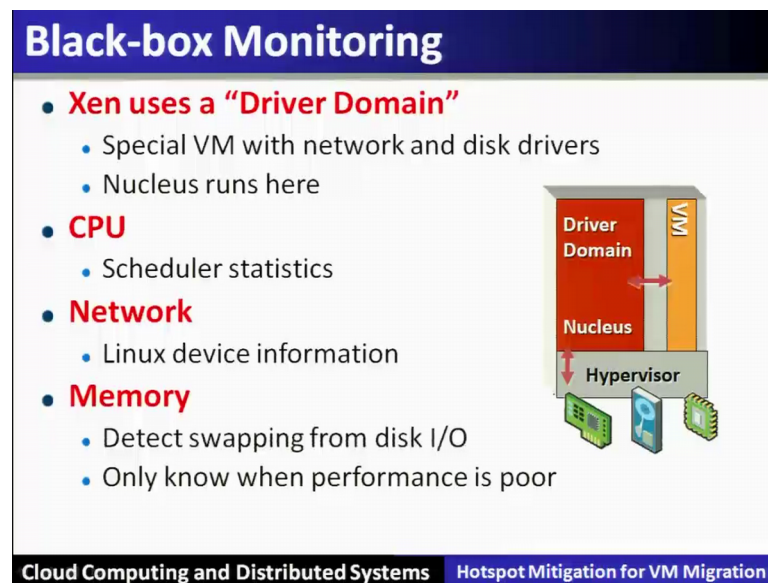
So, it will detect when the hotspots are formed we will see how this particular detection of hotspots are done in this part of the discussion, but let us see in the architecture where these algorithms will work. Finally, there is a manager which is called a migration manager, after detecting the hotspot the migration manager has to decide about how to mitigate that hotspot. Therefore, it has to choose a physical machine where sufficient resources required by the hotspot virtual machine can be made available for the migration plan. This all will be decided by the migration manager where to migrate that particular virtual machine to mitigate the hotspots.

This particular architecture has given also 2 different strategies or a schemes for hotspot mitigation, the first one is called black box method here in the black box method this particular operating system and the application is agnostic; therefore, only it has to collect the data from outside the virtual machine it cannot go and see what is happening or the resource profile within that virtual machine.

So, outside the virtual machine it will collect the data or the profile and therefore it is called a black box. The other method is called the grey box; here it is possible to look

into inside the virtual machine and to collect the information of operating system in statistics and application logs of that virtual machine. This particular information will generate a profile and also detect the request level data which will be required in the detection method.

(Refer Slide Time: 22:14)



Now, the question is that black box good enough to solve all the problem, if not then what do we gain from the grey box method. Let us understand these methods in more detail to answer this particular question. Black box scheme for monitoring of hotspots in this method since it is operating system and application agnostic; therefore, it uses the profile and the information from the outside that is it will depend on the Xen hypervisor.

So, Xen hypervisor will require to have a special virtual machine which will collect the data from network and disk drivers and the nucleus will run based on that particular information, this will be an outside monitoring without looking into the details of what virtual machine is doing.

Therefore hypervisor will collect the scheduler statistics it will collect the network device information statistics it will also collect the memory. But memory will be detected only through the swapping from the disk IOS and this will only be known when the performance is poor why because it is only allowed to collect from the outside information.

(Refer Slide Time: 20:53)

Black-box Monitoring

- **CPU Monitoring:** By incrementing the Xen hypervisor, it is possible to provide domain-0 with access to CPU scheduling events which indicate when a VM is scheduled and when it relinquishes the CPU. These events are tracked to determine the duration for which each virtual machine is scheduled within each measurement interval I .
- **Network Monitoring:** Domain-0 in Xen implements the network interface driver and all other domains access the driver via clean device abstractions. Xen uses a **virtual firewall-router (VFR) interface**; each domain attaches one or more virtual interfaces to the VFR. Doing so enables Xen to multiplex all its virtual interfaces onto the underlying physical network interface.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, black box monitoring will perform the CPU monitoring that is why incrementing the Xen hypervisor it is possible to provide the virtual machine that is domain 0, a special control virtual machine with access to the CPU scheduling events which is available with the hypervisor. These events are tracked to determine the duration for which each virtual machine is scheduled and the measurements of that interval when the CPU was allocated or being used by running that particular virtual machine.

