Resource Management and Fault Tolerant Principles for Supporting Distributed Real-time and Embedded Systems in the Cloud

Kyoungho An
kyoungho.an@gmail.com

Institute for Software Integrated Systems (ISIS)
Department of Electrical Engineering and Computer Science
Vanderbilt University

December 3, 2012

Outline of Presentation

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results
- Proposed Ideas

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results
- Proposed Ideas
- Concluding Remarks

Overview of DRE Systems

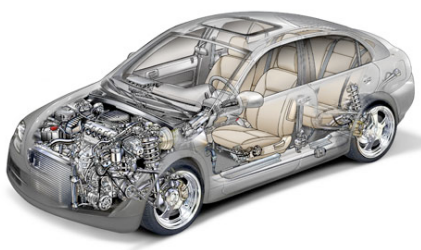
Overview of DRE Systems

- Distributed Real-time Embedded (DRE) system are mission-critical and requires stringent quality of service (QoS)

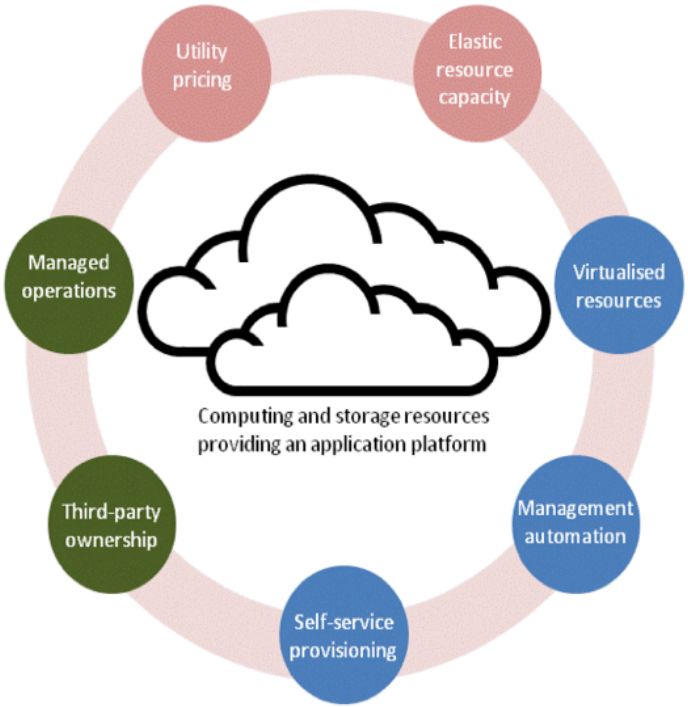
Overview of DRE Systems

- Distributed Real-time Embedded (DRE) system are mission-critical and requires stringent quality of service (QoS)
- In most of DRE systems, the "right answer" delivered too late becomes the "wrong answer"



Overview of Cloud Computing



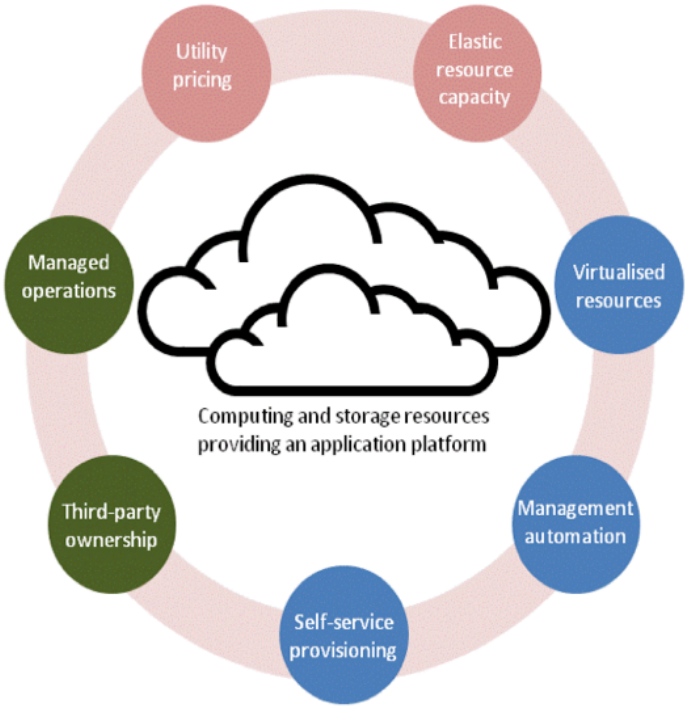
Economic element: Pay-as-you-go, pay-as-you-grow, no capex

Architectural element: Simple, abstract environment for development

Strategic element: Focus on what makes you better, leave the rest to someone else

