

Application Performance Management for Cloud

CMG

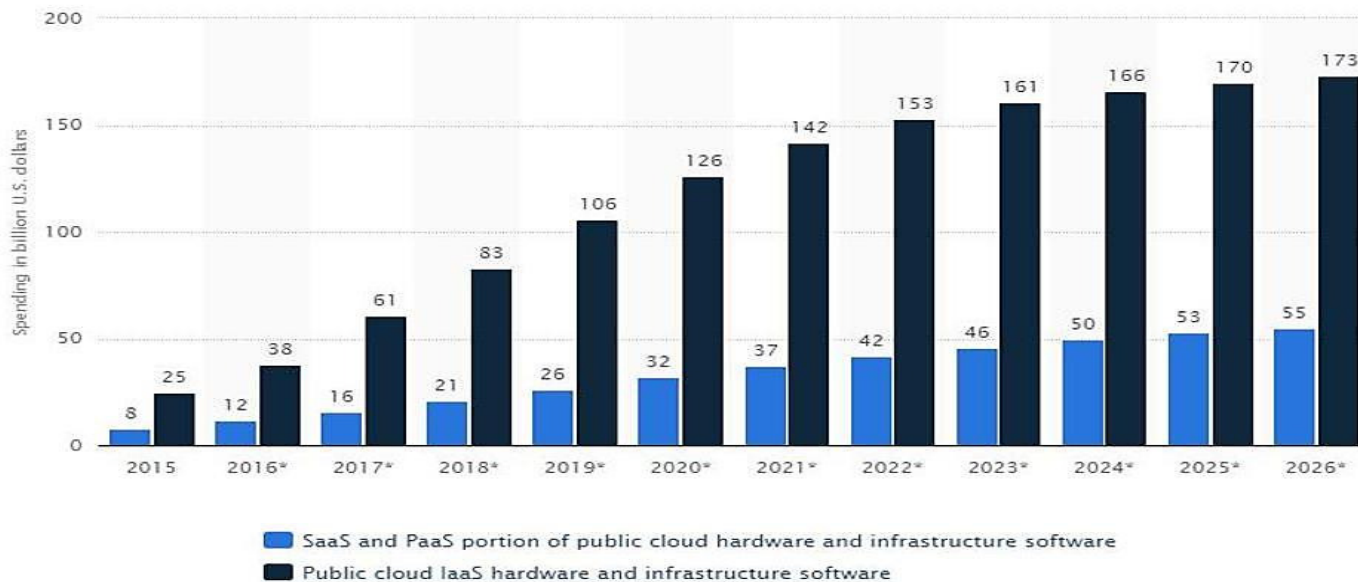


By Priyanka Arora
prarora803@gmail.com

Cloud Adoption Trends

2

Public cloud Infrastructure as a Service (IaaS) hardware and software spending from 2015 to 2026, by segment (in billion U.S. dollars)



- Spending on public cloud Infrastructure as a Service hardware and software is forecast to reach \$173B in 2026
- SaaS and PaaS portion of cloud hardware and infrastructure software spending are projected to reach \$55B in 2026

Agenda

3

- Understanding Cloud
 - ▣ What is Cloud Computing
 - ▣ Cloud Deployment Models
 - Public vs Private, Hybrid?
 - ▣ Cloud Service Models
 - IaaS/PaaS/SaaS
- Problem Statement
 - ▣ Obscurity of Cloud
- Solution
 - ▣ Understanding APM
 - ▣ Considerations for Cloud Applications
- Making the right APM choice
 - ▣ APM for XaaS
 - ▣ Synthetic Vs Real User Monitoring
 - ▣ Magic Quadrant and Market Solutions
 - ▣ Evaluating APM Solutions
- Examples

4

Understanding Cloud

What is cloud computing?

5



Traditional IT

- ❑ Cost of Ownership and Maintenance
- ❑ Cost of Operation
- ❑ Total Control



Cloud Computing

- ❑ Flexible availability
- ❑ Cost effective, Pay per use
- ❑ Unlimited Scalability
- ❑ Fit for Purpose

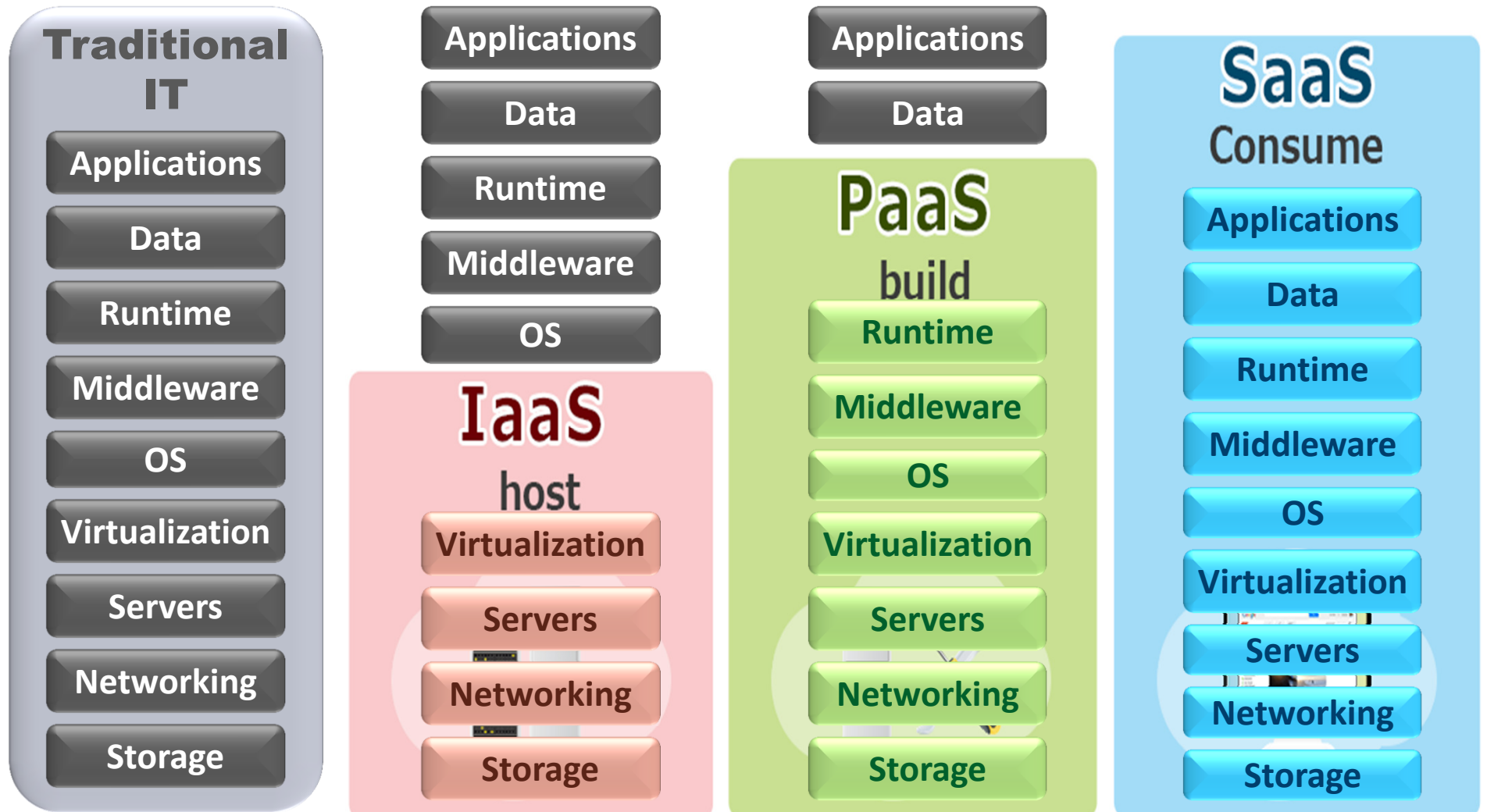
Cloud Deployment Models

6

- **Private Cloud**
 - ▣ Privately shared virtualized resources
 - ▣ Systems and Services operated and delivered exclusively for an organization
- **Public cloud**
 - ▣ Enterprise taps into a public set of resources delivering standardized, highly automated offerings
 - ▣ Compute resources, storage and networking capabilities, are owned by a service provider and offered on demand with limited SLAs regarding tenancy, isolation, and performance.
 - ▣ Shifts CapEx to OpEx – Pay as you go
- **Hybrid cloud**
 - ▣ Implementation of “Hybrid” includes a mix of internal (on-premises, often virtualized) and external (cloud-hosted) solutions, with applications switching between resources as needed
 - ▣ Allows manipulation of CapEx and OpEx

Cloud Service Models

7



8

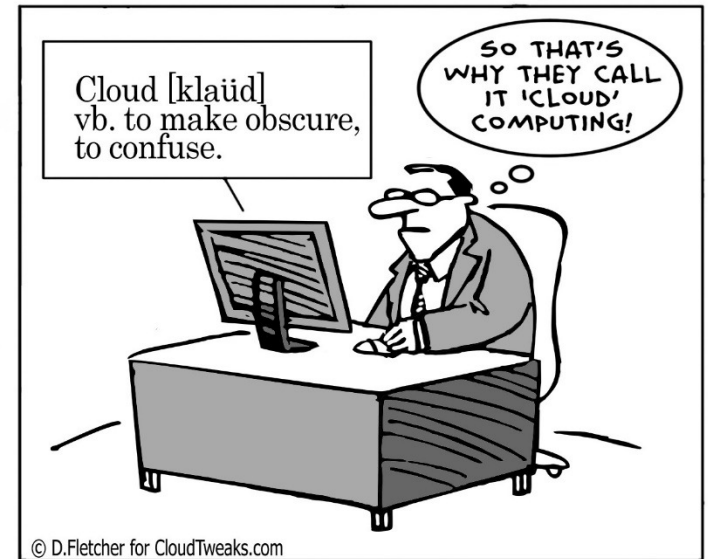
Problem Statement

Obscurity of Cloud

Cloud Challenges

9

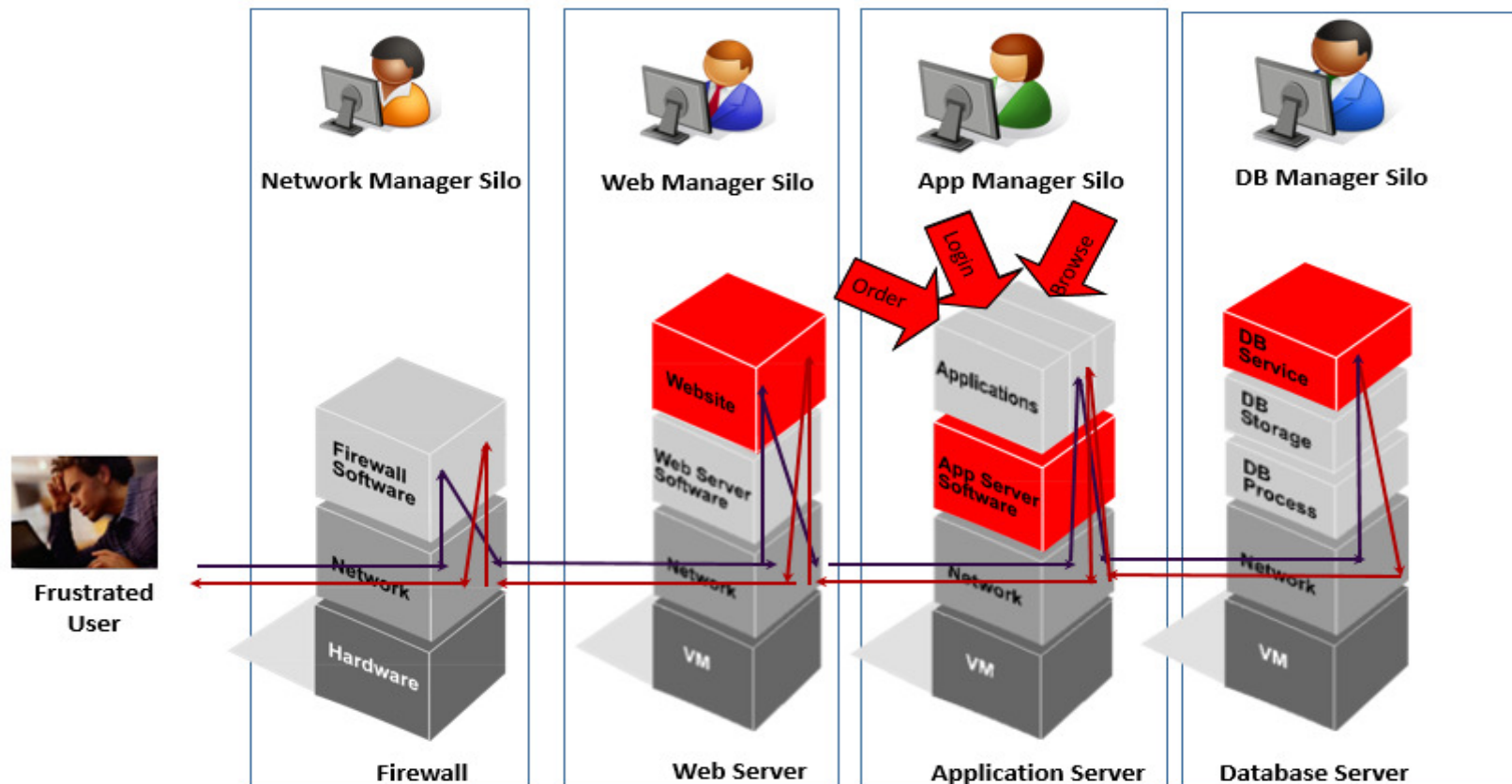
- Security and Privacy
 - ▣ Compliance
- Obscurity
 - ▣ Lack of Visibility
 - ▣ Loss of Control
- Availability, Reliability, Service Quality
 - ▣ Performance Unpredictability
- Lack of Skills, Expertise
- Integration with Existing Infrastructure