Similarly the network monitoring will be done by special control virtual machine that is domain 0, in the Xen hypervisor by collecting the data from the network interface driver for other domain accesses. So, Xen uses the virtual firewall router interface for this particular purpose this particular information will be collected.

(Refer Slide Time: 25:05)

Black-box Monitoring

- **Memory Monitoring:** Black-box monitoring of memory is challenging since Xen allocates a user specified amount of memory to each VM and requires the OS within the VM to manage that memory; as a result, the memory utilization is only known to the OS within each VM.
- It is possible to instrument Xen to observe memory accesses within each VM through the use of shadow page tables, which is used by Xen's migration mechanism to determine which pages are dirtied during migration.
- However, trapping each memory access results in a significant application slowdown and is only enabled during migrations. Thus, memory usage statistics are not directly available and must be inferred.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

(Refer Slide Time: 25:07)

Black-box Monitoring

- Black-box monitoring is useful in scenarios where it is not feasible to **"peek inside"** a VM to gather usage statistics. Hosting environments, for instance, run third-party applications, and in some cases, third-party installed OS distributions.
- Amazon's Elastic Computing Cloud (EC2) service, for instance, provides a **"barebone"** virtual server where customers can load their own OS images.
- While OS instrumentation is not feasible in such environments, there are environments such as corporate data centers where both the hardware infrastructure and the applications are owned by the same entity.
- In such scenarios, it is feasible to gather OS-level statistics as well as application logs, which can potentially enhance the quality of decision making.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

In black box for monitoring purposes, so black box monitoring is useful in the scenarios where it is not visible to peek inside the virtual machine together, the usage statistics that is possible in most of the applications. So, the hosting environments for instance runs the third party application and for example amazons elastic computing cloud easy to service provides a bare bone virtual server where the customer can load their operating system images. So, in such scenarios it is feasible together the operating system level statistics as well as the application logs to enhance the quality, therefore it is possible to do grey box monitoring.

So, grey box monitoring can be supported when feasible using a demon which can access to the operating level statistics and also the logs. So, this particular demon can process the logs of the applications and to derive the statistics such as request rate request drop and service times. These direct monitoring of application level statistics will enable the explicit detection of service level violation.

(Refer Slide Time: 26:30)

Gray-box Monitoring

- **Gray-box monitoring** can be supported, when feasible, using a light-weight monitoring daemon that is installed inside each virtual server.
- In Linux, the monitoring daemon uses the **/proc** interface to gather **OS level statistics of CPU, network, and memory usage**. The memory usage monitoring, in particular, enables proactive detection and mitigation of memory hotspots.
- The monitoring daemon also can process **logs of applications** such as web and database servers to **derive statistics such as request rate, request drops and service times**.
- Direct monitoring of such **application-level statistics enables explicit detection of SLA violations**, in contrast to the black-box approach that uses resource utilizations as a proxy metric for SLA monitoring.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So that means, in a grey box has more access inside the statistics of the virtual machine in contrast to the black box approach.

(Refer Slide Time: 26:40)

What is a Hotspot ?

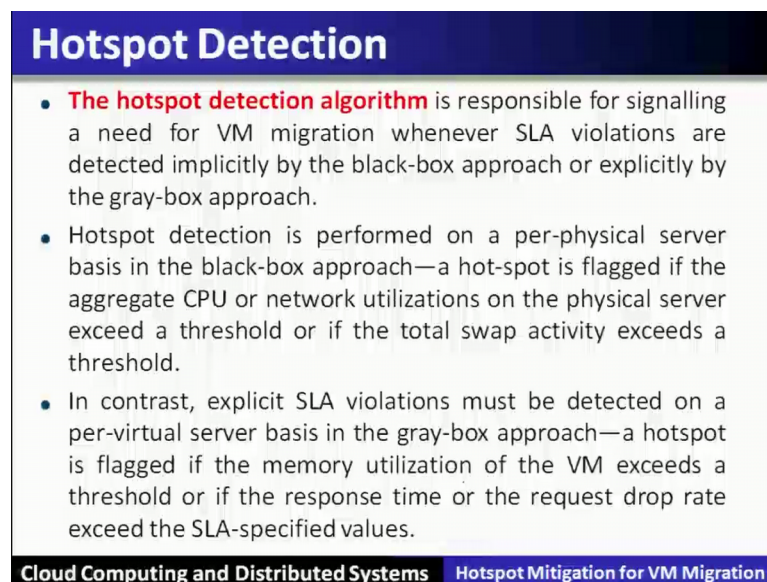
- **A hotspot** indicates a resource deficit on the underlying physical server to service the collective workloads of resident VMs.
- Before the hotspot can be resolved through migrations, The system must first estimate how much additional resources are needed by the overloaded VMs to fulfill their SLAs; these estimates are then used to locate servers that have sufficient idle resources.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, what is the hotspot? So hotspot indicates a resource deficit on the underlying physical server to service the collective workloads of the resident virtual machine. So, if there is a resource deficit's then it will be called hotspot, why because that current server will not be able to allocate the physical resources to handle the workloads of the resident virtual machine. So, before the hotspots can be resolved through the migration, the service first determine the estimate how much of such additional resources are needed by the overloaded virtual machine to fulfill their service level agreements.