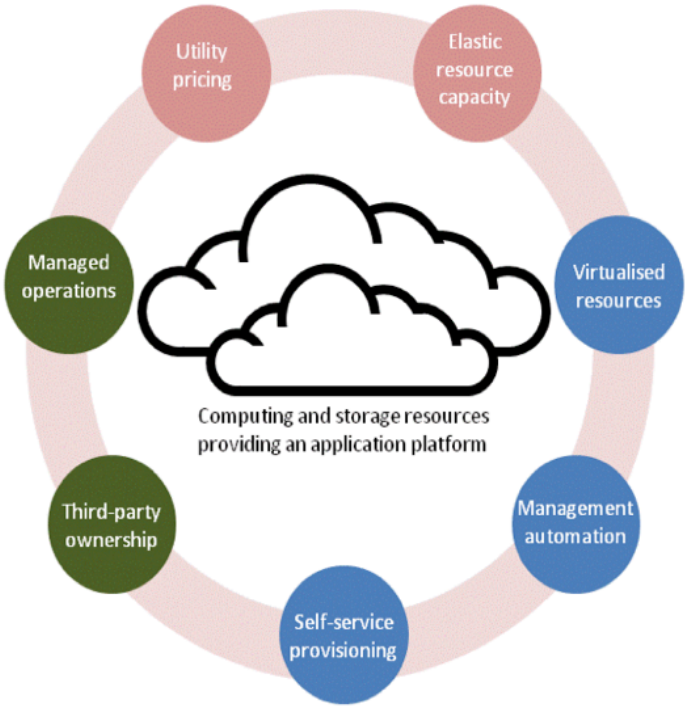
Overview of Cloud Computing

- Cloud computing delivers computing as a service rather than a product



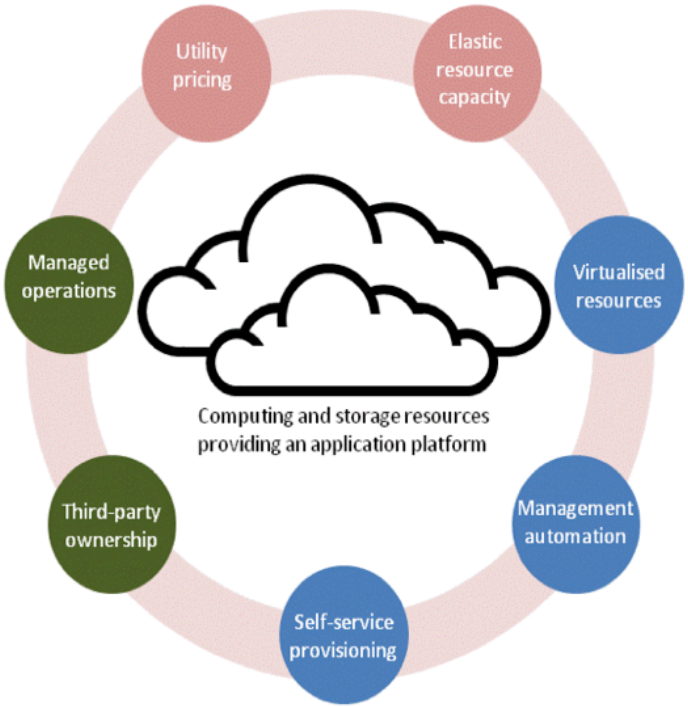
Overview of Cloud Computing

- Cloud computing delivers computing as a service rather than a product
- Cloud computing enables economies of scale via multi-tenancy & elasticity

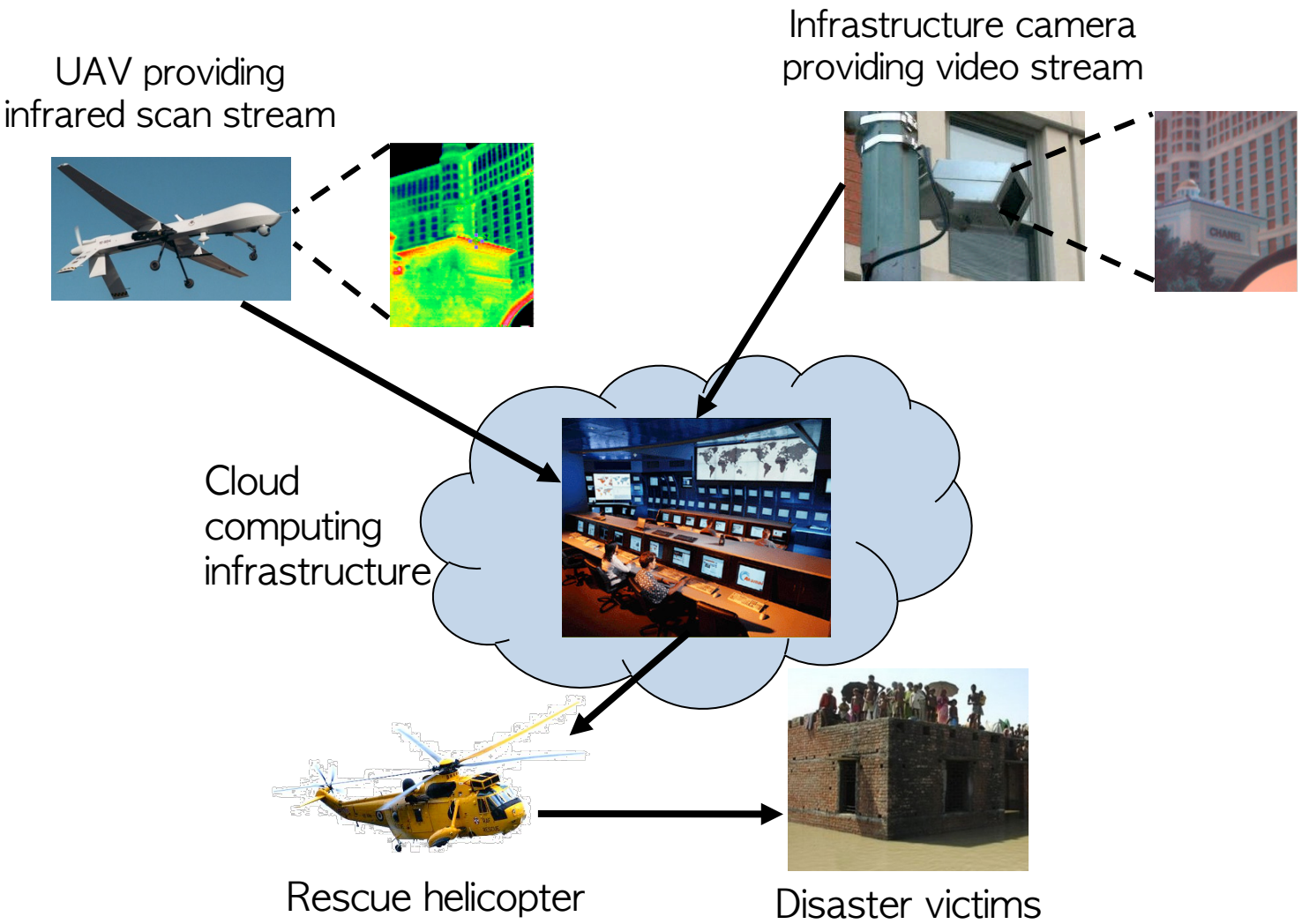


Overview of Cloud Computing

- Cloud computing delivers computing as a service rather than a product
- Cloud computing enables economies of scale via multi-tenancy & elasticity
- Cloud services don't require end-user knowledge of the physical location & configuration of the computing infrastructure delivering the services

