



P O W E R B R I E F

# APM in a Cloud Environment

Maintain performance throughout and after the migration

# Public / Private Cloud (Risk/Reward)

- Cloud deployment and virtualization is a key trend in IT today
- Primarily being driven by financial considerations
  - Reduced hardware costs
  - Reduced management and administrative costs
  - Reduce the footprint of the datacenter
- Application performance is often an afterthought in the decision to migrate to the cloud
  - Decision was made without considering performance
  - Decision was made with assumption that IT will deal with performance
- Migration simultaneously reduces traditional visibility while increasing application complexity
- The migration towards cloud should be accompanied with a comprehensive implementation of application monitoring and troubleshooting capabilities (APM)

# APM in the Cloud



## Virtual Appliances

- Monitor traffic across the WAN and through the cloud
- Monitor traffic between the VM guests
- Measures response time, server delay, throughput, transaction rate....
- Baseline application behavior prior to migration to benchmark performance
- Monitor application performance after migration
- Integrates with VMware and Cisco Nexus 1000v



## Deep Application Instrumentation

- Monitor the application components (Java, .NET, SOA, DB, OS)
- Correlate performance metrics across components
- Code-level visibility to trace transaction through the cloud
- Real-time memory analysis
- Understand physical resource consumption by virtual machines



## Run-Time Application Maps

- Understand application dependencies to plan for migration
- Maintain up to date application maps for cloud deployments



## Deep Transaction Analysis

- Analyze detailed behavior of application transactions
- Predict the performance for alternative deployments (“what if”)

# Use Cases for APM in a Cloud Environment

- Migration to cloud is a gradual evolution
- Each phase of the migration introduces risks and challenges
- A comprehensive APM strategy can mitigate risk and streamline the migration life cycle
  - Map application dependencies
  - Baseline application performance
  - Maintain visibility across physical and virtual
  - Manage application sprawl
  - Manage server mobility
  - Monitor applications in the cloud
- An APM strategy is critical to maintain performance going forward

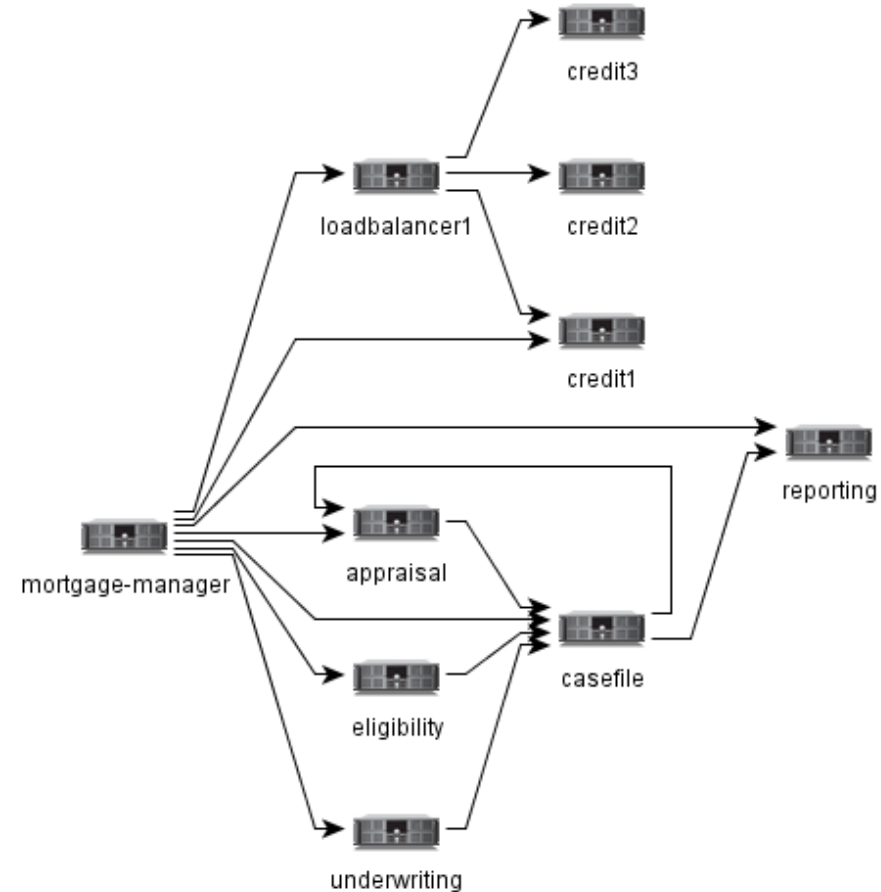
# Use Case 1: Map Application Dependencies