These estimates are then used to locate the servers with sufficient idle resources and then we will initiate the virtual machine migration for hotspot mitigation. So, let us see the hotspot detection how this is done this is very important algorithm.

(Refer Slide Time: 27:53)



Hotspot Detection

- **The hotspot detection algorithm** is responsible for signalling a need for VM migration whenever SLA violations are detected implicitly by the black-box approach or explicitly by the gray-box approach.
- Hotspot detection is performed on a per-physical server basis in the black-box approach—a hot-spot is flagged if the aggregate CPU or network utilizations on the physical server exceed a threshold or if the total swap activity exceeds a threshold.
- In contrast, explicit SLA violations must be detected on a per-virtual server basis in the gray-box approach—a hotspot is flagged if the memory utilization of the VM exceeds a threshold or if the response time or the request drop rate exceed the SLA-specified values.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

The hotspot detection algorithm is responsible for signaling a need for virtual machine migration, when service level agreements are violated this is are detected implicitly by black box approach or explicitly by the grey box approach. The hotspot detection is performed on a per physical basis in a black box approach.

That is a hotspot plan if the aggregate CPU network utilizations exceed the threshold or if the total swap activities exceed the threshold the explicit service level agreement violations must be detected on a per server basis in a grey box approach. A hotspot is flagged; if the memory utilization of virtual machine exceeds the threshold or if the response time or the request drop rate exceed the service level.

(Refer Slide Time: 28:46)

Hotspot Detection

- To ensure that a small transient spike does not trigger needless migrations, a hotspot is flagged only if thresholds or SLAs are exceeded for a sustained time. Given a time-series profile, a hotspot is flagged if at least k out of the n most recent observations as well as the next predicted value exceed a threshold. With this constraint, we can filter out transient spikes and avoid needless migrations.
- The values of k and n can be chosen to make hotspot detection aggressive or conservative. For a given n , small values of k cause aggressive hotspot detection, while large values of k imply a need for more sustained threshold violations and thus a more conservative approach.
- In the extreme, $n = k = 1$ is the most aggressive approach that flags a hotspot as soon as the threshold is exceeded. Finally, the threshold itself also determines how aggressively hotspots are flagged; lower thresholds imply more aggressive migrations at the expense of lower server utilizations, while higher thresholds imply higher utilizations with the risk of potentially higher SLA violations.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Agreement is specified to ensure a small transient spike needlessly trigger the needless migration in the hotspot detection mechanism; therefore, at least k out of n most recent observation as well as the next predicted value exceeding that particular threshold is used. With this constraint we can filter out the transient spikes and avoid the needless migration. So, the value of k and n can be chosen to make the hotspot detection aggressive or conservative. So, in the extreme when n is equal to k is equal to 1 is the most aggressive approach.

(Refer Slide Time: 29:27)

Hotspot Detection

- In addition to requiring k out of n violations, we also require that the next predicted value exceed the threshold.
- The additional requirement ensures that the hotspot is likely to persist in the future based on current observed trends. Also, predictions capture rising trends, while preventing declining ones from triggering a migration.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Let us see the hotspot detection that in addition to requiring k out of violations, we also require the next predicted value exceeds the threshold. The additional requirement ensures that the hotspot is likely to persist in the future based on the current observed trend also the predictions captures the rising trends while preventing the declining ones from triggering a migration.