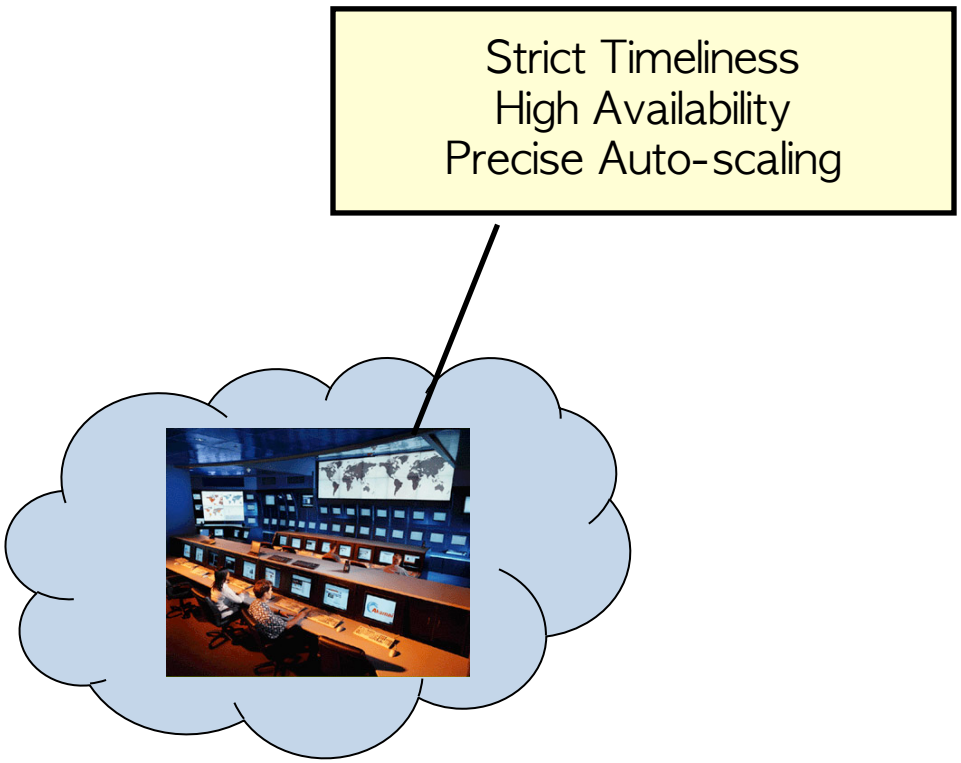


Cloud Computing for DRE Systems



Cloud Computing for DRE Systems and Limitations

Cloud Computing for DRE Systems and Limitations



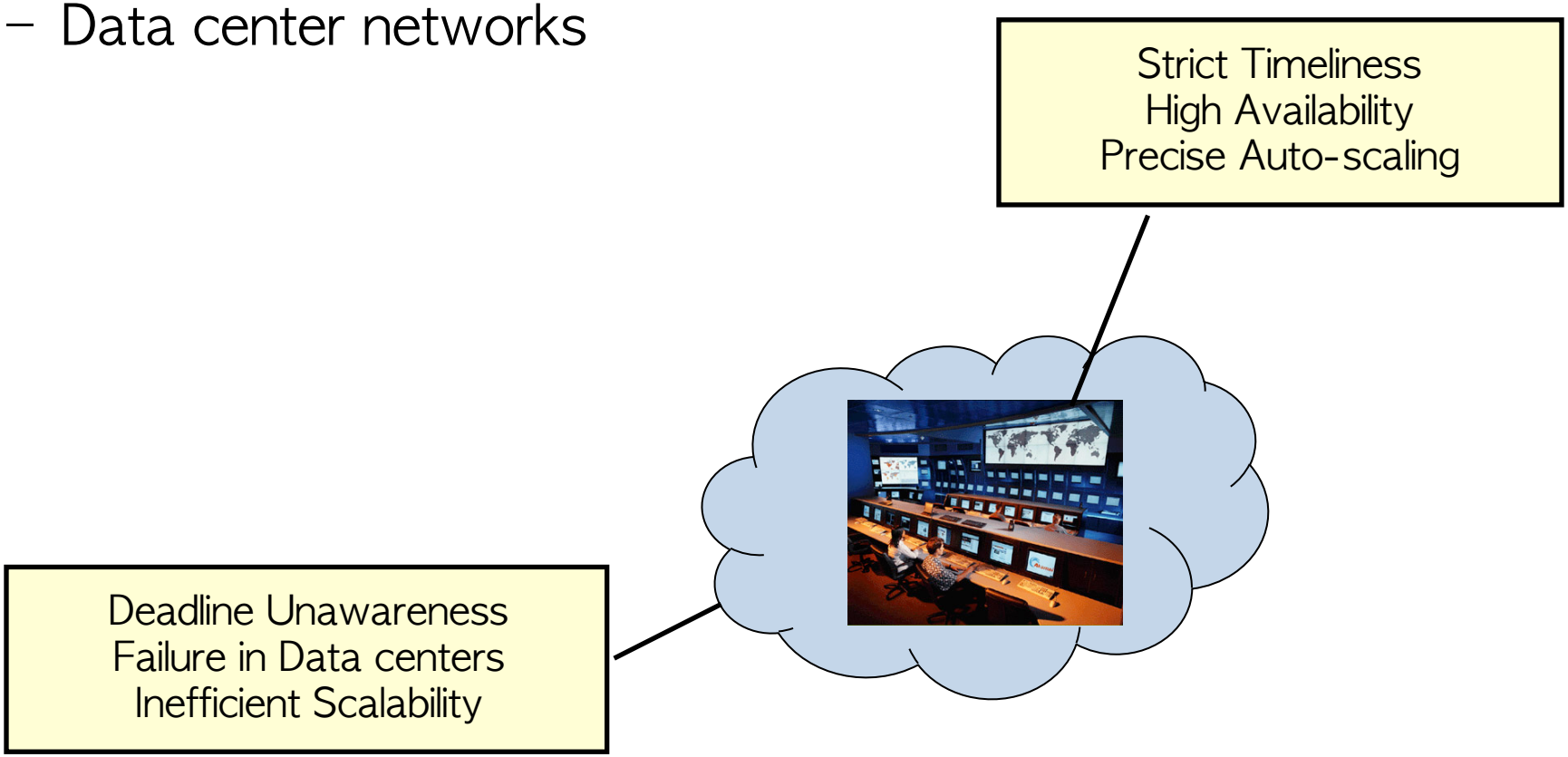
Cloud Computing for DRE Systems and Limitations

Strict Timeliness
High Availability
Precise Auto-scaling

Deadline Unawareness
Failure in Data centers
Inefficient Scalability

Cloud Computing for DRE Systems and Limitations

- Strict timeliness of services
  - Hypervisors
  - Data center networks



Cloud Computing for DRE Systems and Limitations

- Strict timeliness of services
  - Hypervisors
  - Data center networks
- Fault-tolerant mechanisms
  - Failure of hardware and software

Strict Timeliness
High Availability
Precise Auto-scaling

Deadline Unawareness
Failure in Data centers
Inefficient Scalability

Cloud Computing for DRE Systems and Limitations

- Strict timeliness of services
  - Hypervisors
  - Data center networks
- Fault-tolerant mechanisms
  - Failure of hardware and software
- Effective and precise auto-scaling

Strict Timeliness
High Availability
Precise Auto-scaling

Deadline Unawareness
Failure in Data centers
Inefficient Scalability

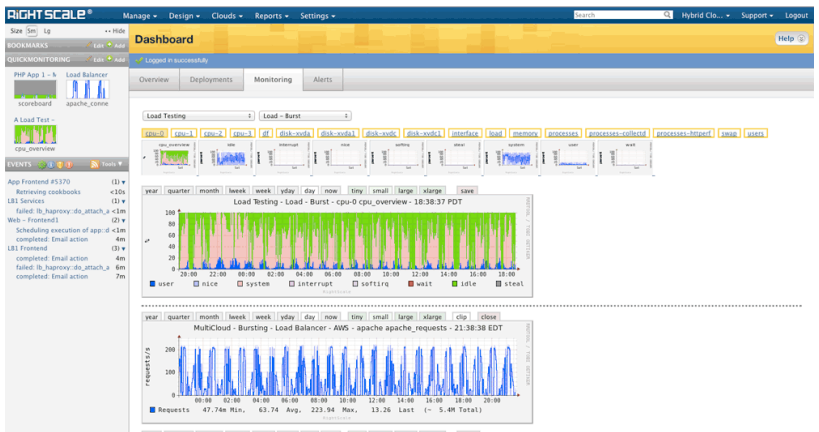
Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results
- Proposed Ideas
- Concluding Remarks