Traditional Management Paradigm

10

- Traditional data center management and monitoring protocols focus on technology elements in silos
- Lack of visibility and control puts organization at the mercy of cloud vendor
 - ▣ Vendor's commitment and capability to meet SLAs

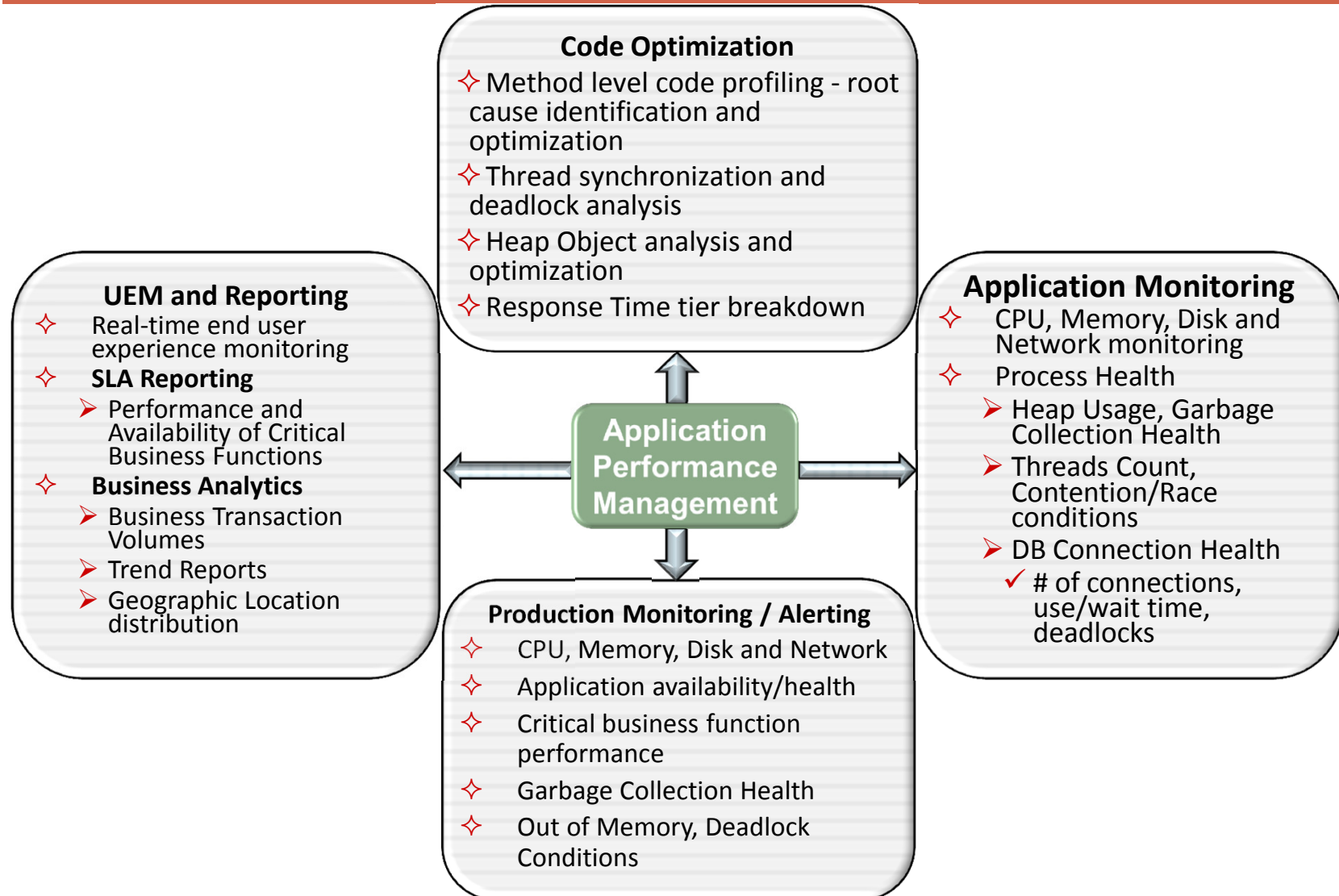


11

Solution - CAPM

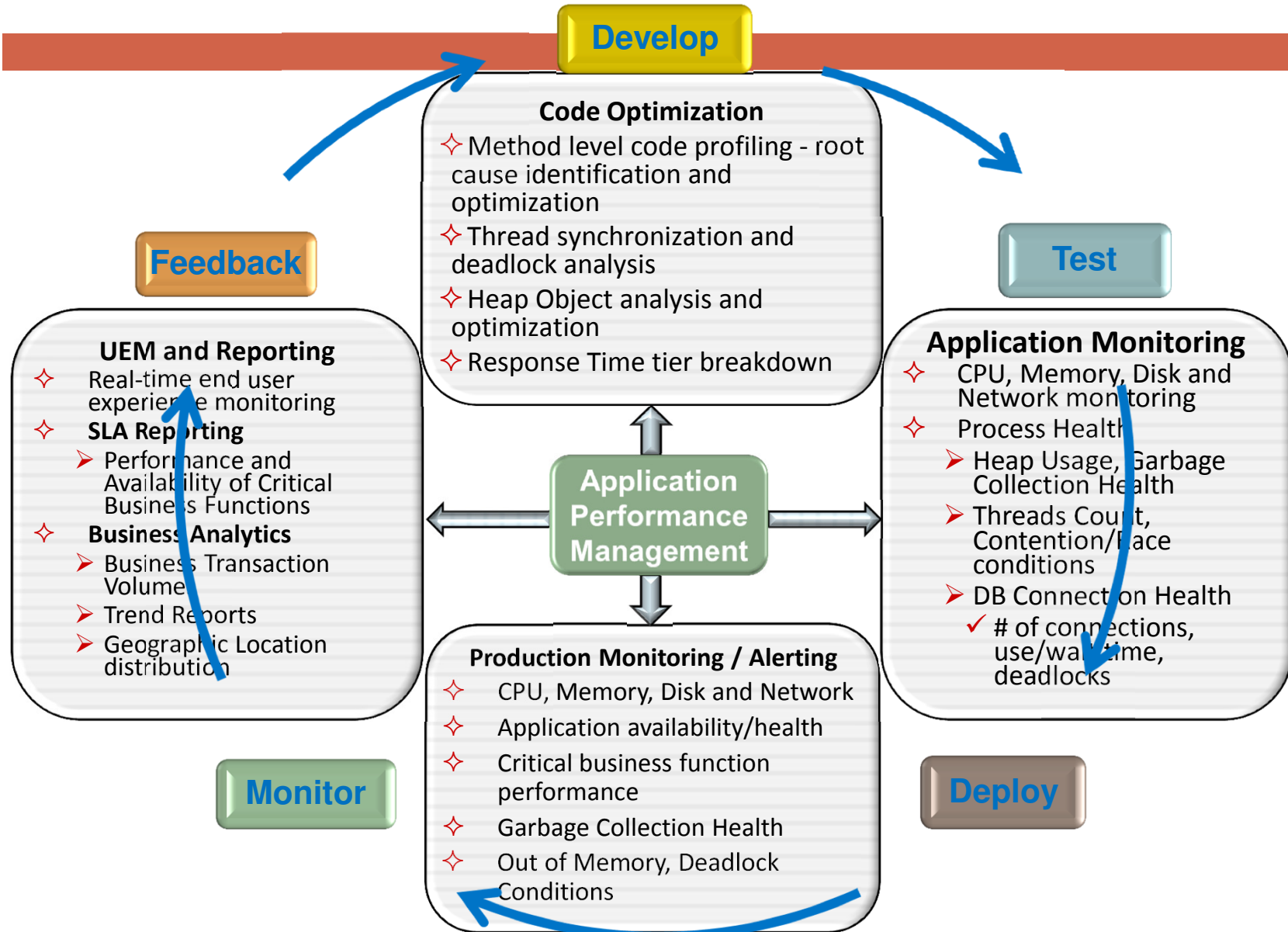
Application Performance Measurement/Monitoring/Management?

12



Application Performance Measurement/Monitoring/Management?

13



APM & Cloud: Why is it Important?

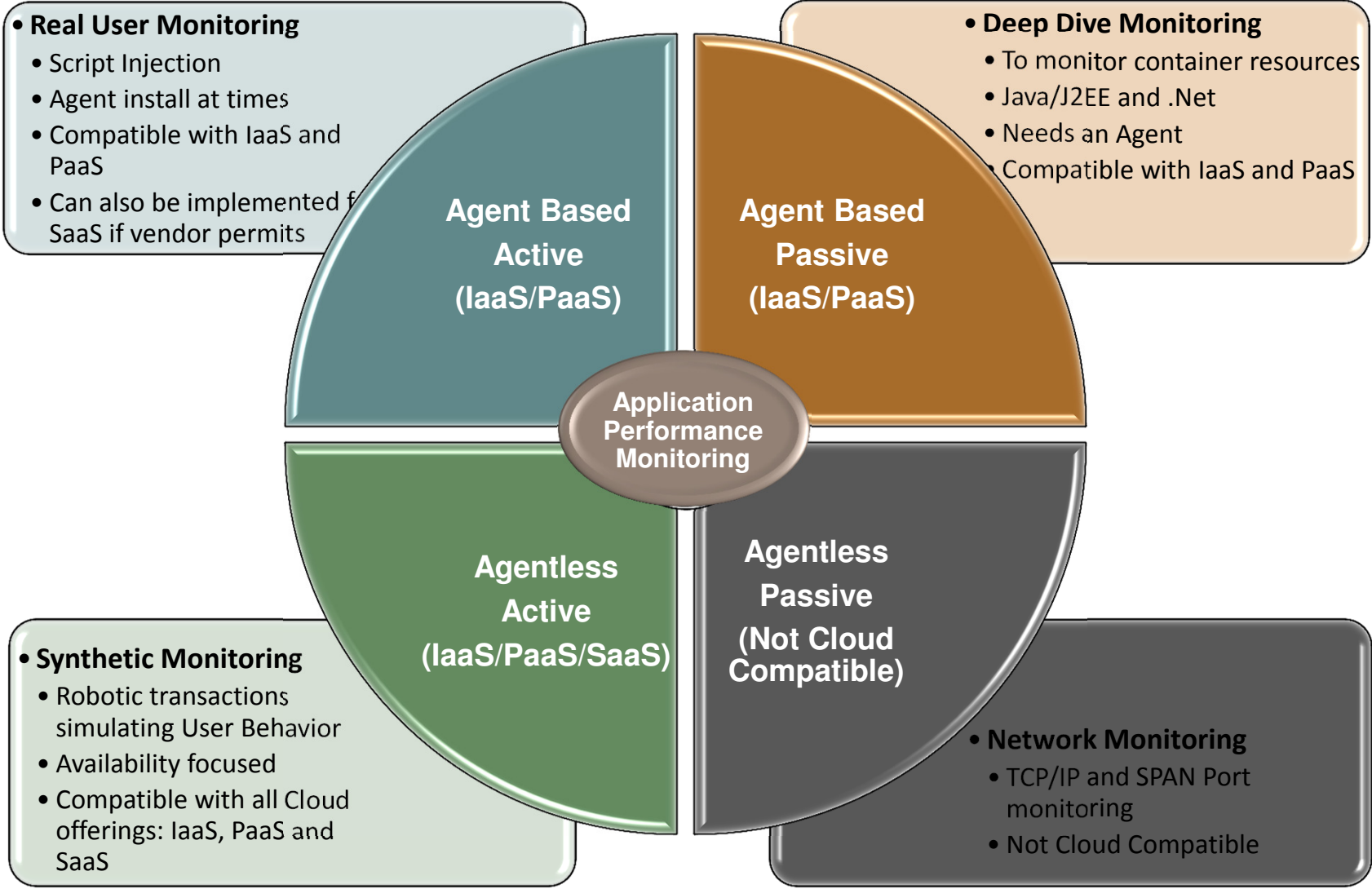
14

- Ensuring excellent end-user experience
- ‘Right Sizing’, validation and forecasting future growth, smart scaling
- Mitigating Risk of Unpredictable Performance
- Regaining Visibility and Control
- SLA Verification

15

CAPM – Making the right choice

APM for XaaS



Synthetic Vs RUM

17

- Synthetic
 - ▣ Active Monitoring, special focus on Availability
 - ▣ Based on 'Scripted' User Interactions
 - ▣ Gauges possible user experience
 - ▣ Executed from any location across the globe
 - ▣ Is Agentless
- RUM
 - ▣ Based on 'Real' User Interactions
 - ▣ A more definitive indicator of User Experience
 - ▣ Also provides a global perspective
 - ▣ Can be Agentless(JavaScript, Network tap) or Agent based

Some things to consider..

18

- Focus on
 - Business Transactions
 - Tiers
 - Baselines
 - End User Experience
- Unified View across Hybrid Environments
- APM Approach
 - Agent Vs Agentless
 - Synthetic Vs Real User Monitoring (RUM)
- Development and Optimization considerations

Gartner's Magic Quadrant for APM Suites

19

Magic Quadrant for Application Performance Monitoring Suites



Dimensions of APM Suites

End-user experience monitoring (EUEM)

- The capture of data about how end-to-end latency, execution correctness and quality appear to the real user of the application
- A secondary focus on application availability may be accomplished by synthetic transactions emulating the end user

Application topology discovery and visualization

- The discovery of the software and hardware infrastructure components involved in application execution, and the array of possible paths across which these components communicate to deliver the application

User-defined transaction profiling

- The tracing of user-grouped events, which comprise a transaction as they occur within the application as they interact with components discovered in the second dimension; this is generated in response to a user's request to the application.

Application component deep dive

- The fine-grained monitoring of resources consumed and events occurring within the components discovered in the application topology discovery and visualization dimension
- This includes the server-side components of software being executed.