(Refer Slide Time: 29:52)

Hotspot Detection

- Sandpiper employs **time-series prediction techniques to predict future values**. Specifically, the system relies on the auto-regressive family of predictors, where the n -th order predictor $AR(n)$ uses n prior observations in conjunction with other statistics of the time series to make a prediction. To illustrate the first-order $AR(1)$ predictor, consider a sequence of observations: **u_1, u_2, \dots, u_k** . Given this time series, we wish to predict the demand in the **$(k+1)$ th** interval. Then the first-order **$AR(1)$** predictor makes a prediction using the previous value **u_k** , the **mean of the time series values μ** , and the parameter which captures the variations in the time series. The prediction \hat{u}_{k+1} is given by:

$\hat{u}_{k+1} = \mu + \phi(u_k - \mu)$

mean (circled around μ)
Prediction (with arrow pointing to the equation)
- As new observations arrive from the nuclei, the hot spot detector updates its predictions and performs the above checks to flag new hotspots in the system.

Cloud Computing and Distributed Systems
Hotspot Mitigation for VM Migration

Let us understand the more details of hotspot detection; the sandpiper method will employ employs time series prediction techniques. To predict the future values this particular scheme prediction is based on auto regression family of predictors, where n th order predictor uses the prior observations in conjunction with the other statistics of the time series to make the prediction.

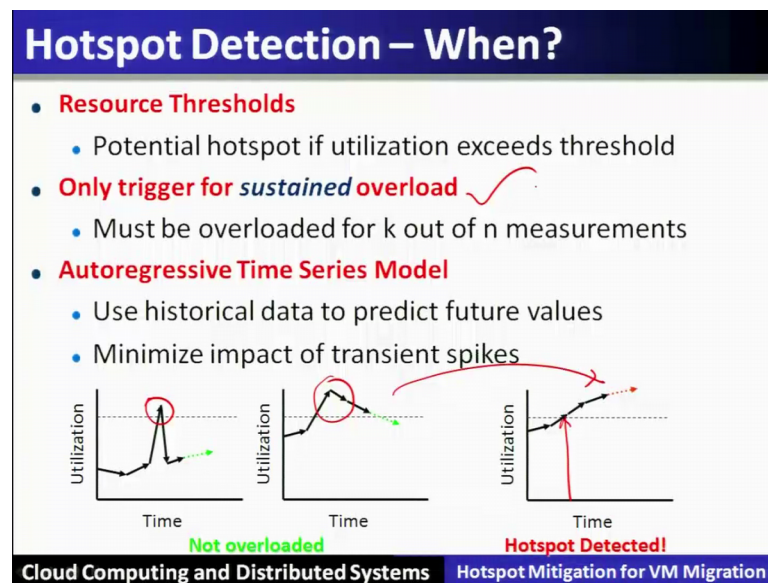
To illustrate this let us understand the first order predictor $AR(1)$ and then consider a sequence of observations let us say that u_1 to u_k . So, given this particular time series auto regressive collection of observations we use to predict the demand in the $k+1$ interval.

Then using the first order prediction the predictor makes a prediction using the previous k values the mean of the time series value μ and the parameter which captures the variation in the time series. The prediction u_{k+1} is given by this particular formulation over here, it depends upon the mean of the time series value μ and also it

will apply the first order prediction on the values on the sequence of observations which are collected over here.

As the new observations arrive from the nucleus the hotspot detector update it is prediction perform the above checks to flag the new hotspot in the system. So, just see that this particular method we will use the predictions based on the mean and also the previous k different observations to predict the k plus 1 th possible value.

(Refer Slide Time: 32:18)



Using this it will detect it will conclude about the hotspot detection. Now when this hot spot detection will be done hotspot detection will be done when this particular resource requirements they will touch the thresholds. So, the potential thresholds if utilization exceeds the threshold then it will be flagged and the hotspot detection will be triggered it is not to be triggered during a transient spikes. Therefore, it will only trigger for sustained overloads and it must be overloaded for k out of n measurements to avoid the outliers of the spikes.