Real-time and Scalable Resource Monitoring

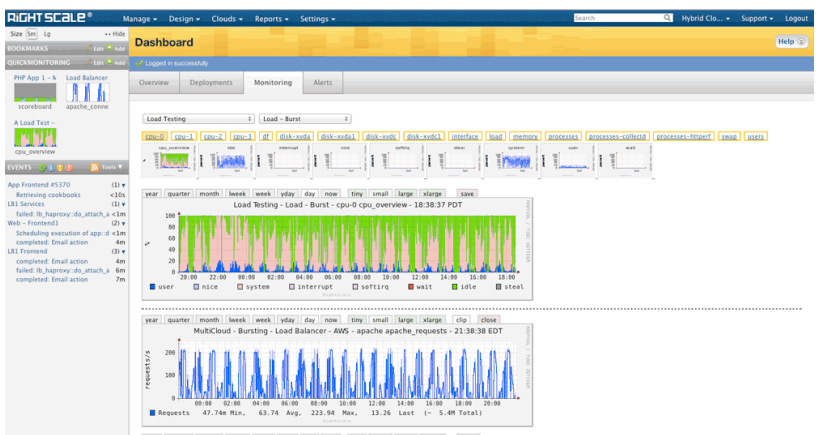
Real-time and Scalable Resource Monitoring

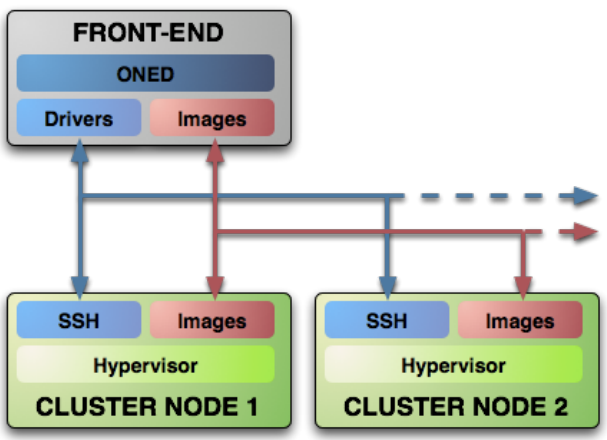
- Context
  - Providing scalable and QoS-enabled monitoring of resources in the cloud is essential
    - To support application QoS properties
    - To identify security threats
  - Existing approaches to resource monitoring in the cloud
    - RESTful APIs, SOAP, AMQP, and XML-RPC
    - Cannot provide real-time information efficiently and scalably



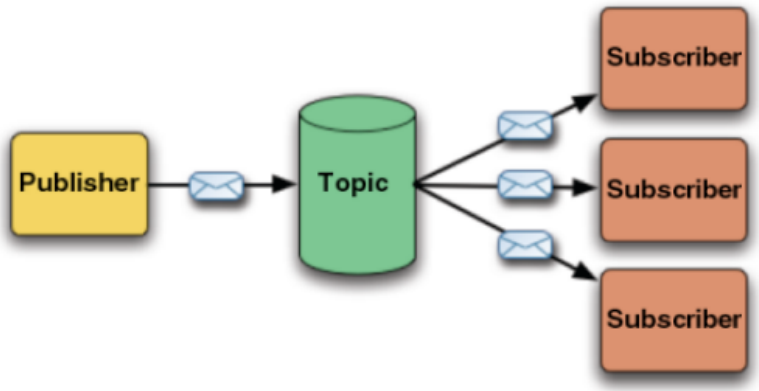
Real-time and Scalable Resource Monitoring

- Context
  - Providing scalable and QoS-enabled monitoring of resources in the cloud is essential
    - To support application QoS properties
    - To identify security threats
  - Existing approaches to resource monitoring in the cloud
    - RESTful APIs, SOAP, AMQP, and XML-RPC
    - Cannot provide real-time information efficiently and scalably



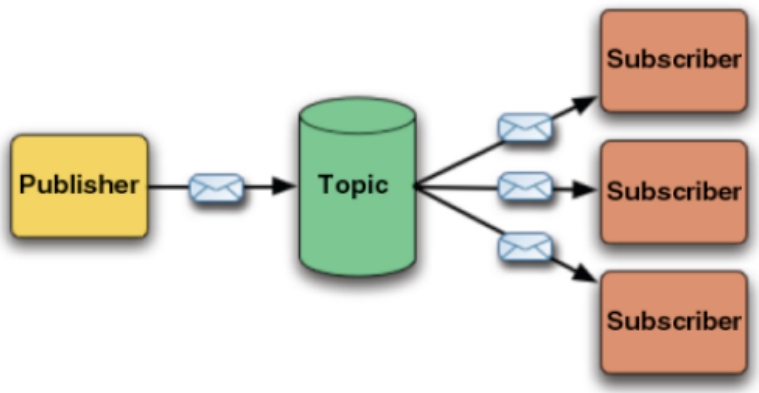


Real-time and Scalable Resource Monitoring



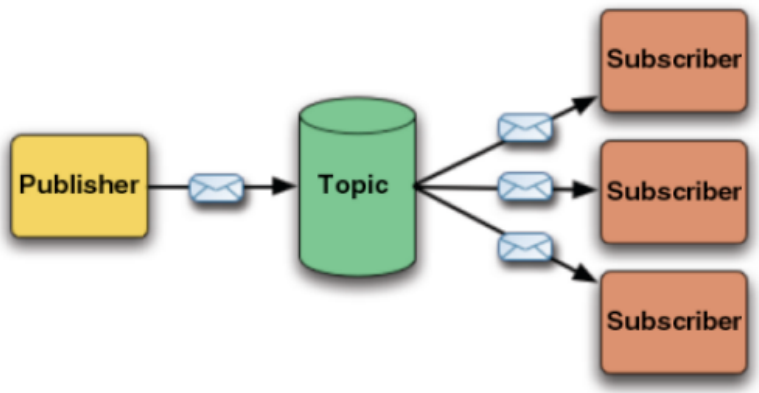
Real-time and Scalable Resource Monitoring

- Related Research
  - Ganglia
  - Nagios
  - Pub/Sub middleware for real-time grid monitoring