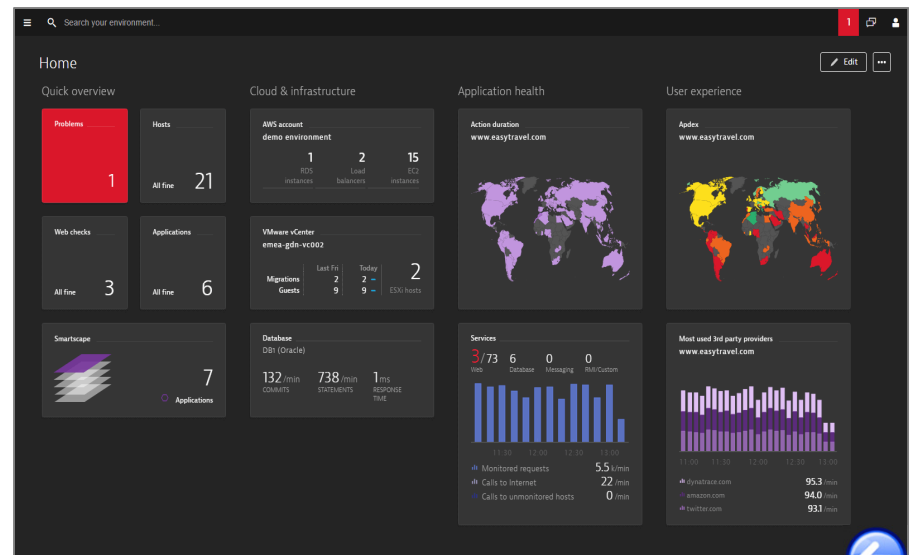
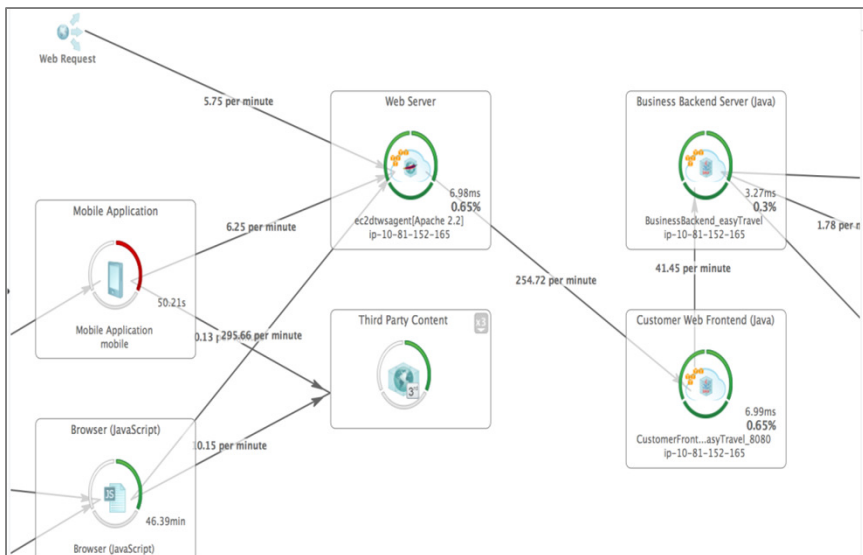
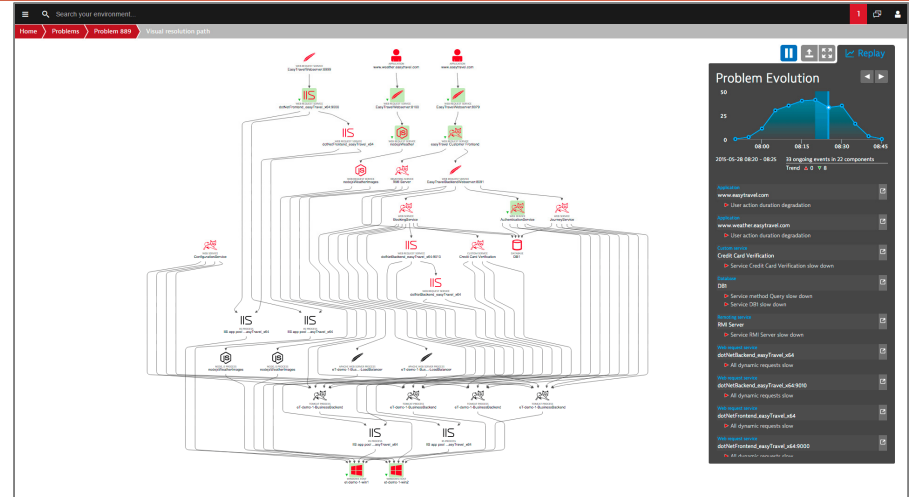
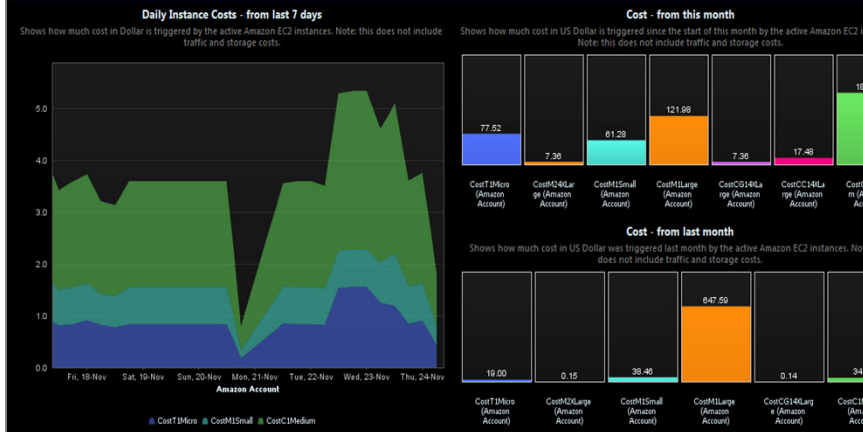
IT operations analytics (ITOA)

- The combination or usage of the following techniques: complex operations event processing, statistical pattern discovery and recognition, unstructured text indexing, search and inference, topological analysis, and multidimensional database search and analysis.

Dynatrace

20

Amazon EC2 Instance Costs - Dev Account



AppDynamics

21

The screenshot displays the AppDynamics interface for 'Customer-Survey-Services'. The top navigation bar includes Home, Applications, User Experience, Databases, Servers, Analytics, Dashboards & Reports, and Alert & Respond. The left sidebar lists various monitoring categories such as Application Dashboard, Business Transactions, Service Endpoints, Tiers & Nodes, Servers, Database Calls, Remote Services, Troubleshoot, More, Alert & Respond, Metric Browser, and Configuration.

The main view is divided into two sections. The upper section, 'Tier Flow Map', shows a flow diagram with nodes for 'Java' (2 Nodes, 4 calls/min, 28 ms) and 'Active MQ-CustomerQueue' (6 calls/min, - (async) - JMS). A 'QUEUE' icon is also visible. The lower section, 'End User Experience', features a world map with colored regions representing different geographic areas: Alaska, Canada, Virginia, Germany, France, Italy, India, Hong Kong, and Australia. To the right of the map, a panel displays 'End User Response Time 3857 ms' with a breakdown: First Byte Time - 1693 ms, Document Ready Time - 1429 ms, and Page Render Time - 705 ms. Below this, two tables list 'Highest End User Response Time' and 'Highest Requests / min' by region.

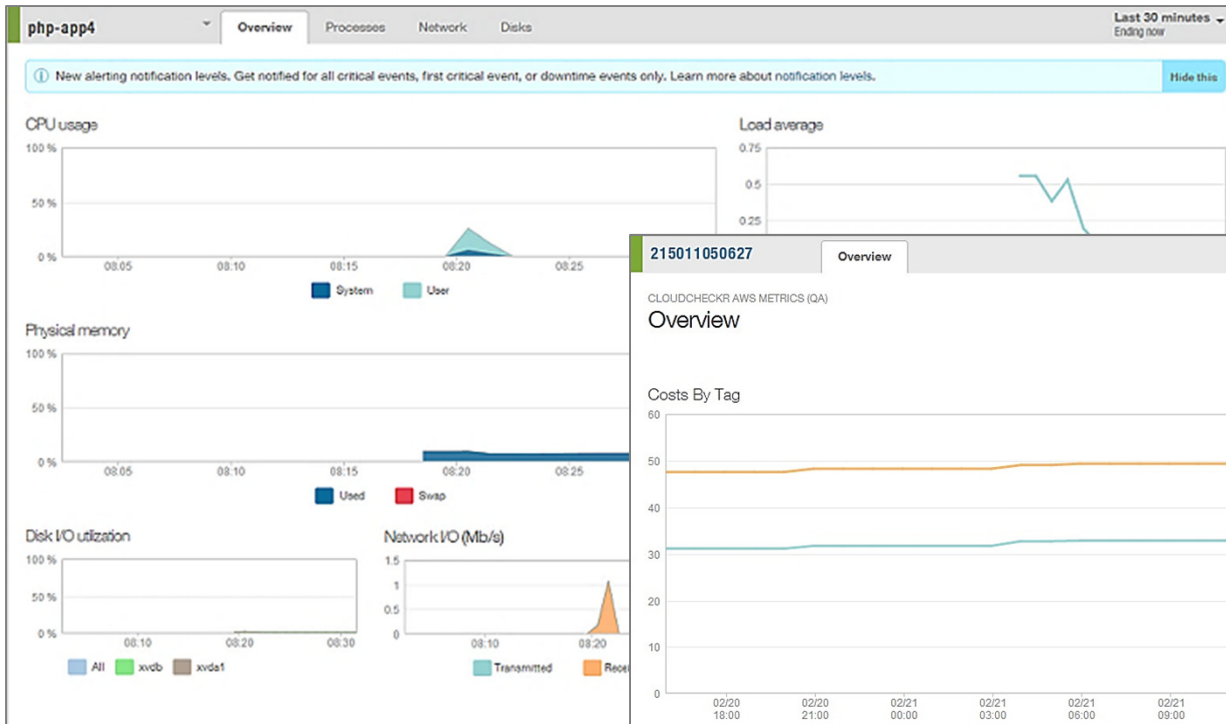
Region	Highest End User Response Time (ms)	Highest Requests / min
India	18528 ms	21 req / min
Virginia	9981 ms	24 req / min
Brazil	8417 ms	24 req / min
Germany	1608 ms	27 req / min
Italy	1173 ms	26 req / min
Australia	1033 ms	26 req / min
France	818 ms	27 req / min
Canada	508 ms	24 req / min

At the bottom, three charts provide a time-series view of performance metrics from 5:00 PM to 2:00 PM. The 'Requests' chart shows a total of 0.1M and 232 / min. The 'End User Response Time' chart shows a 3,857 ms average. The 'JavaScript Errors' chart shows 0% errors, with 0 total and 0 / min.



New Relic

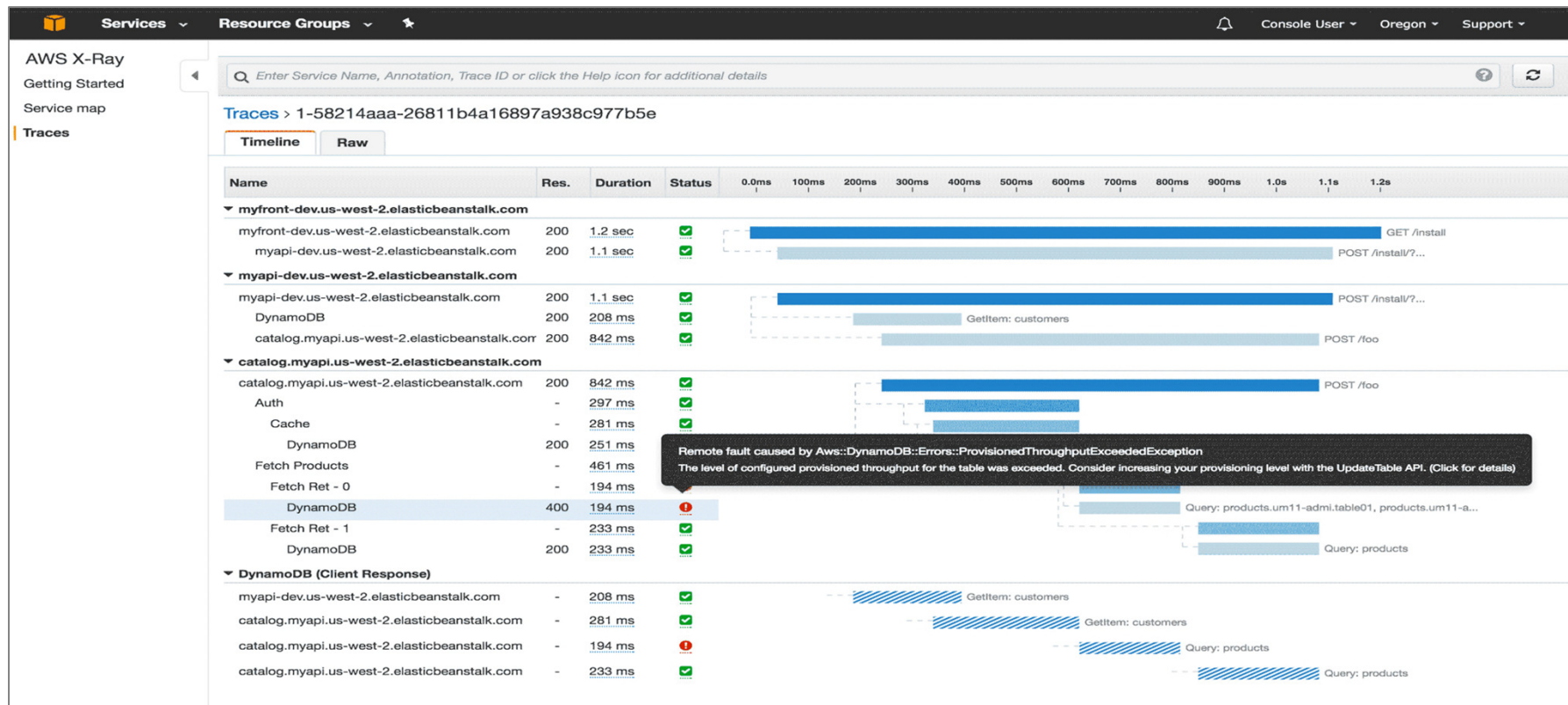
22



Monitoring Capabilities of Cloud Providers

23

- AWS- Amazon Cloud Watch and now X-Ray
 - ▣ Chargeback/Cost Metrics, Status and Availability Monitoring
 - ▣ Log Monitoring, Transaction Tracing, Service Mapping
 - ▣ For AWS only