So, auto regression time series model we will see or we have been or it has been used here in hotspot detection, which will be based on the historical data to predict the future values and also minimize in this process the impact of transient spike. So, here in this particular example that if there is a transgender spy this will not trigger detection into the hottest part. But this particular k out of n observation we will is moved this and only

whenever there is a real sustained overload then only it will be triggered as an hotspot detection.

(Refer Slide Time: 33:55)

Resource Provisioning: Black-box Provisioning

- The provisioning component needs to estimate the peak CPU, network and memory requirement of each overloaded VM; doing so ensures that the SLAs are not violated even in the presence of peak workloads.
- **Estimating peak CPU and network bandwidth needs:** Distribution profiles are used to estimate the peak CPU and network bandwidth needs of each VM. The tail of the usage distribution represents the peak usage over the recent past and is used as an estimate of future peak needs.
- This is achieved by computing a high percentile (**e.g., the 95th percentile**) of the CPU and network bandwidth distribution as an initial estimate of the peak needs.

Cloud Computing and Distributed Systems

Hotspot Mitigation for VM Migration

So, let us see the resource provisioning in black box method. So, the provisioning components needed to estimate the peak CPU network and memory requirement for each overloaded virtual machine, to ensure that service level agreements are not violated even in the presence of peak workloads. So, estimating the peak CPU network bandwidth will need the distribution profiles to estimate the peak CPU network utilization needs of the virtual machines, that will be the tail of usage distribution which will represent the peak usage what the recent past and is used as an estimate for the future peak needs. Now, this particular estimation can be achieved by computing the high percentile of CPU and the network bandwidth distribution as the initial estimate of the peak needs.

(Refer Slide Time: 34:53)

Limitation of the black-box approach

- **Example:** Consider two virtual machines that are assigned CPU weights of 1:1 resulting in a fair share of 50% each. Assume that VM1 is overloaded and requires 70% of the CPU to meet its peak needs. If VM2 is underloaded and only using 20% of the CPU, then the work conserving Xen scheduler will allocate 70% to VM1.
- In this case, the tail of the observed distribution is a good indicator of VM1's peak need. In contrast, if VM2 is using its entire fair share of 50%, then VM1 will be allocated exactly its fair share. In this case, the peak observed usage will be 50%, an underestimate of the actual peak need. Since the system can detect that the CPU is fully utilized, it will estimate the peak to be $50 + \Delta$.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

To understand this there is an example shown over here, if there are 2 virtual machines that are assigned the CPU weights 1 is to 1 resulting in a fair share of 50 percent each assume that VM 1 is overloaded and required 70 percent. Now, to meet its peak needs whereas virtual machine 2 is under loaded which requires only 20 percent of the CPU, then the work conserving Xen scheduler will allocate seventy percent to the to the virtual machine 1. In this case the tail of the observed distribution is a good indicator of virtual machine one's peak needs, in contrast to the virtual machine 2 is using its entire fair share of 50 percent then virtual machine 1 will be allocated exactly its fair share.

In this case the peak observed usage will only be 50 percent and underestimate of the actual peak needs. Since the system can detect the CPU as fully neutralized it will estimate the peak to be 50 plus delta.

(Refer Slide Time: 36:08)

Resource Provisioning: (i) Black-box Provisioning

- **Estimating peak memory needs:** Xen allows a fixed amount of physical memory to be assigned to each resident VM; this allocation represents a hard upper-bound that can not be exceeded regardless of memory demand and regardless of the memory usage in other VMs.
- Consequently, the techniques for estimating the peak CPU and network usage do not apply to memory. The provisioning component uses observed swap activity to determine if the current memory allocation of the VM should be increased.
- If swap activity exceeds the threshold indicating memory pressure, then the current allocation is deemed insufficient and is increased by a constant amount Δm .

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Similarly it is possible to estimate the peak memory needs, where Xen hypervisor allows fixed amount of physical memory to be assigned to each resident virtual machine this allocation represents a hard upper bound that cannot be exceeded regardless of the memory demand and regardless of the memory usage in other virtual machines. So, therefore, the swap activity exceeds the threshold indicating the memory pressure, when the current allocation is deemed to be insufficient and is increased by the constant amount of memory requirement.