Real-time and Scalable Resource Monitoring

- Related Research
  - Ganglia
  - Nagios
  - Pub/Sub middleware for real-time grid monitoring
- Challenges in Prior Work
  - Not for virtualized resources
  - Lack of scalability and support for QoS (timeliness, availability …)



Time-critical Data Center Networks

Time-critical Data Center Networks

- Context

Time-critical Data Center Networks

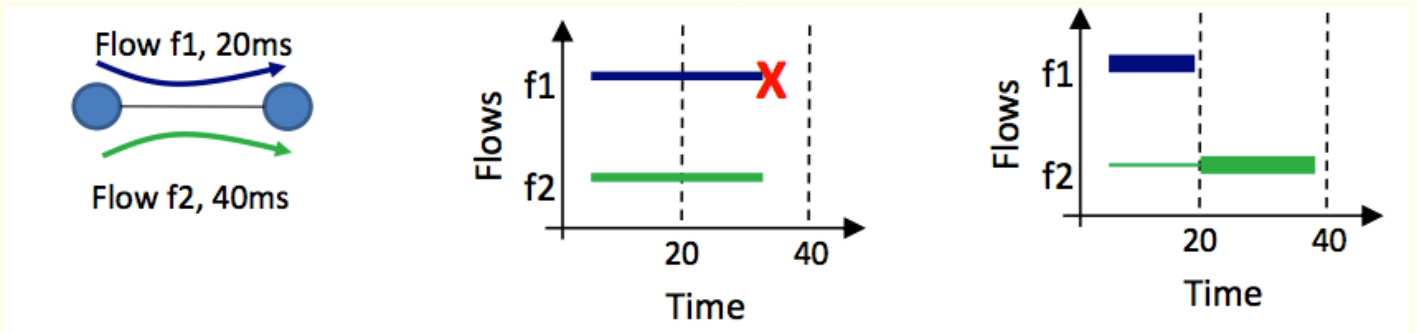
- Context
  - Assuring timeliness of network flows is crucial to complete requested application tasks within expected deadlines

Time-critical Data Center Networks

- Context
  - Assuring timeliness of network flows is crucial to complete requested application tasks within expected deadlines
  - Today's transport protocol in datacenter networks are deadline agnostic and strive for fairness

Time-critical Data Center Networks

- Context
  - Assuring timeliness of network flows is crucial to complete requested application tasks within expected deadlines
  - Today's transport protocol in datacenter networks are deadline agnostic and strive for fairness
  - Case for unfair sharing



Time-critical Data Center Networks

Time-critical Data Center Networks

- Related Research

Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP

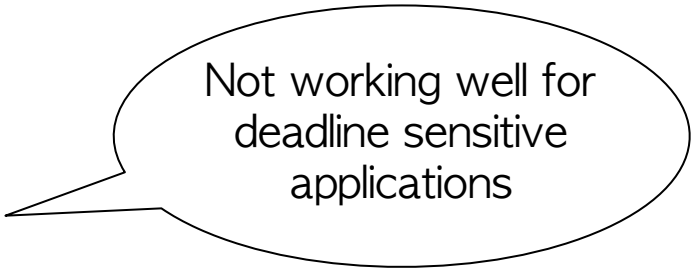
Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP

Not working well for deadline sensitive applications

Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP
  - D^3
    - A deadline-aware control protocol



Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP
  - D^3
    - A deadline-aware control protocol

Not working well for deadline sensitive applications

Priority Inversion
Customized Hardware Needed

Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP
  - D^3
    - A deadline-aware control protocol
  - D^2TCP
    - Reduces deadline miss ratio

Not working well for deadline sensitive applications

Priority Inversion
Customized Hardware Needed

Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP
  - D^3
    - A deadline-aware control protocol
  - D^2TCP
    - Reduces deadline miss ratio
- Challenges in Prior Work
  - The recent research on data center networks has been addressing throughput and deadline issues

Not working well for deadline sensitive applications

Priority Inversion
Customized Hardware Needed

Time-critical Data Center Networks

- Related Research
  - DCTCP (Data Center TCP)
    - Better throughput than TCP
  - D^3
    - A deadline-aware control protocol
  - D^2TCP
    - Reduces deadline miss ratio

Not working well for deadline sensitive applications

Priority Inversion
Customized Hardware Needed

- Challenges in Prior Work
  - The recent research on data center networks has been addressing throughput and deadline issues
  - However… as cloud data centers employ virtualization technology, virtualized network I/O resources in a single physical machine need to be scheduled properly

Real-time Scheduling in Hypervisors

Real-time Scheduling in Hypervisors

- Context

Real-time Scheduling in Hypervisors

- Context
  - Resource virtualization is a key challenge

Real-time Scheduling in Hypervisors

- Context
  - Resource virtualization is a key challenge
    - Improves the utilization of resources

Real-time Scheduling in Hypervisors

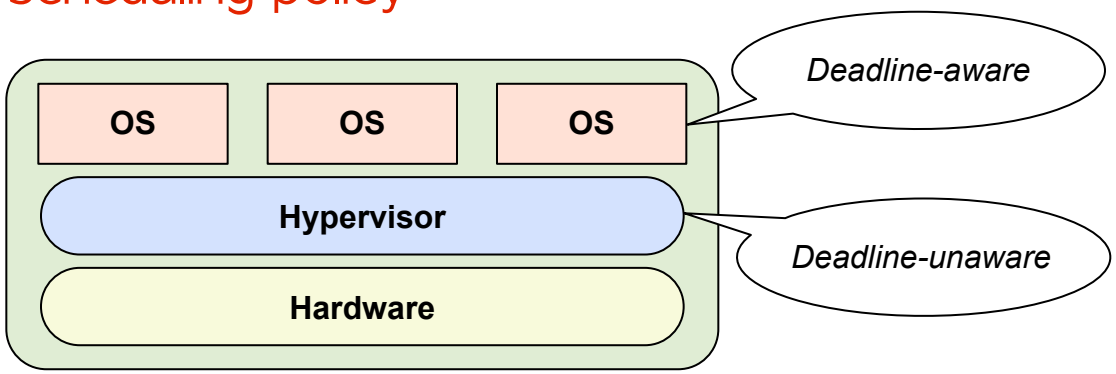
- Context
  - Resource virtualization is a key challenge
    - Improves the utilization of resources
    - Provides isolation among applications

Real-time Scheduling in Hypervisors

- Context
  - Resource virtualization is a key challenge
    - Improves the utilization of resources
    - Provides isolation among applications
  - Virtualization allows physical machines resources to be shared among different virtual machines by using a software layer called a hypervisor or virtual machine monitor (VMM)

Real-time Scheduling in Hypervisors

- Context
  - Resource virtualization is a key challenge
    - Improves the utilization of resources
    - Provides isolation among applications
  - Virtualization allows physical machines resources to be shared among different virtual machines by using a software layer called a hypervisor or virtual machine monitor (VMM)
  - As virtual CPUs are scheduled by the hypervisor, completion time of applications in guest domains are dependent on a hypervisor scheduling policy