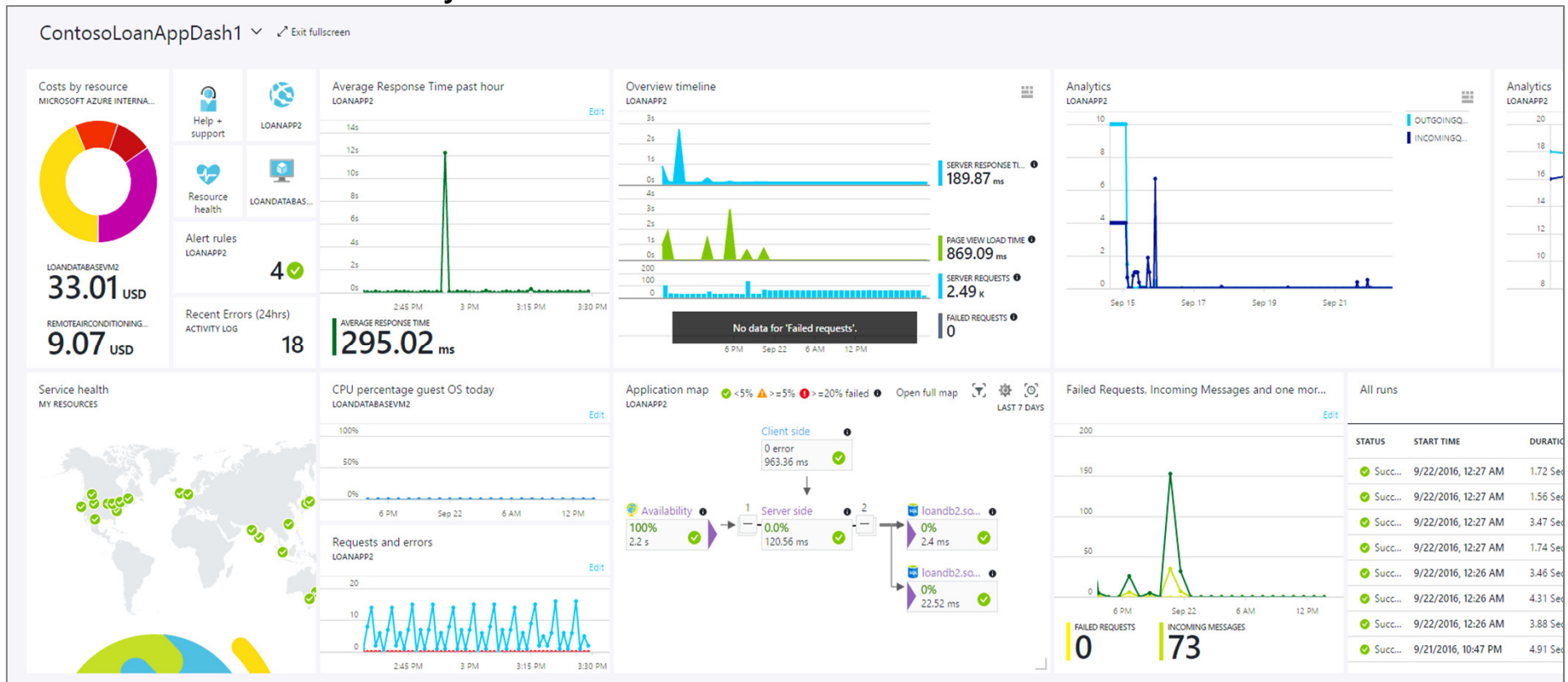


Monitoring Capabilities of Cloud Providers

24

□ Azure - Azure Monitor

- ▣ Activity and Diagnostic Logs, Alerts and Automated actions
- ▣ Metrics – Resources, Application Performance, Storage, Service Bus etc
- ▣ For Azure only

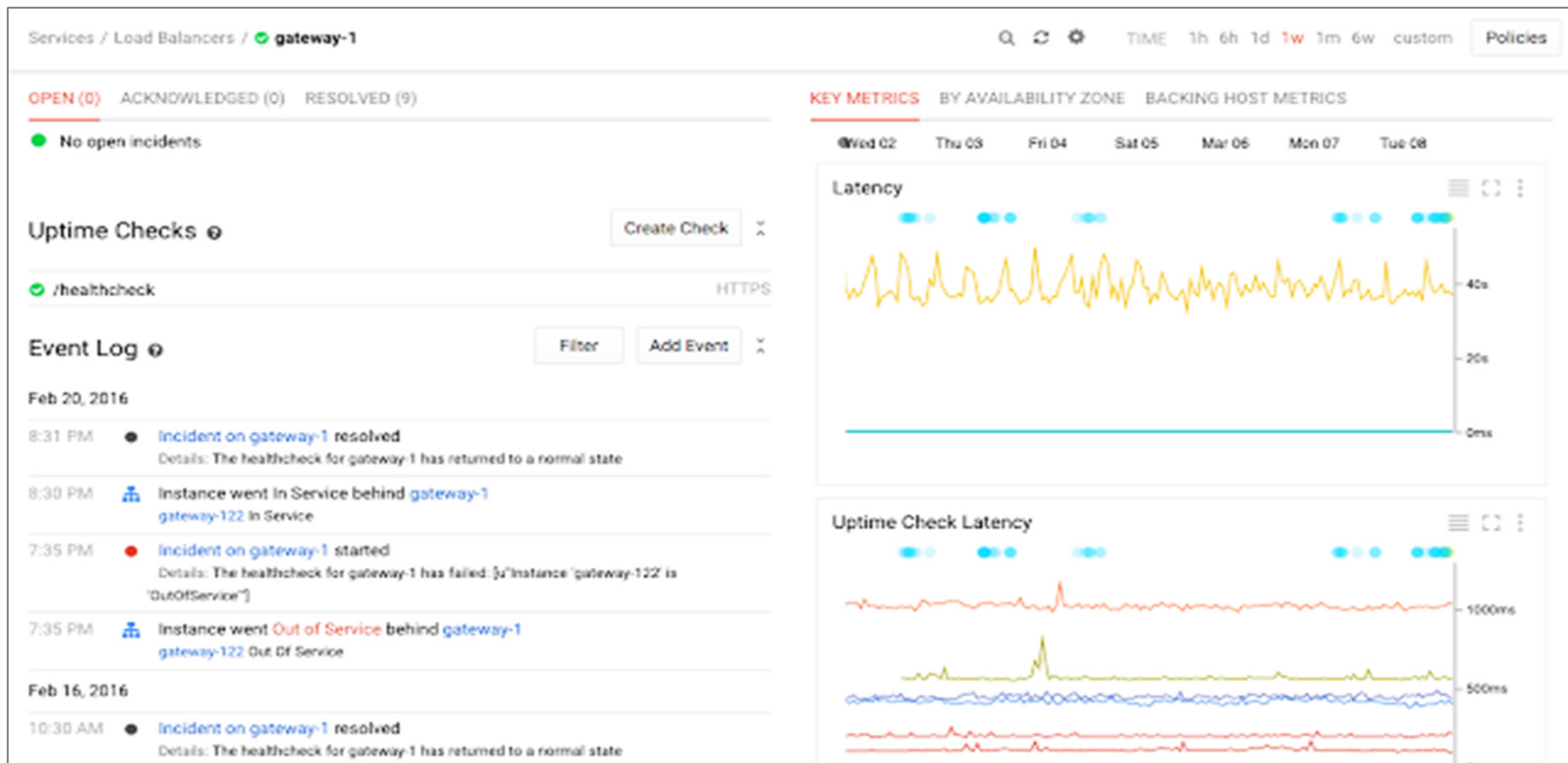


Monitoring Capabilities of Cloud Providers

25

□ Google - StackDriver

- ▣ Logging, event monitoring, diagnostics,
- ▣ Visibility into performance, uptime, and overall health of cloud applications
- ▣ For Google Cloud and AWS



26

Evaluating APM Solutions

Defining APM Requirements

27

Real End User Monitoring Solution

□ Must Have

- **Real-time capture and reporting on User Activity**, Response Times and Errors
- **Page Rendering** and object level details
- Business Transaction and **SLA monitoring**
- Business Analytics and User Session Insights
- Omni-channel visibility across Browsers and Mobile platforms
- **Geographic perspective**
- User defined transaction profiling for critical business applications
 - Integration with Runtime application performance
 - 'User-Complaint' to 'Code-Level' root cause for faster triage of Production issues

□ Good to Have

- User defined transaction profiling for non critical applications

Defining APM Requirements

28

Synthetic Monitoring Solution for Active Availability coverage

□ Must Have

- Page rendering and **object level details for Root Cause** analysis for critical business applications
- **Last Mile response** time from across the geography
- Multi Browser support for playback

□ Good to Have

- Third part content analytics for critical business applications
- Competitive Benchmarking
- **Integration with UEM, Runtime Application Performance**

Defining APM Requirements

29

Runtime Application Performance Monitoring Solution

□ Must Have

- Process health, **Container resource, event monitoring** (Heap, Thread, Connection pools, Servlets, EJBs)
- Auto discovery of **transaction topology** for service dependency mapping
- Transaction tracking/**stitching** for response time tier breakdown
- Code level '**deep-dive**'
- DB Performance (**SQL Captures**, Pool usage, Deadlocks, Hotspots)
- WebService, Messaging and Remote calls Performance
- Support for Java/J2EE, .Net, Microservices and other runtimes
- Monitoring of **on premises and cloud hosted** applications – seamless integration, presentation
- Real time **high granularity collection** and reporting
- No impact to application performance with minimal resource overhead

□ Good to Have

- Transaction/Method level resource cost evaluation for code optimization
- Runtime Thread Dump and Heap Dump capabilities
- Application Events- Errors, Exceptions, logs and Stack Trace
- Integration with DevOps, support development/test lifecycle for agility
 - Integration with Load Generation tools (HP Performance Center) for monitoring of Performance tests
 - Integration with IDEs and CI/CD solutions for code optimization

Defining APM Requirements

30

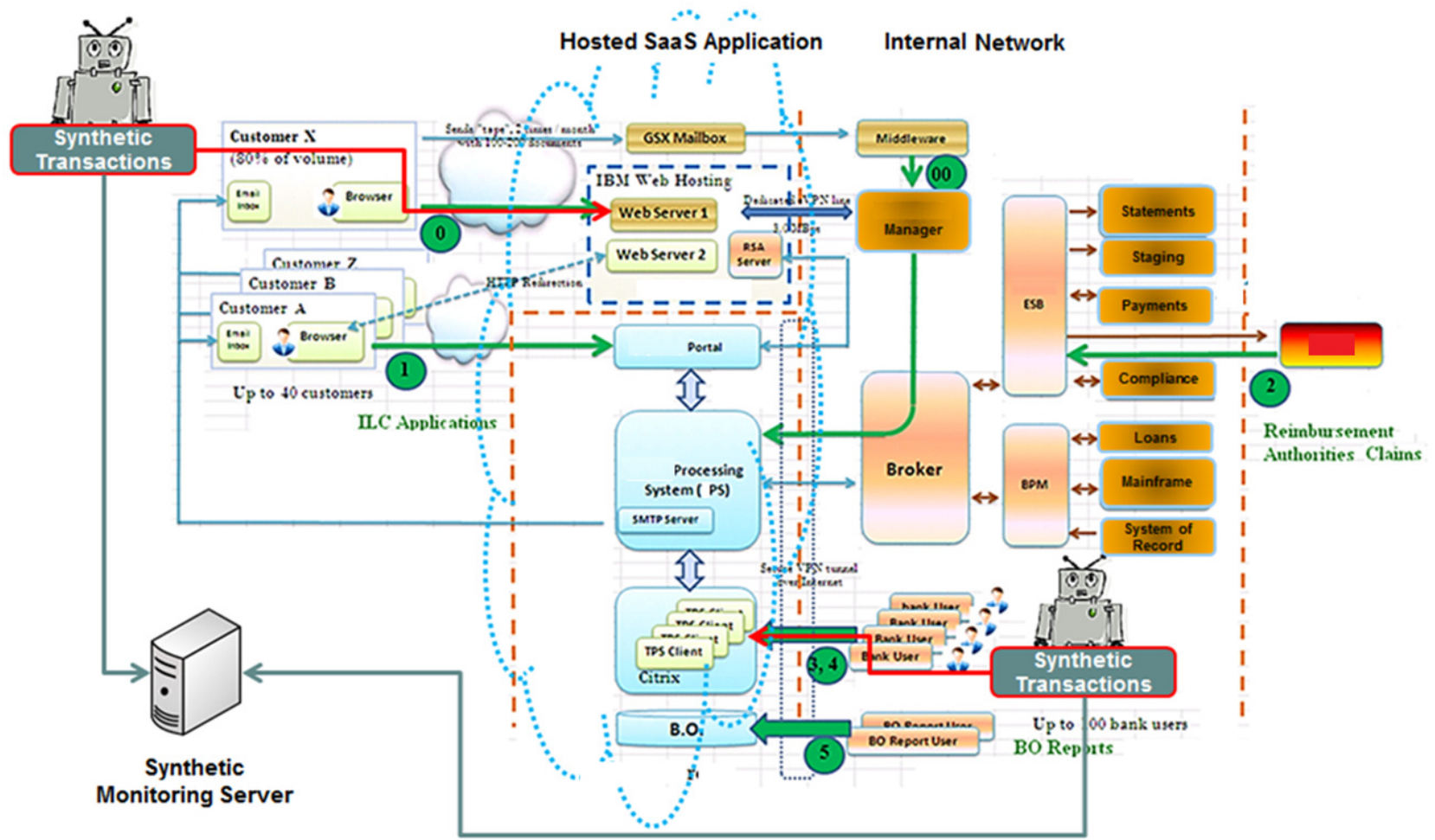
Other Considerations

- Data analysis, **presentation and reporting capabilities**
- **Integration with MoM** solutions
- Ease of deployment and Use
- Architecture of solution (**Aggregation vs Gapless design**), Data model, Agent design (**application resource reliant vs independent**)
- Converged platforms providing one consolidated view – Single pane of glass
- One **unified solution** for multi purpose APM
 - On-premises and Cloud (IaaS/PaaS/SaaS) applications
 - Monitoring capability for Containers and Microservices
- APM-aaS availability
- Vendor Evaluation
 - Roadmap and technology/trend adoption towards Digital Performance Management
 - **Licensing Model**
 - Current Market Share
 - Customer Service and Product Support
 - **Gartner's Evaluation**
- **Cost of Deployment, Maintainability and Scalability**