(Refer Slide Time: 36:57)

Resource Provisioning: (ii) Gray-box Provisioning

- Since the **gray-box approach** has access to application level logs, information contained in the logs can be utilized to estimate the peak resource needs of the application.
- Unlike the black-box approach, the peak needs can be estimated even when the resource is fully utilized.
- To estimate peak needs, the peak request arrival rate is first estimated. Since the number of serviced requests as well as the number of dropped requests are typically logged, the incoming request rate is the summation of these two quantities.
- Given the distribution profile of the arrival rate, the peak rate is simply a high percentile of the distribution. Let λ_{peak} denote the estimated peak arrival rate for the application.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Let us see the resource provisioning in the grey box method, since the grey box method has access to the application level logs the information contained the logs can be utilized to estimate the peak resource requirement of the application. To estimate the peak needs, the peak request arrival rate is first estimated. Since, the number of service request as well as the number of dropped request are typically logged the incoming request rate is the summation of these 2 quantities. Given the distribution profile of arrival rate, the peak rate is simply a high percentile of the distribution which is modeled as the G/G/1 queuing system. The behavior of such queuing system can be captured using the formula which is available in the theory of queues.

(Refer Slide Time: 37:50)

Estimating peak CPU needs:

- An application model is necessary to estimate the peak CPU needs. Applications such as web and database servers can be modeled as **G/G/1 queuing systems**. The behavior of such a **G/G/1 queuing system** can be captured using the following queuing theory result:

$$\lambda_{cap} \geq \left[s + \frac{\sigma_a^2 + \sigma_b^2}{2 \cdot (d - s)} \right]^{-1}$$
- where d is the mean response time of requests, s is the mean service time, and λ_{cap} is the request arrival rate. σ_a^2 and σ_b^2 are the variance of inter-arrival time and the variance of service time, respectively. Note that response time includes the full queueing delay, while service time only reflects the time spent actively processing a request.

Cloud Computing and Distributed Systems
Hotspot Mitigation for VM Migration

(Refer Slide Time: 37:52)

Estimating peak CPU needs:

- While the desired response time d is specified by the SLA, the service time s of requests as well as the variance of inter-arrival and service times σ^2_a and σ^2_b can be determined from the server logs. By substituting these values into equation, a lower bound on request rate λ_{cap} that can be serviced by the virtual server is obtained.
- Thus, λ_{cap} represents the current capacity of the VM. To service the estimated peak workload λ_{peak} , the current CPU capacity needs to be scaled by the factor $\lambda_{peak} / \lambda_{cap}$
- Observe that this factor will be greater than 1 if the peak arrival rate exceeds the currently provisioned capacity. Thus, if the VM is currently assigned a CPU weight w , its allocated share needs to be scaled up by the factor $\lambda_{peak} / \lambda_{cap}$ to service the peak workload.

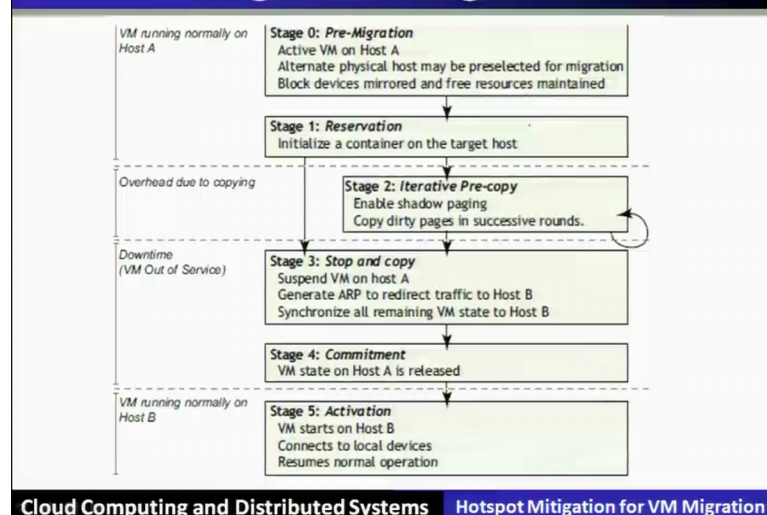
Cloud Computing and Distributed Systems

Hotspot Mitigation for VM Migration

This way the peak CPU can be estimated. Similarly, the peak network also can be estimated in this manner. Now, having estimated all that requirement that is having detected the hotspot.