Real-time Scheduling in Hypervisors

Real-time Scheduling in Hypervisors

- Related Research

Real-time Scheduling in Hypervisors

- Related Research
  - Scheduler S

Real-time Scheduling in Hypervisors

- Related Research
  - Scheduler S
    - Laxity-based scheduler

Real-time Scheduling in Hypervisors

- Related Research
  - Scheduler S
    - Laxity-based scheduler
    - Laxity determines priority of a task

Real-time Scheduling in Hypervisors

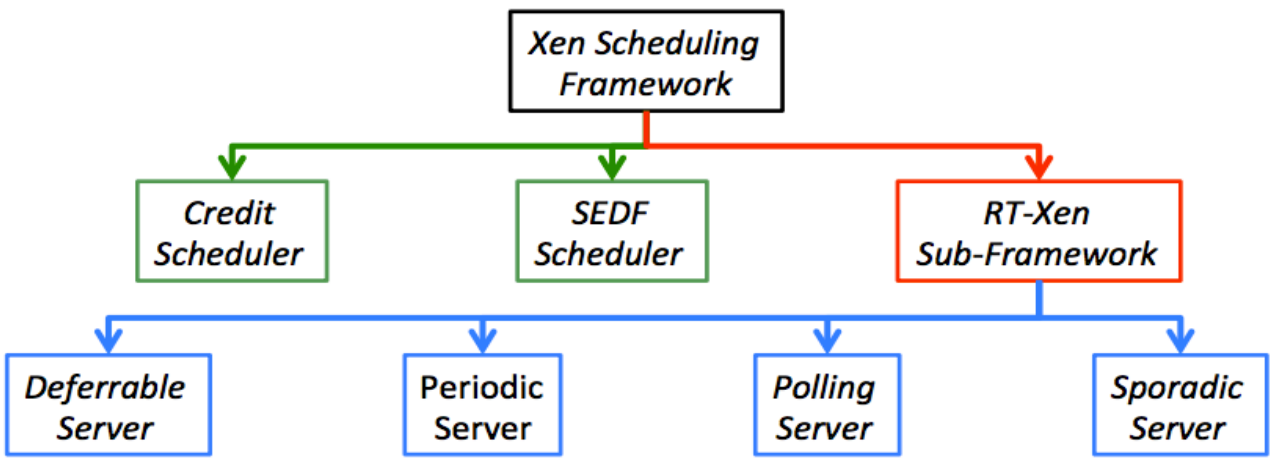
- Related Research
  - Scheduler S
    - Laxity-based scheduler
    - Laxity determines priority of a task
  - RT-Xen

Real-time Scheduling in Hypervisors

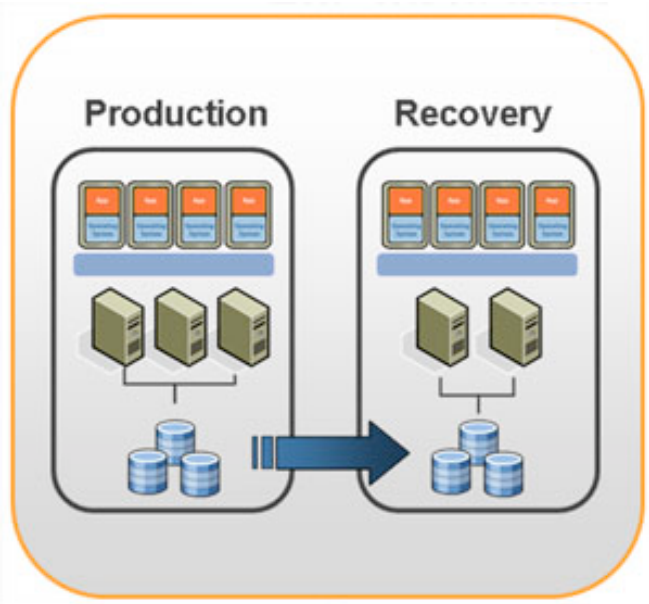
- Related Research
  - Scheduler S
    - Laxity-based scheduler
    - Laxity determines priority of a task
  - RT-Xen
    - Implementation of fixed-priority servers

Real-time Scheduling in Hypervisors

- Related Research
  - Scheduler S
    - Laxity-based scheduler
    - Laxity determines priority of a task
  - RT-Xen
    - Implementation of fixed-priority servers
    - Deferrable server, Polling server, Periodic server, Sporadic server