31

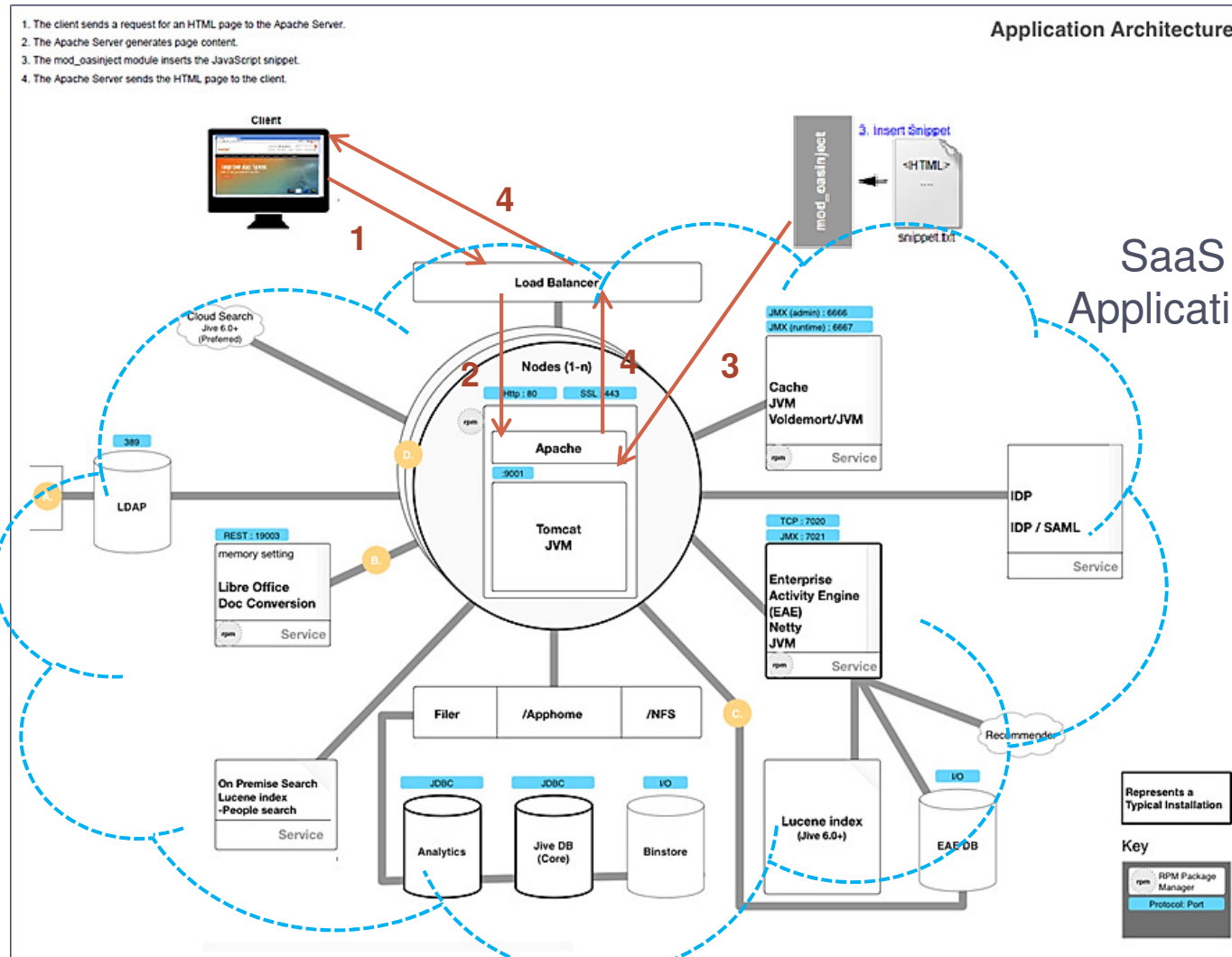
Examples

Synthetic Monitoring



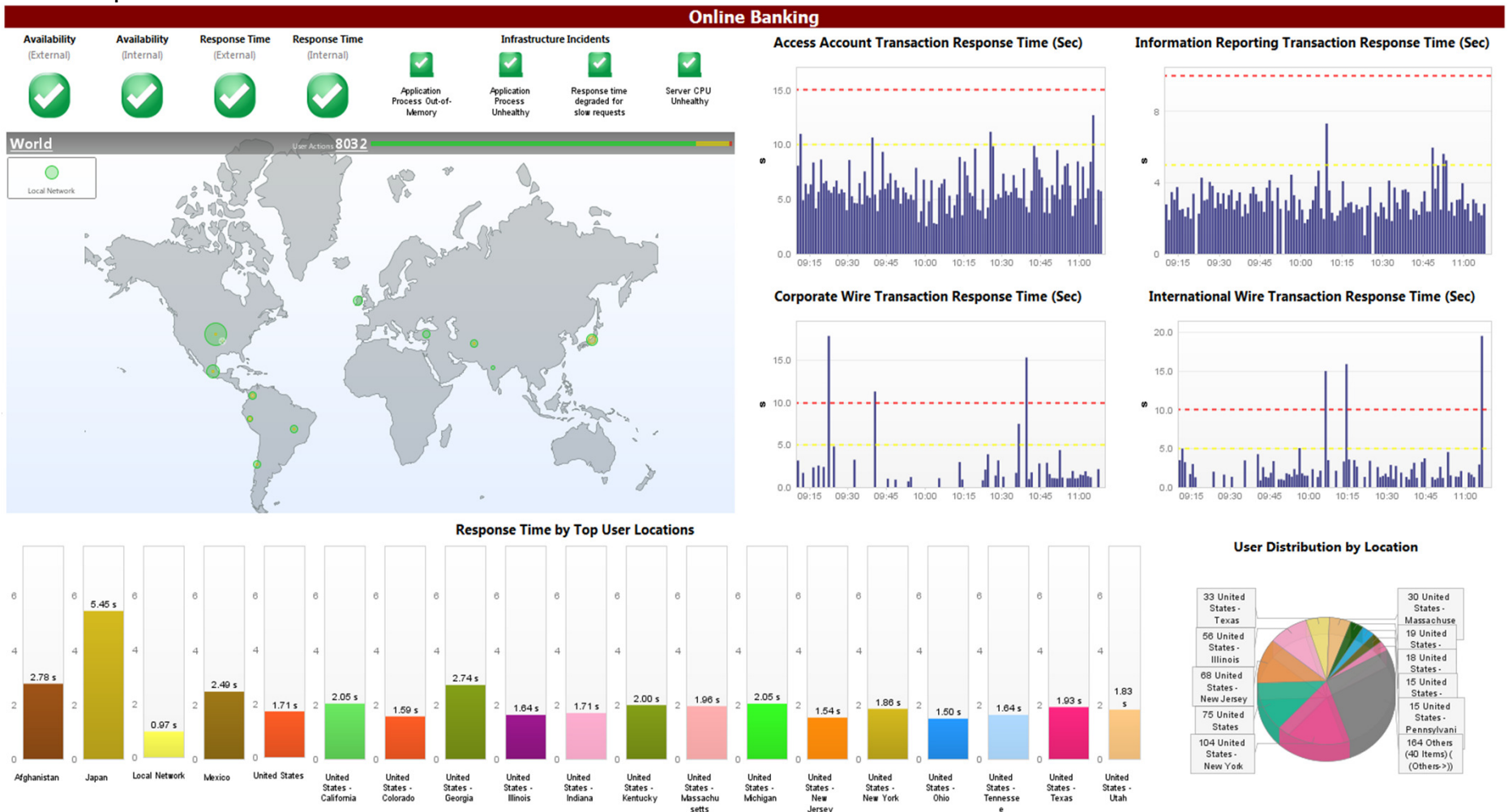
Real User Monitoring (JavaScript Injection)

34



RUM Reports

- Capability to track 'critical' transactions, visibility into top customer locations, branches and associated user experiences



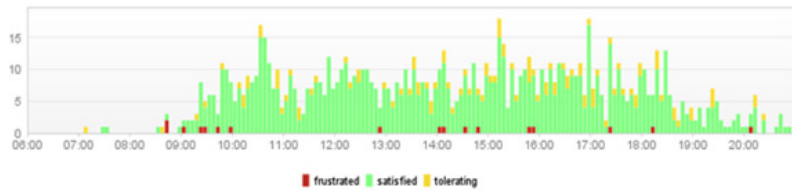
RUM Reports

36

User Experience Overview

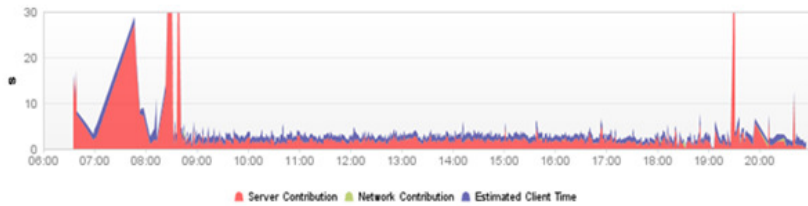
User Experience

Green- Response Time < 4 Sec Yellow- Response Time 4 - 8 Sec Red- Response Time > 8 Sec



Response Time Breakdown

Breakdown of user action time into client, network and server time.



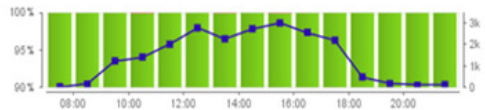
User Actions Health

Compare 'Today' against 'Yesterday'

From last Mon 07:00 to last Mon 22:00



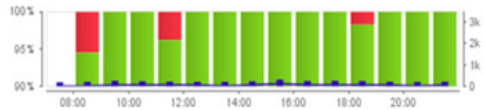
10
Failed User Actions
0.05 %



From May-30 07:00 to May-30 22:00



3
Failed User Actions
0.49 %

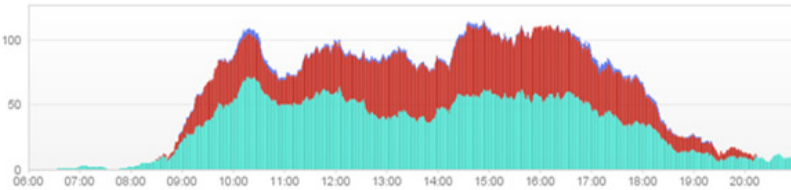


Error Hotspots by Application (Custor

9
1

Error Hotspots (Custom)

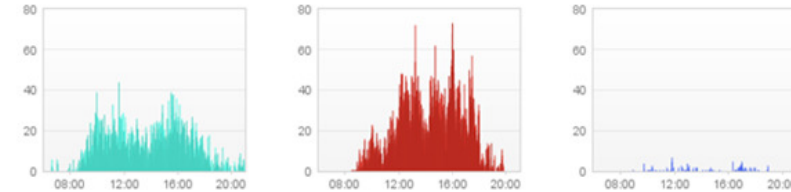
Concurrent Sessions



Session Length (min)