## Challenge

- Complex applications often have unexpected dependencies
- Even application owners may not understand all relationships
- Incomplete understanding of the application makes migration risky

## Solution

- Monitor the real application to understand the true dependencies
- Map the application architecture to develop migration strategy
- Maintain application map before, during, and after migration



***AppMapper Xpert discovers application topology based on live application behavior. It often identifies dependencies that were previously unknown.***

# Use Case 2: Baseline Application Performance

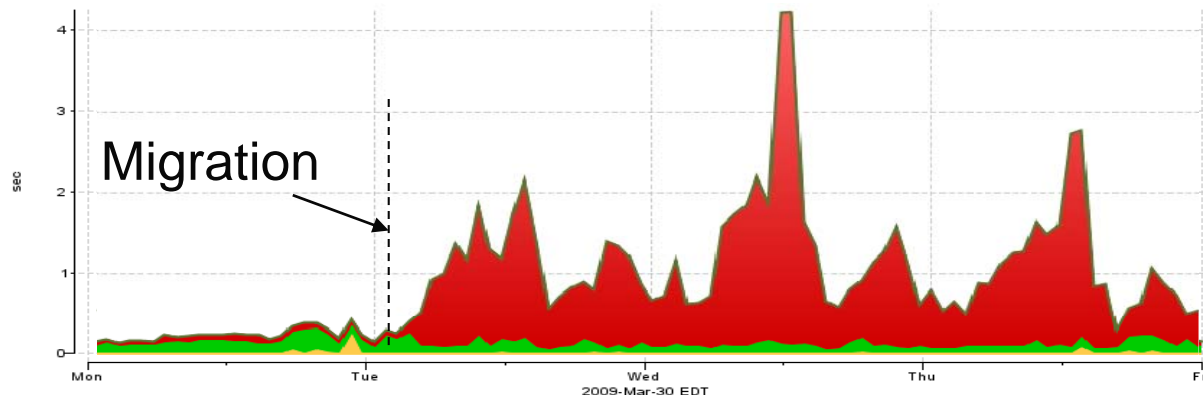
## Challenge

- Virtualization alters the underlying framework of the application
- Real performance of the application can change
  - Change in response time, peak load capacity, traffic patterns...
- Perception of performance may change
  - Users may claim that the virtualization slowed down the application
  - Cloud migration becomes a scapegoat for unrelated performance gripes

## Solution

- Monitor user experience prior to the migration
- Develop a baseline of performance to judge success
  - Objectively identify performance problems
  - Defend the migration if everything went smoothly
- Gauge success of migration on ability to maintain acceptable application performance

***AppResponse Xpert shows that migration in this case has degraded the user performance of this application***



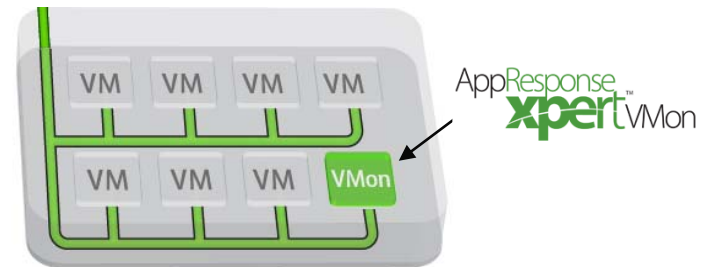
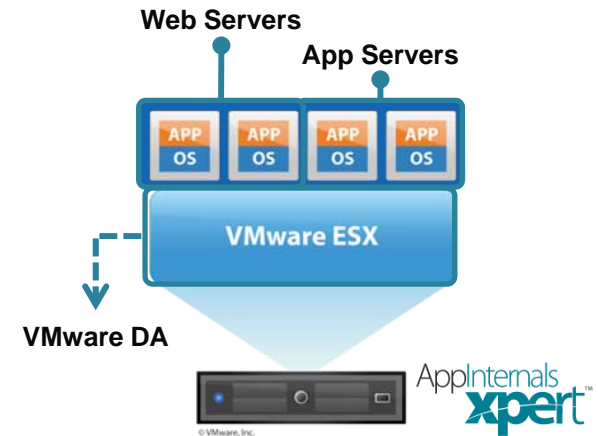
# Use Case 3: Virtual Infrastructure Visibility