(Refer Slide Time: 38:09)

Live VM Migration Stages



Cloud Computing and Distributed Systems

Hotspot Mitigation for VM Migration

Let us see the virtual machine migration will take place before that let us see what are the steps in a live virtual machine migration stages. So, stage 0 will say that during the pre migration when the virtual machine running normally on a host A; so that means, it will ensure that which are the active VMs on the host A and what are the alternative alternate physical host maybe preselected for the migration and what are the block devices which are mirrored and free resources which are maintained; this is basically a stage 0 that is

pre migration. The next stage will be stage 1, we will call reservation that is it will in a slice the container in the target host.

Now, the overhead due to the copying that is stage 2; what it will do? It will do iterative precopy, it will enable the shadow paging and then it will keep on copying the dirty pages in the successive rounds. Then stage number 3 is called stop and copy that way it will suspend the VM on the host A and it will generate the network traffic to redirect to the host B over the network and it will synchronize all the remaining virtual machine state to the host B that is called a stop and copy on the host B, stage 4 is called commitment.

So, virtual machine state on host A is released finally in stage 5, this is called activation. So, virtual machine will now start running on host B, it will connect to the local devices and resumes in local operations.

(Refer Slide Time: 40:11)

Live VM Migration Stages

- **Stage 0: Pre-Migration** We begin with an active VM on physical host A. To speed any future migration, a target host may be preselected where the resources required to receive migration will be guaranteed.
- **Stage 1: Reservation** A request is issued to migrate an OS from host A to host B. We initially confirm that the necessary resources are available on B and reserve a VM container of that size. Failure to secure resources here means that the VM simply continues to run on A unaffected.
- **Stage 2: Iterative Pre-Copy** During the first iteration, all pages are transferred from A to B. Subsequent iterations copy only those pages dirtied during the previous transfer phase.

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, these stages we can understand in more details in this particular figure.

(Refer Slide Time: 40:19)

Placement Algorithm

- **First try migrations**
 - Displace VMs from high *Volume* servers
 - Use *Volume*/RAM to minimize overhead
- **Don't create new hotspots!**
 - What if high average load in system?
- **Swap if necessary**
 - Swap a **high Volume** VM for a **low Volume** one
 - Requires 3 migrations
 - Can't support both at once

Swaps increase the number of hotspots we can resolve

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

(Refer Slide Time: 40:32)

Determining Placement – Where to?

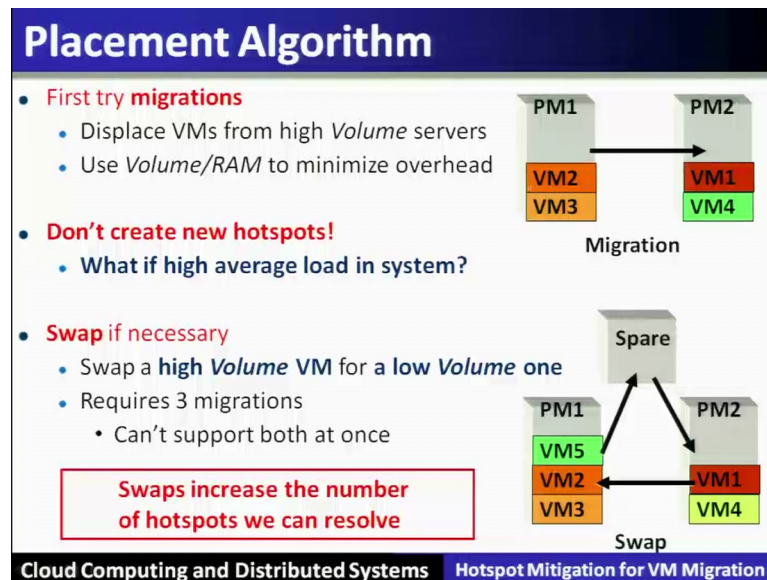
- **Migrate VMs from overloaded to underloaded servers**
- **Use Volume to find most loaded servers**
 - Captures load on multiple resource dimensions
 - Highly loaded servers are targeted first
- **Migrations incur overhead**
 - Migration cost determined by RAM
 - Migrate the VM with highest Volume/RAM ratio

Maximize the amount of load transferred while minimizing the overhead of migrations

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

So, we have to determine the in this particular virtual machine migration the placement where the placement is to be done, finally which has the sufficient number of resources. So here in this particular method, we have to maximize the amount of load which is transferred while minimizing the overhead of migration.