User Actions per min



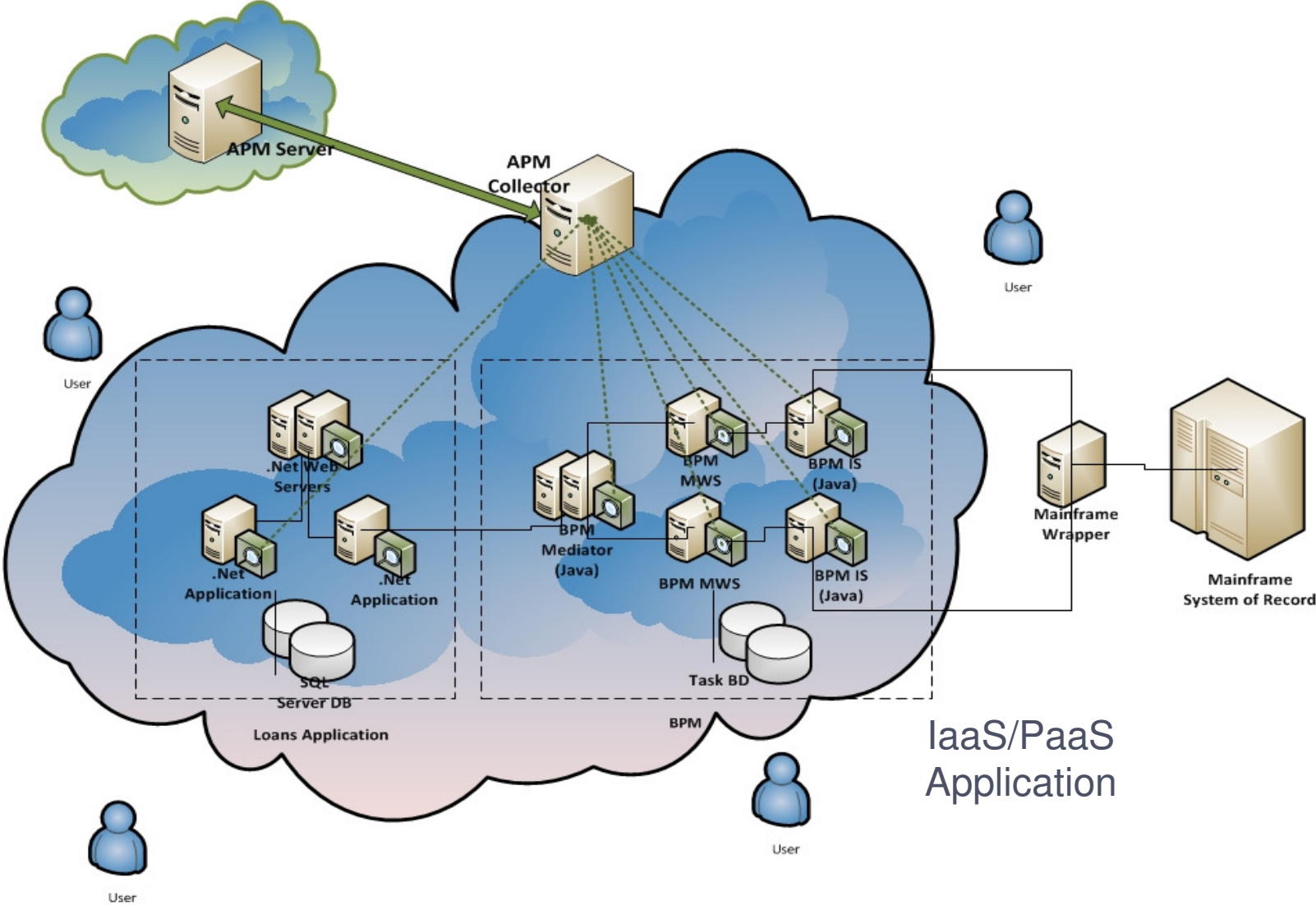
Top Slowest User Actions

Name	Response Time [s]	Count
▷ /Web/	56.38	2
▷ keypress <BACKSPACE> on "Page: Inde	29.64	1
▷ /Web/FX/	27.03	1
▷ click on	26.03	4
▷ /Web/MM/	23.09	6
▷ /Web/FX/	21.06	1
▷ /Web/MM,	20.89	1
▷ click or	18.94	2
▷ click on "Cancel	16.10	1
▷ click on "ui-button-icon-primary ui-ico	15.73	1
▷ click on	15.32	1
▷ click on "Reject"	13.97	28
▷ click on	13.81	1
▷ /Web/MM	13.50	4
▷ click on	13.00	2
▷ click on "iicon-copy	11.40	18
▷ click on	11.38	1
▷ click on	11.04	1
▷ click on "Pending	10.66	1

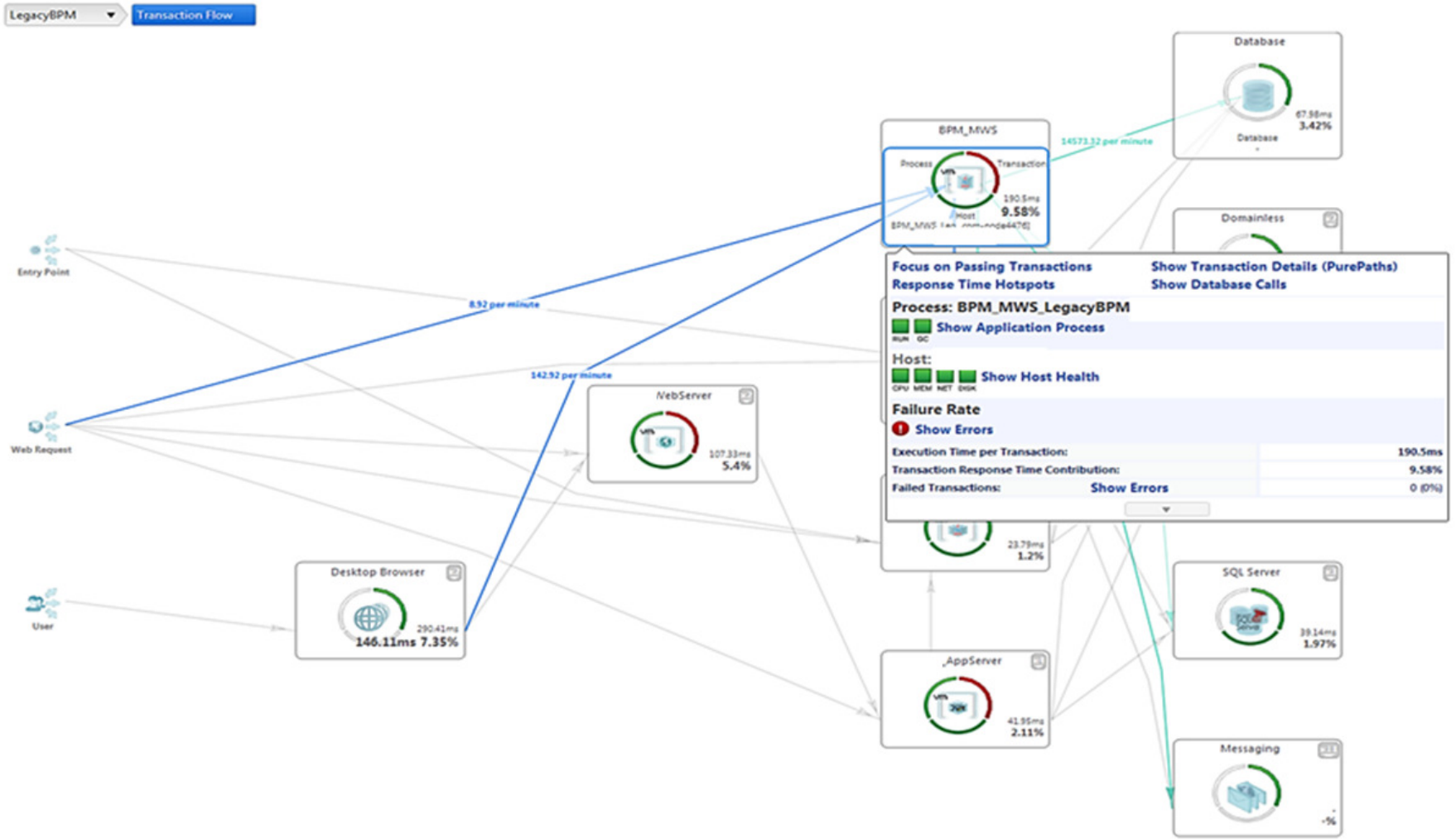
Most Frequent User Actions

Name	Count	Response Time [s]
▷ click on "Search"	2950	1.97
▷ click on "Submit"	1298	3.22
▷ click on "Ok"	1126	4.67
▷ /Web/	1085	3.84
▷ click on "iew"	1052	2.52
▷ click on "Incoming	821	0.41
▷ click on "ui-button-icon-primary ui-ic	607	4.50
▷ click on "javascript:Home();"	599	3.92
▷ click on "iicon	513	2.38
▷ /	513	4.43
▷ click on	482	0.51
▷ click on "Home"	462	4.40
▷	405	0.62
▷ click on "ui-button-icon-primary ui-ic	339	7.10
▷ click on "Yes"	279	5.24
▷ click on "icon-edit"	269	5.13
▷ click on "Dashboard"	221	5.70
▷ click on "Approve"	192	5.47
▷ click on	188	2.94

Agent Based Monitoring

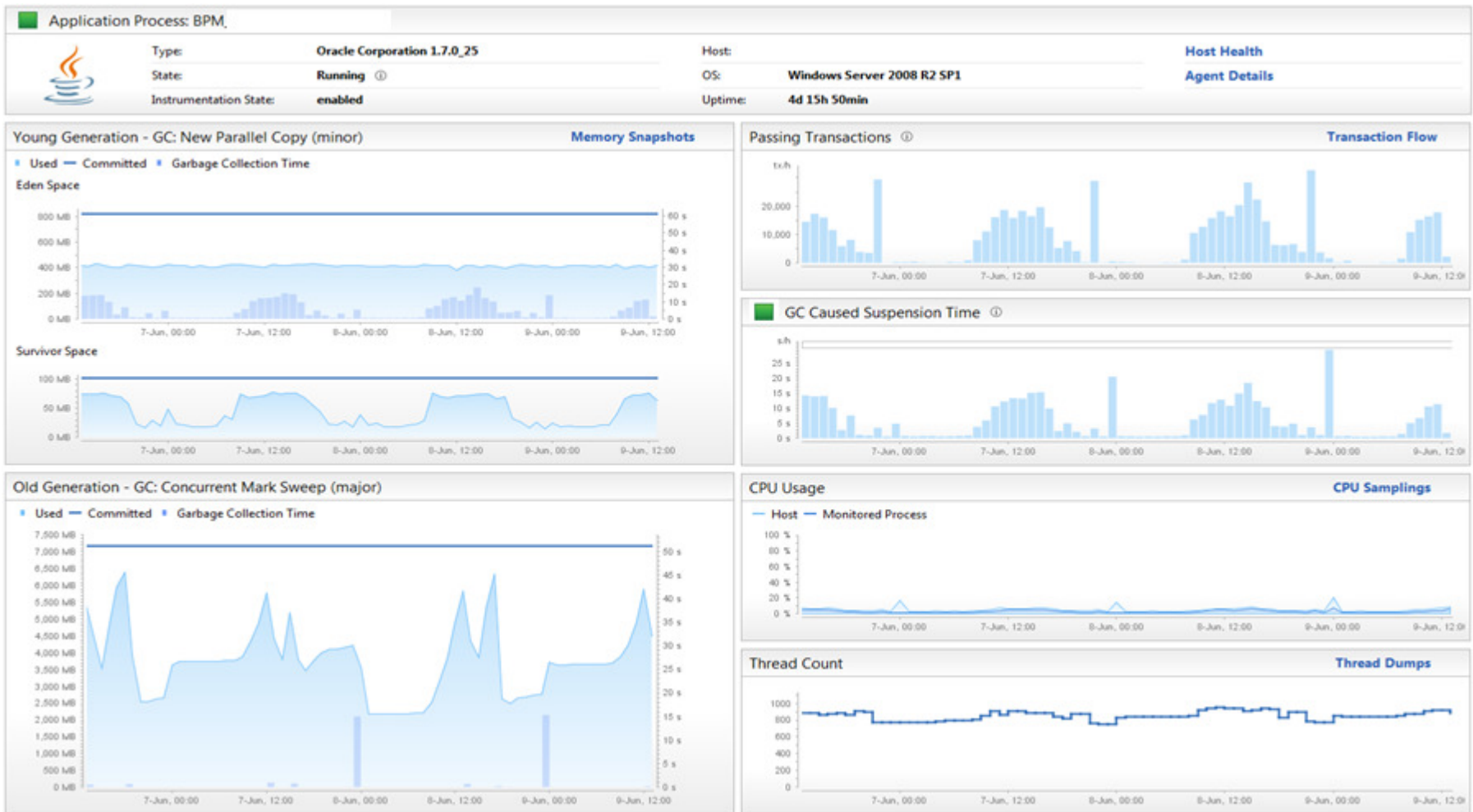


Agent Based Monitoring - Reports



Agent Based Monitoring - Reports

- Garbage Collection health, Suspension time, Process throughput, Thread Count



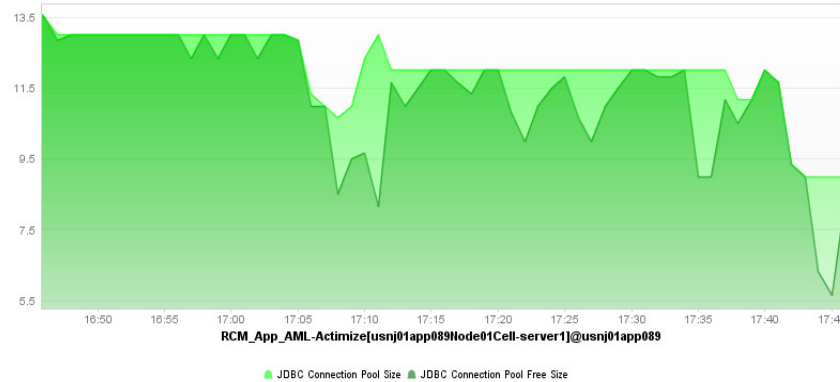
Agent Based Monitoring - Reports



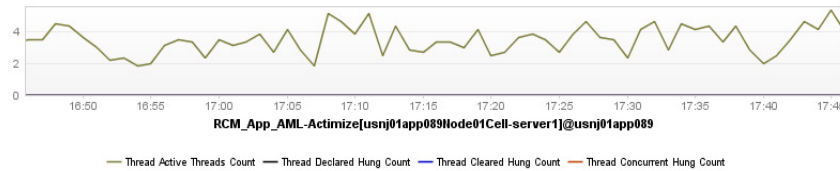
WebSphere Server Health

This Dashboard shows a few important WebSphere health metrics. Use it via an agent Drill down or by setting the Dashboard filter to your WebSphere agent.