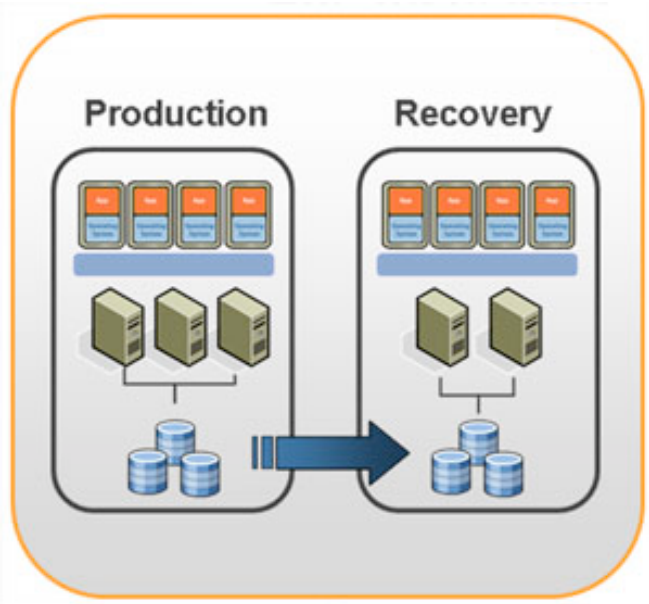


High Availability



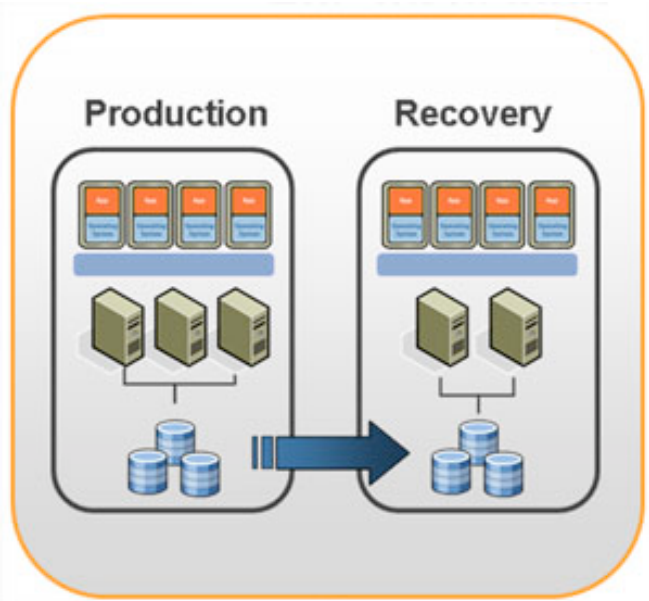
High Availability

- Context



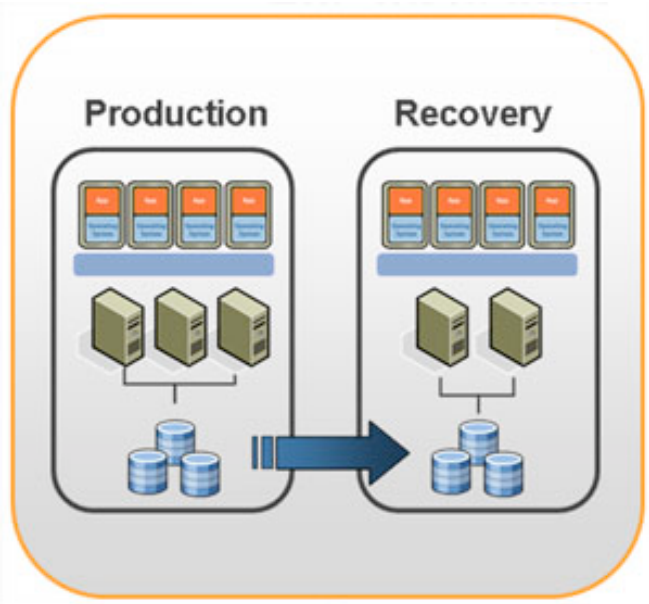
High Availability

- Context
  - Hardware or VM failures occur frequently which requires elegant mechanisms to survive the failure to deliver high availability of services demanded by DRE systems



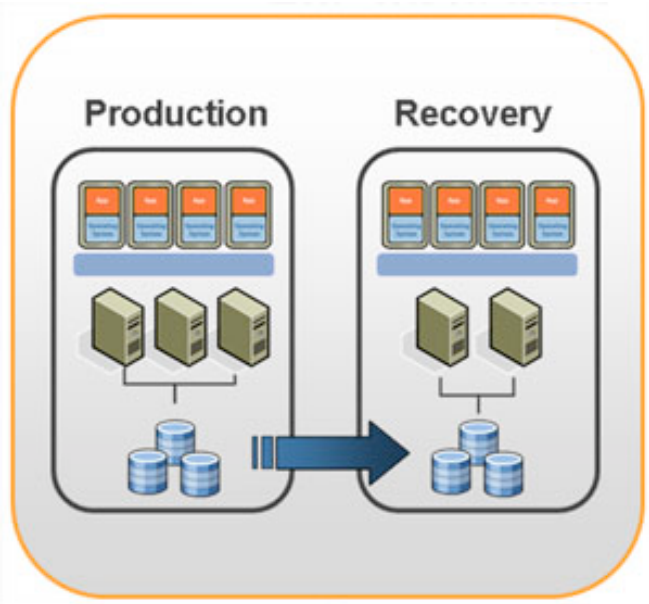
High Availability

- Context
  - Hardware or VM failures occur frequently which requires elegant mechanisms to survive the failure to deliver high availability of services demanded by DRE systems
    - Special-purpose hardware



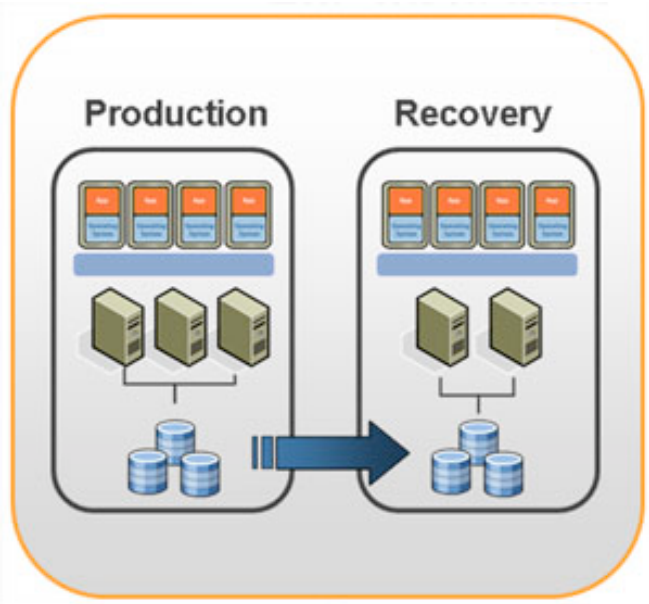
High Availability

- Context
  - Hardware or VM failures occur frequently which requires elegant mechanisms to survive the failure to deliver high availability of services demanded by DRE systems
    - Special-purpose hardware
    - Re-engineering software



High Availability

- Context
  - Hardware or VM failures occur frequently which requires elegant mechanisms to survive the failure to deliver high availability of services demanded by DRE systems
    - Special-purpose hardware
    - Re-engineering software
  - Efficient replicating virtual machines are needed in a general and transparent way



High Availability

High Availability

- Related Research

High Availability

- Related Research
  - Remus

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory
  - Kemari

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory
  - Kemari
    - Lock-stepping and continuous check-pointing

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory
  - Kemari
    - Lock-stepping and continuous check-pointing
  - HydraVM

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory
  - Kemari
    - Lock-stepping and continuous check-pointing
  - HydraVM
    - Storage-based

High Availability

- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory
  - Kemari
    - Lock-stepping and continuous check-pointing
  - HydraVM
    - Storage-based
    - Check-pointing like Remus, but an image of VM is in storage