(Refer Slide Time: 40:53)



So, the placement algorithm will first try the migration displays the virtual machine from the high volume servers, use the volume of the RAM to minimize the overhead. It will swap if necessary; swap a high volume virtual machine to a low volume 1 requires 3 different.

(Refer Slide Time: 41:17)

Reading

USENIX NSDI '07

Black-box and Gray-box Strategies for Virtual Machine Migration

Timothy Wood, Prashant Shenoy, Arun Venkataramani, and Mazin Yousif[†]
Univ. of Massachusetts Amherst [†]Intel, Portland

Source: https://www.usenix.org/legacy/event/nsdi07/tech/full_papers/wood/wood.pdf

USENIX NSDI '05

Live Migration of Virtual Machines

Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hansen[†],
Eric Jul[†], Christian Limpach, Ian Pratt, Andrew Warfield
*University of Cambridge Computer Laboratory [†]Department of Computer Science
15 JJ Thomson Avenue, Cambridge, UK University of Copenhagen, Denmark
firstname.lastname@cl.cam.ac.uk {jacobg,eric}@diku.dk*

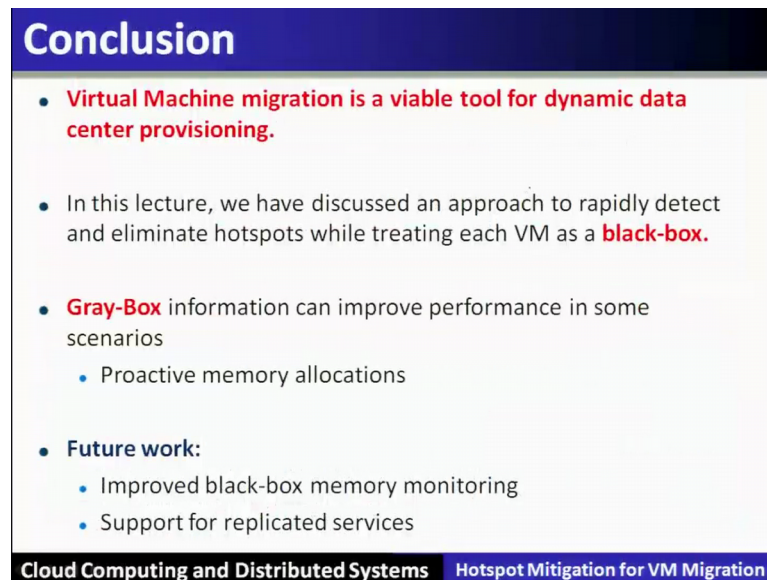
Source: https://www.usenix.org/legacy/event/nsdi05/tech/full_papers/clark/clark.pdf

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

This entire paper is published in the form of black box and grey box strategies for virtual machine migration. This is published by the author Timothy Wood, Prashant Shenoy, Arun Venkataramani and Mazin Yousif of university of Massachusetts Amherst and Intel

Portland, this is published in NSDI in 2007. There is another paper which is referred here is called live virtual machine; live migration of virtual machines, this is by Christopher Clark and is published NSDI in 2005.

(Refer Slide Time: 42:08)



Conclusion

- **Virtual Machine migration is a viable tool for dynamic data center provisioning.**
- In this lecture, we have discussed an approach to rapidly detect and eliminate hotspots while treating each VM as a **black-box**.
- **Gray-Box** information can improve performance in some scenarios
 - Proactive memory allocations
- **Future work:**
 - Improved black-box memory monitoring
 - Support for replicated services

Cloud Computing and Distributed Systems Hotspot Mitigation for VM Migration

Conclusion: Virtual machine migration is a viable tool for dynamic data center provisioning, this lecture we have covered an approach to rapidly detect and eliminate hotspots while treating each virtual machine as a black box. A grey box method wherein we can look inside the virtual machine statistics, grey box information can further be used to improve the performances in some of the scenarios. We can also do the proactive memory allocations in the grey box to make the virtual machines migration more efficient.

There are several future work possible in this particular direction and this is going to be very useful topic in automating the virtualized data center and for building the cloud computing system. The future work and tails as the improved black box memory based monitoring and also support for the replicated services.

Thank you.