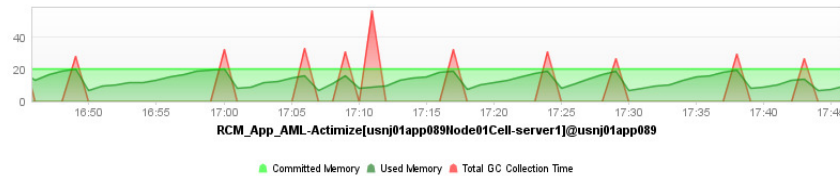
Managed Connections Pool Sizes



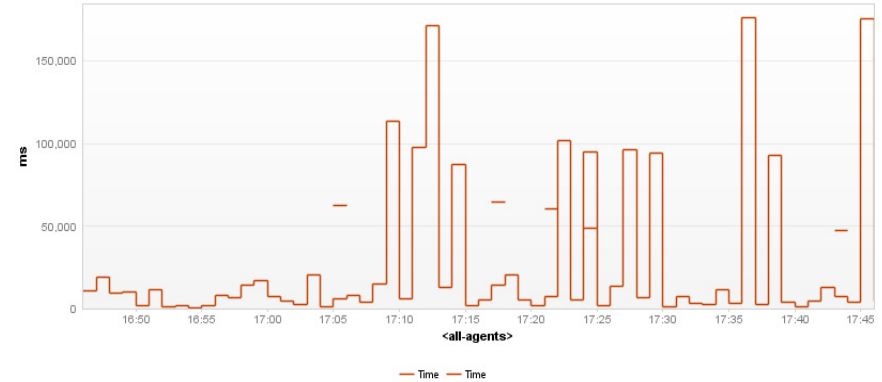
Thread Info



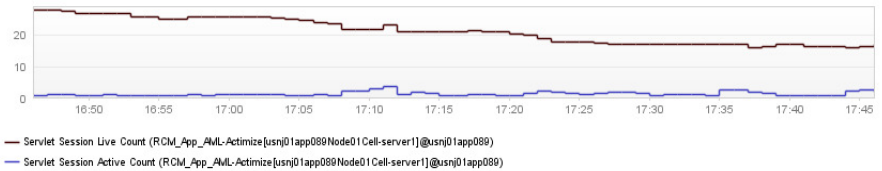
Memory



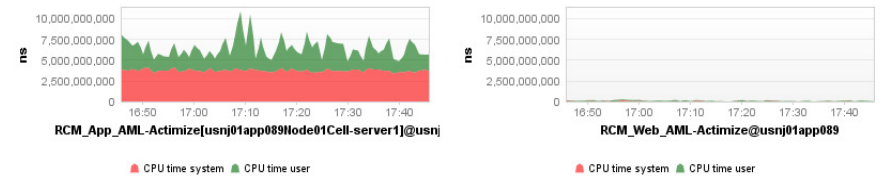
Servlet Response Times



Servlet Session Counts



CPU Usage



Agent Based Monitoring - Reports

- Giving the Developers capability to drill in all the way to the source of the problem
 - ✦ Deep dive into the code, identifying root cause at the method level
 - Long running methods contributing to latency; Also identifying resource intensive methods

The screenshot displays a monitoring tool interface with two main sections:

PurePath Table:

PurePath	Response Time [ms]	Breakdown	Size	Agent	Application	Start Time
/Web/Loan/Advance	3541.25	cpu (79.0%)	32	Loan	_WebServer...	2016-06-07 13:08:07.482
/Web/Loan/Fee	2059.23	io (100.0%)	7	Loan	_WebServer...	2016-06-07 13:08:07.497
/Web/Loan/CreateNewLoan	1747.22	cpu sync (49.0%)	190	Loan	_WebServer...	2016-06-07 13:08:01.897
/Web/Loan/CashFlow	1622.42	io (95.0%)	7	Loan	_WebServer...	2016-06-07 13:08:07.497
/Web/Loan/LoanInputSheet	1138.81	cpu (41.0%) io (57.0%)	73	Loan	_WebServer...	2016-06-07 13:08:03.644
/Web/Loan/Comments	1060.81	io (97.0%)	7	Loan	_WebServer...	2016-06-07 13:08:07.497

PurePath Hotspots: A horizontal bar chart showing execution time for various PurePaths. The x-axis represents execution time in milliseconds, ranging from 0 to 4890. The y-axis lists PurePaths. The longest bar corresponds to the /Web/Loan/Advance PurePath.

PurePath Tree (showing only relevant nodes):

Method	Argument	Exec Total [ms]	Breakdown	Class	API	Agent
run()		0.03	cpu (91.0%)	DCProcessor	Softwareag	BPM_MED_CBR@...
submit(Runnable)		0.07	cpu (98.0%)	AbstractExecuto...	Threading	BPM_MED_CBR@...
Asynchronous Invocation		-	-	-	-	BPM_MED_CBR@...
Asynchronous Path (Thread)		-	-	-	-	BPM_MED_CBR@...
run()		0.61	cpu (95.0%)	DCProcessor	Softwareag	BPM_IntegrationS...
Synchronous Path (HTTP)		-	-	-	-	BPM_IntegrationS...
process(ProtocolState state)	ws/BPMCBRLan...	2519.22	io (99.0%)	HTTPRootWebSer...	Servlet	BPM_IntegrationS...
process(ProtocolState state)		2519.07	io (99.0%)	HTTPTransportLis...	Servlet	BPM_IntegrationS...
doService(HTTPRequest, HTTPResponse, HTTPState, Message)		2517.49	io (99.0%)	HTTPTransportList...	Wm	BPM_IntegrationS...
service(AxisHttpRequest, AxisHttpResponse, MessageCont)		2517.49	io (99.0%)	HTTPWorker	Java Web ...	BPM_IntegrationS...
receive(MessageContext msgContext)		2517.49	io (99.0%)	AxisEngine	Java Web ...	BPM_IntegrationS...
receive(MessageContext)		2499.17	io (100.0%)	ISDynamicMessag...	Wm	BPM_IntegrationS...
receive(MessageContext)		2499.17	io (100.0%)	AbstractMessageR...	Java Web ...	BPM_IntegrationS...
invokeBusinessLogic(MessageContext)		2499.17	io (100.0%)	ISDynamicMessag...	Wm	BPM_IntegrationS...
invokeBusinessLogic(MessageContext)		2499.17	io (100.0%)	AbstractInOutMes...	Java Web ...	BPM_IntegrationS...
invokeBusinessLogic(MessageContext, A)		2499.17	io (100.0%)	ISDynamicMessag...	Wm	BPM_IntegrationS...
baseInvoke(IData pipe)		2499.17	io (100.0%)	FlowSvcImpl	Java Web ...	BPM_IntegrationS...
DatabaseSummary		11.20	-	-	-	BPM_IntegrationS...
1 x Connection Acquisition @		8.59	-	-	Connectio...	BPM_IntegrationS...
1 x select-replace(key) CBRL		1.40	-	-	JDBC	BPM_IntegrationS...
1 x select user		1.17	-	-	JDBC	BPM_IntegrationS...
1 x Create Statement @ usnj01		0.05	-	-	JDBC	BPM_IntegrationS...
FlowException: [ISC.0049.9028] In	[ISC.0049.9028] In...	-	-	-	FlowException	Exception
FlowException: [ISC.0049.9028] In	[ISC.0049.9028] In...	-	-	-	FlowException	Exception
send(MessageContext msgContext)		1.39	cpu (99.0%)	AxisEngine	Java Web ...	BPM_IntegrationS...
ObjectDisposedException: Cannot access a disposed object. Object name: 'S	Cannot access a d...	-	-	-	ObjectDisposEx...	Exception
SleepInternal(int)		4889.66	sync (92.0%)	Thread	Btmu	Loan
get_Channel()		152.80	io (92.0%)	ClientBase'1	.NET WCF	Loan
ImportMembersMapping(String, String, XmlReflectionMember[], boolean, boolean,		152.80	io (92.0%)	XmlReflectionImpa...	XML Proce...	Loan
GetCustomAttributes(RuntimeType, RuntimeType, boolean)		152.80	io (92.0%)	CustomAttribute	Reflection	Loan

And that's a wrap!

42

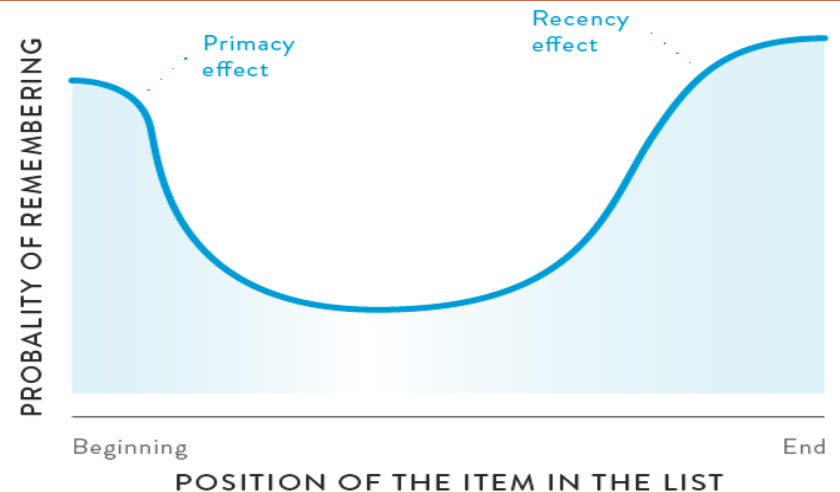
□ We talked about

- Cloud
 - Cloud deployment models
 - Service offerings
- Cloud Challenges
- APM Solutions
- Considerations for APM solutions

□ Irrespective of

- What cloud solution you adopt
- Which applications you migrate to it
- What monitoring methodologies you choose
- What APM solutions you deploy

....as an IT organization, you have to move from traditional monitoring/management paradigms to business service and end user focused, holistic solutions to survive this new age of Cloud



Questions?

43



References

44

[Gartner's Magic Quadrant for Application Performance Monitoring Suites](#)

[Things to Know About APM in the Cloud](#)

[Managing Performance of Cloud Based Applications](#)

[APM for Cloud Computing](#)

[Anatomy of APM](#)

[Cloud Predictions](#)

[Obscurity of Cloud](#)

[Duality of APM](#)

[Azure Monitor](#)

[AWS X-Ray](#)

[Google Stackdriver](#)

[Dynatrace](#)

[AppDynamics](#)

[New Relic](#)

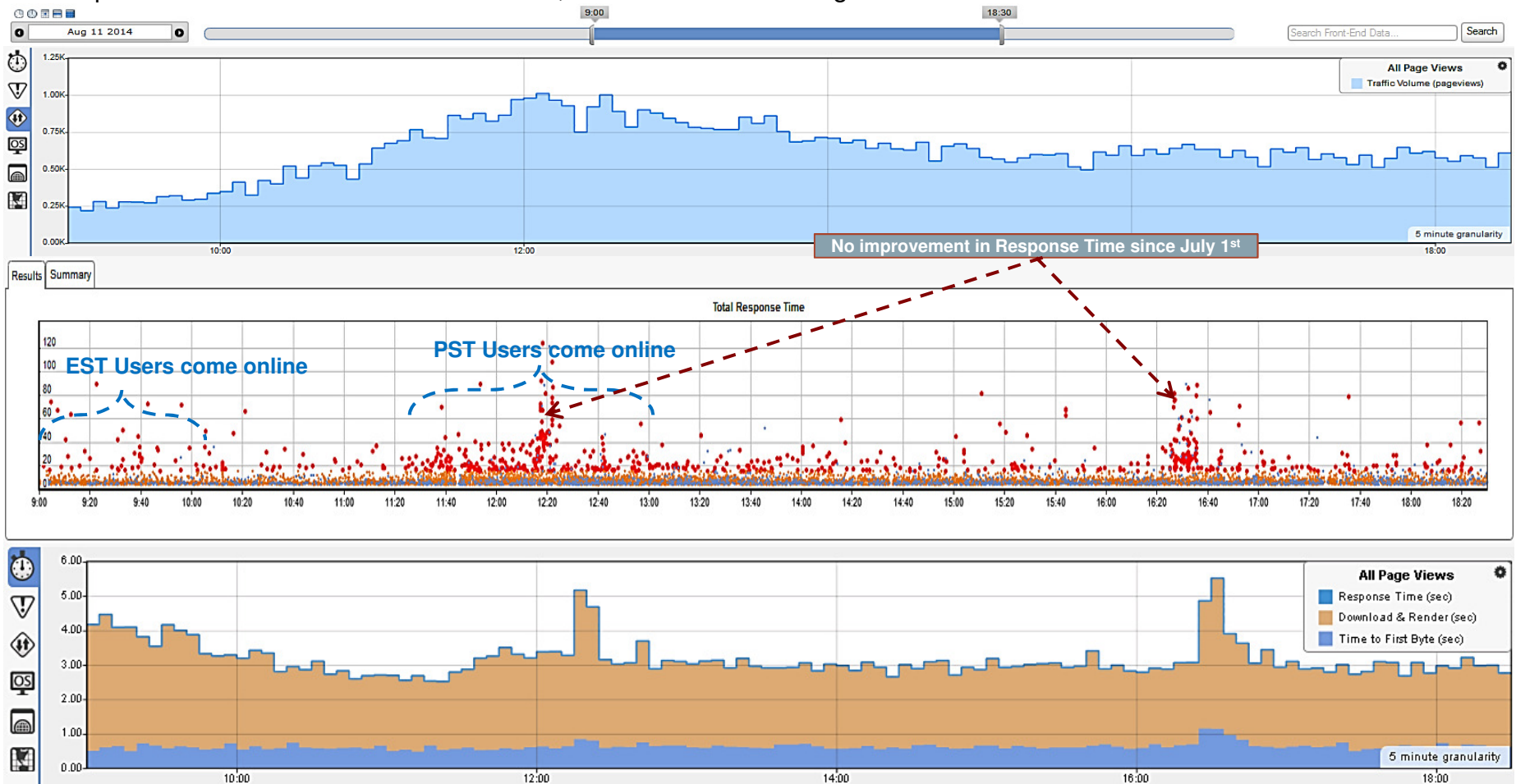
45

APPENDIX

RUM Reports (JavaScript Injection)