High Availability

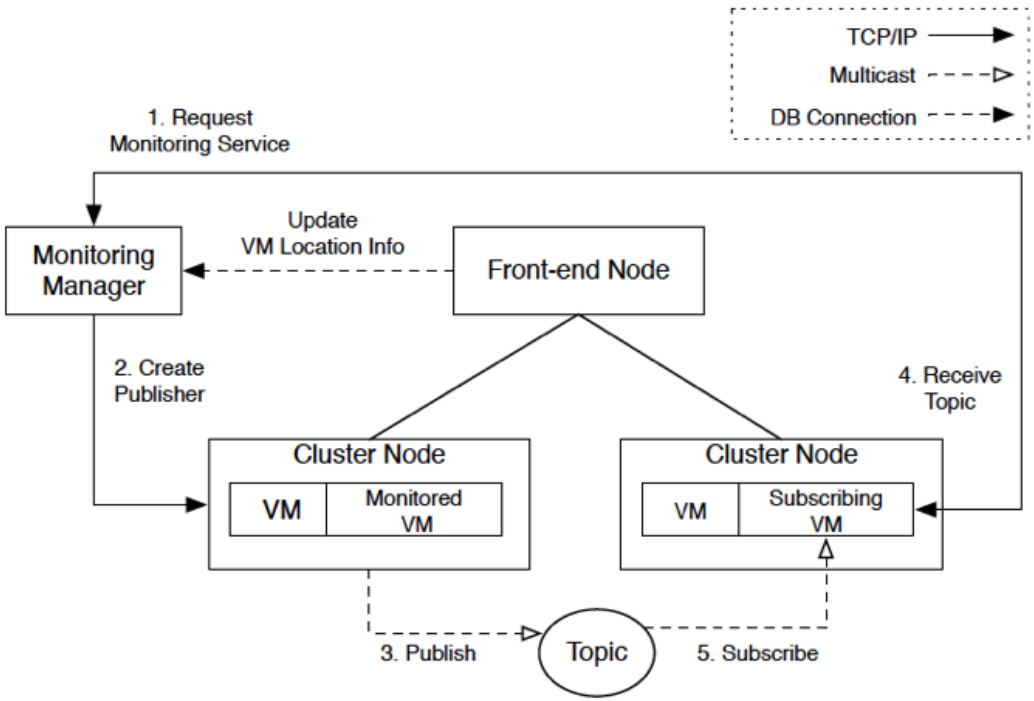
- Related Research
  - Remus
    - Extending the technique to make snapshots used for Xen live migration
    - Frequent checkpoints
    - An image of VM is resident in memory
  - Kemari
    - Lock-stepping and continuous check-pointing
  - HydraVM
    - Storage-based
    - Check-pointing like Remus, but an image of VM is in storage
    - Reduces hardware costs (memory efficient)

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results
- Proposed Ideas
- Concluding Remarks

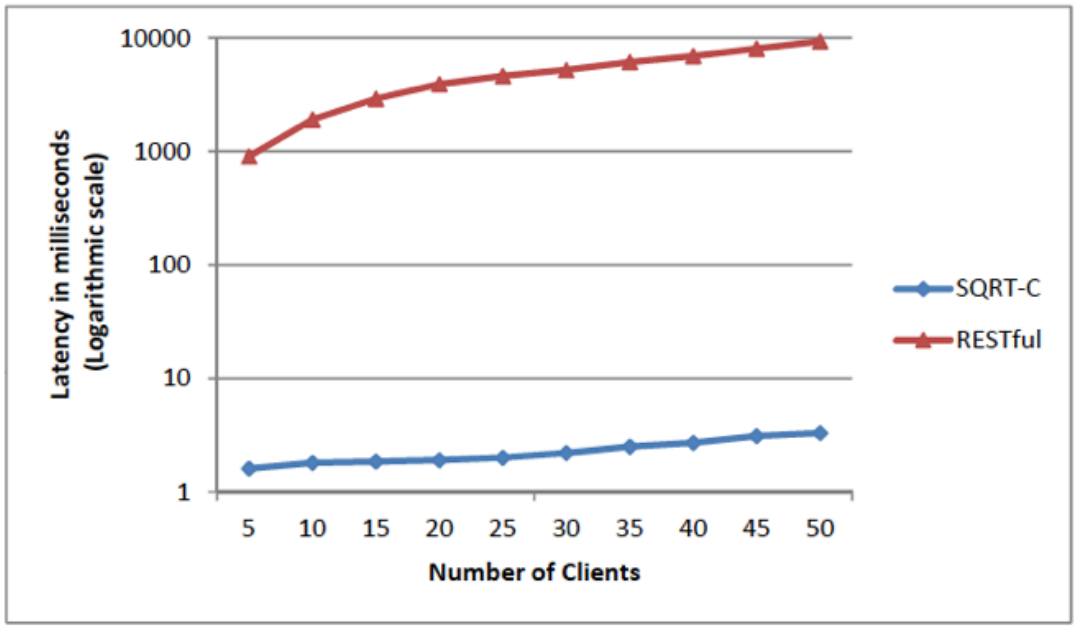
Architecting SQRT-C

- SQRT-C System Architecture



Experimental Results

- Average Message Latency Comparison of SQRT-C and RESTful



Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results
- Proposed Ideas
- Concluding Remarks

Proposed Ideas

Proposed Ideas

- Our survey of related research in QoS support in the cloud has focused on
  - Timeliness in datacenter networks and hypervisors
  - High availability via replications of virtual machines

Proposed Ideas

- Our survey of related research in QoS support in the cloud has focused on
  - Timeliness in datacenter networks and hypervisors
  - High availability via replications of virtual machines
- Research Challenges
  - Tradeoff between timeliness and high availability
  - Data center networks in virtualized environment
  - Lack of fault-tolerant cloud middleware with optimizing resource consumption

Proposed Ideas

- Our survey of related research in QoS support in the cloud has focused on
  - Timeliness in datacenter networks and hypervisors
  - High availability via replications of virtual machines
- Research Challenges
  - Tradeoff between timeliness and high availability
  - Data center networks in virtualized environment
  - Lack of fault-tolerant cloud middleware with optimizing resource consumption
- Research Directions
  - Experimental analysis to identify the possible tradeoffs
  - Design and develop a middleware for fault-tolerance for cloud-based real-time applications

Outline of Presentation

- Context & Terminology
  - Overview of DRE Systems and Cloud Computing
  - Cloud Computing for DRE Systems and Limitations
- Challenges Supporting DRE Systems in the Cloud
  - Real-time and Scalable Resource Monitoring
  - Time-critical Data Center Networks
  - Real-time Scheduling in Hypervisors
  - High Availability and Tunable Adaptive Consistency
- Real-time and Scalable Resource Monitoring
  - Architecture of SQRT-C
  - Experimental Results
- Proposed Ideas
- Concluding Remarks

Concluding Remarks

Concluding Remarks

- Supporting DRE systems in the cloud offers significant benefits such as flexibility, scalability, and cost-effectiveness

Concluding Remarks

- Supporting DRE systems in the cloud offers significant benefits such as flexibility, scalability, and cost-effectiveness
- Existing algorithms and mechanisms are not suitable to host DRE systems

Concluding Remarks

- Supporting DRE systems in the cloud offers significant benefits such as flexibility, scalability, and cost-effectiveness
- Existing algorithms and mechanisms are not suitable to host DRE systems
- We have surveyed the literature that attempt to address these challenges and outlined open challenges for doctoral research

Concluding Remarks

- Supporting DRE systems in the cloud offers significant benefits such as flexibility, scalability, and cost-effectiveness
- Existing algorithms and mechanisms are not suitable to host DRE systems
- We have surveyed the literature that attempt to address these challenges and outlined open challenges for doctoral research
- As the first step toward, scalable and QoS enabled monitoring of resources in the cloud has been conducted