## Challenge

- Virtualization has many benefits but reduces visibility into the application
- Traditional server monitoring is confused by contention for CPU, Memory, and I/O on a shared host
- Tiers may communicate within a single VM host hiding traffic from traditional network probes

## Solution

- Instrument the VM hosts with virtual appliances to restore traffic visibility
- Instrument key application components with code-level visibility
- Instrument components to understand physical resource consumption by virtual machines



***AppResponse Xpert VMon resides in the host and monitors all traffic to/from VMs. AppInternals Xpert instruments specific application components for deep visibility***

# Use Case 4: Manage Application Sprawl

## Challenge

- The move to cloud encourages a proliferation of tiers
- Additionally encourages tier mobility
- Must adapt to collect and analyze data from a cloud environment

## Solution

- Combination of physical and virtual appliances to get broad coverage over application tiers and flows
- Continually map the application flows and dependencies to maintain application model
- Agents on application components to measure and correlate performance of the overall application to behavior of specific tiers
- Track application usage patterns and identify VMs and resources that are underutilized



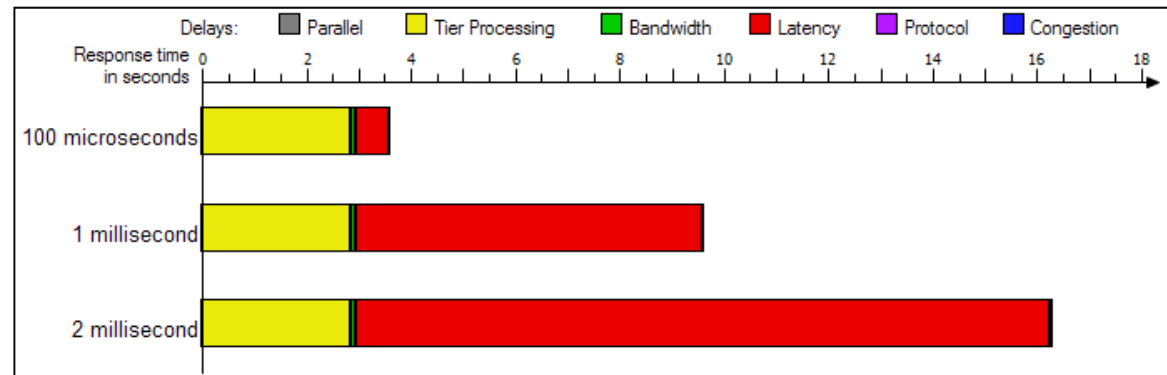
# Use Case 5: Manage Server Mobility

## Challenge

- Cloud encourages mobility
- Some applications are NOT tolerant of increased network latency
- A few additional milliseconds of latency can cripple some applications
- The move a single VM could dramatically affect an application's performance

## Solution

- Profile applications to understand tolerance to tier mobility
- Simulate changes in latency to predict which applications will suffer from tier mobility
- Build policies proactively to identify application tiers that MUST be collocated for performance reasons



***AppTransaction Xpert simulation shows that even small changes in latency have a dramatic impact on application response time***

# Use Case 6: Monitor Applications in the Cloud

## Challenge

- Production applications are in a constant state of flux
  - Network changes, server changes, application changes
- Shared resource and tier mobility due to virtualization result in increased flux

## Solution

- Instrument applications with a broad and deep visibility using OPNET's APM Xpert
- Maintain accurate application dependency maps with AppMapper Xpert
- Monitor end-user experience with AppResponse Xpert
- Monitor performance between VMs with AppResponse Xpert VMon
- Monitor Application components with AppInternals Xpert
  - Resource consumption of physical host and individual VMs
  - Code-level visibility inside Java and .NET components
  - Cross-component behavioral correlation

# Conclusion

- Cloud technology is powerful and revolutionizing IT
- However, the migration to the cloud brings risk and challenges
  - Performance degradation and outages during migration
  - Requirement for performance monitoring and triage after migration
- To maintain continuity and mitigate risks organizations should develop a complete APM strategy
- OPNET's APM Xpert suite, provides tools and methodologies to manage the cloud
  - AppResponse Xpert appliances measure application response times and traffic flows
  - AppInternals Xpert provides deep visibility into key application tiers
    - Correlate performance across the sprawl of application components
    - Map physical to virtual resource utilization
  - AppMapper Xpert creates run-time application maps
    - Understand application dependencies to plan for migration
    - Maintain up to date application maps for cloud deployments
  - AppTransaction Xpert traces individual transactions
    - Deep analysis of application transaction flow
    - Predict changes to underlying conditions like tier mobility