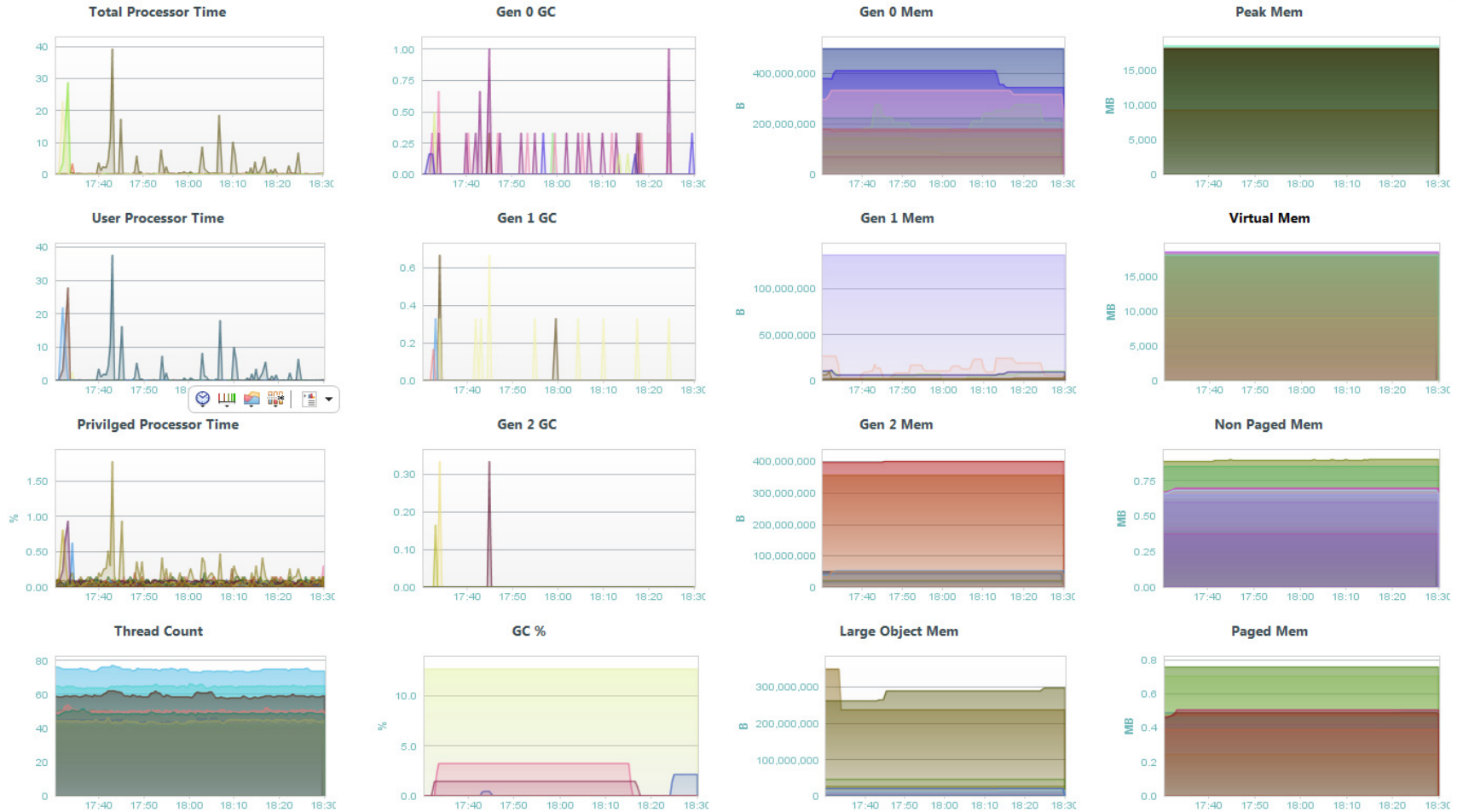
46

- Reporting on Average Application throughput during typical and peak times
- Average response times for users with anomalies
- Response Time Breakdown between Server, Network and Rendering time



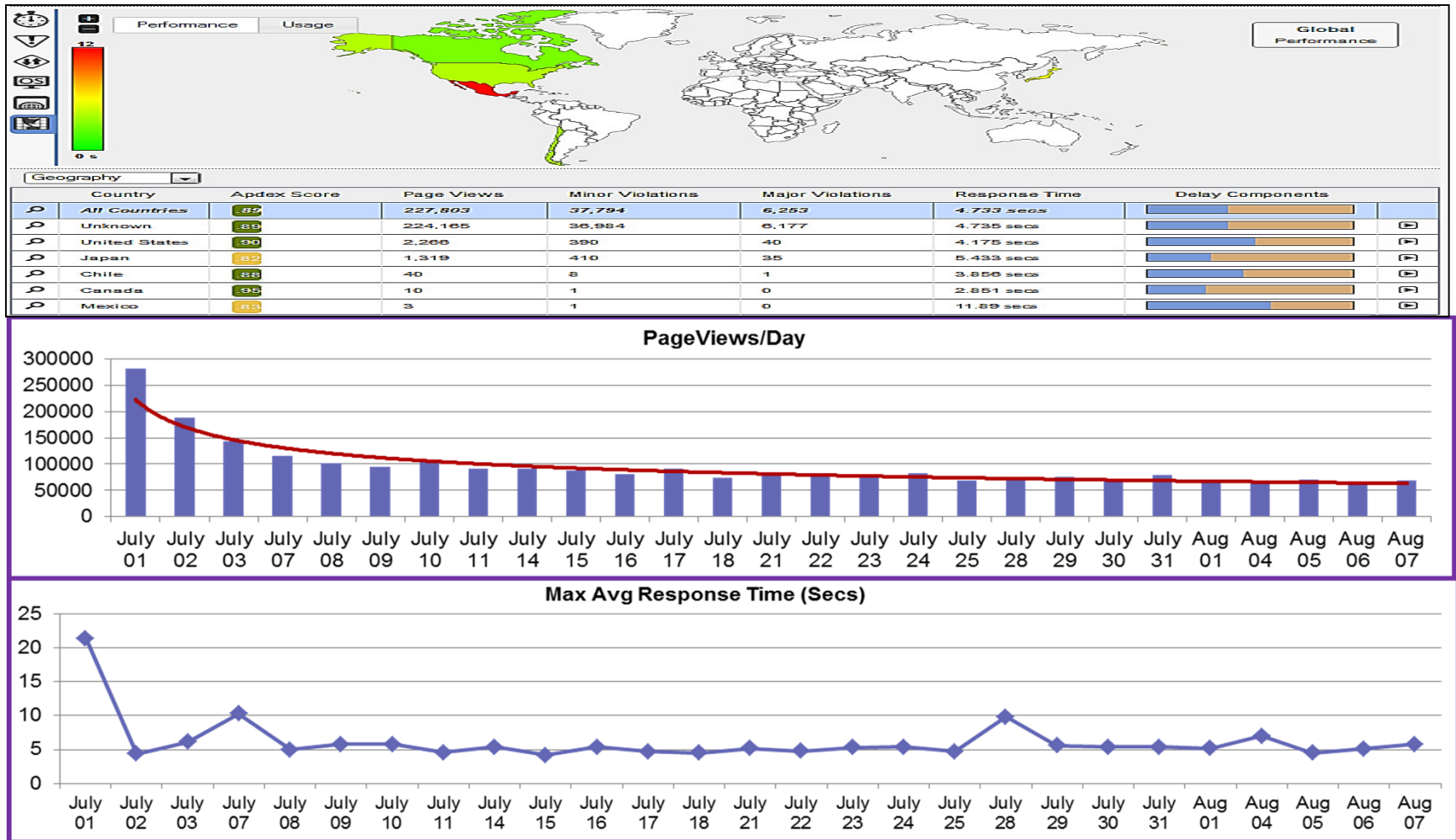
Agent Based Monitoring - Reports

.NET System Monitoring



RUM Reports (JavaScript Injection)

- Understanding User Experience across the globe
- Application Workload Trending for forecasting growth



RUM Reports (Agent Based)

49

Application Process: IIS 7.5



Type: **Microsoft II!**

Host:

Total Requests: **3,006** ⓘ

[Host Health](#)

State: **Running**

OS: **Windows Server 2004**

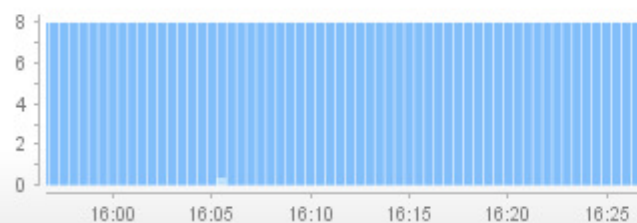
Total Traffic: **11.22** ⓘ

[Agent Details](#)

Uptime: **1d 8h 6min**

Threads

— busy — idle



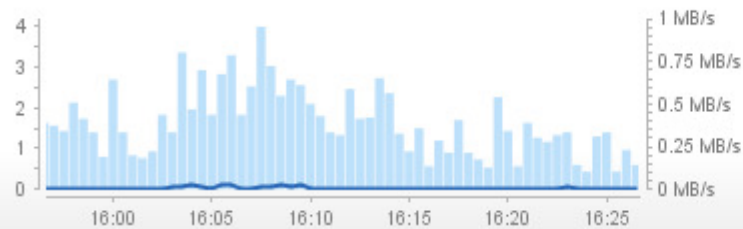
Transfer Rate

— kB/req



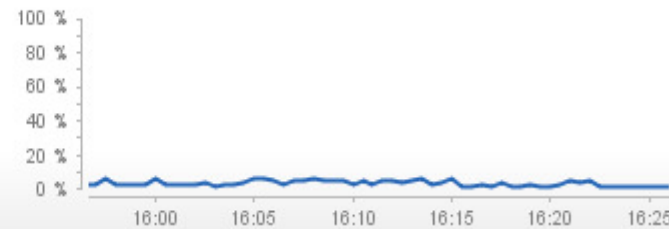
Requests

■ requests/sec — MB/sec



CPU Usage

— Host



SLA/SLM and Reporting

50

- Monitoring/Reporting Service levels for overall application as well as at ‘component level’
- Combining ‘Active’ and ‘Passive’ monitoring for complete visibility
- Proactive alerting
- Trending Transaction Volumes and End User Experience
- Understanding ‘cost per transaction’
 - ▣ Correlating business throughput to system resource for ‘right-sizing’

Optimization for Cloud Applications

51

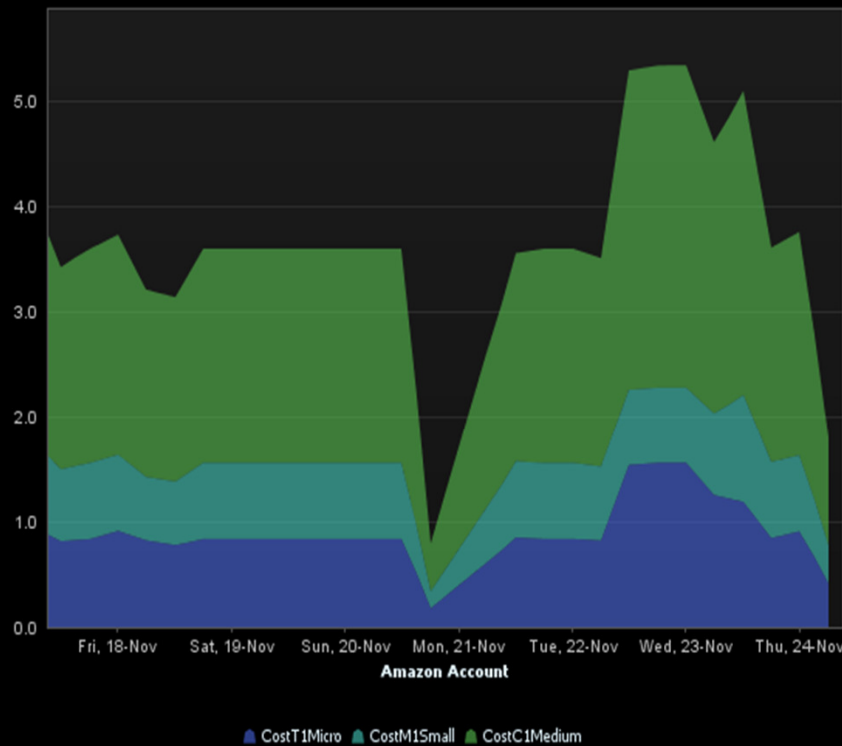
- Content Caching
- Compression algorithms – data is expensive in the cloud
- Optimized page rendering
 - ▣ Parallel rendering of content
- WAN Quality of Service
- Chatty Applications
 - ▣ Reducing acknowledgements to reduce WAN Latency

Dynatrace – Instance Cost

Amazon EC2 Instance Costs - Dev Account

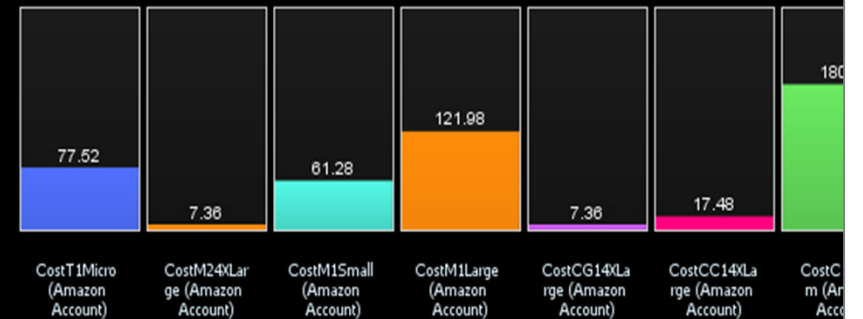
Daily Instance Costs - from last 7 days

Shows how much cost in Dollar is triggered by the active Amazon EC2 instances. Note: this does not include traffic and storage costs.



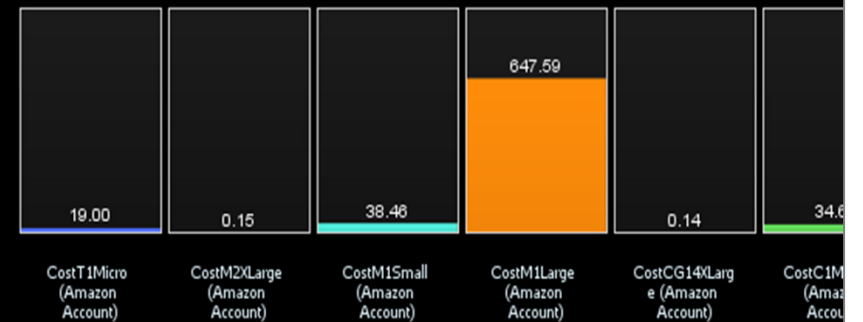
Cost - from this month

Shows how much cost in US Dollar is triggered since the start of this month by the active Amazon EC2 instances. Note: this does not include traffic and storage costs.



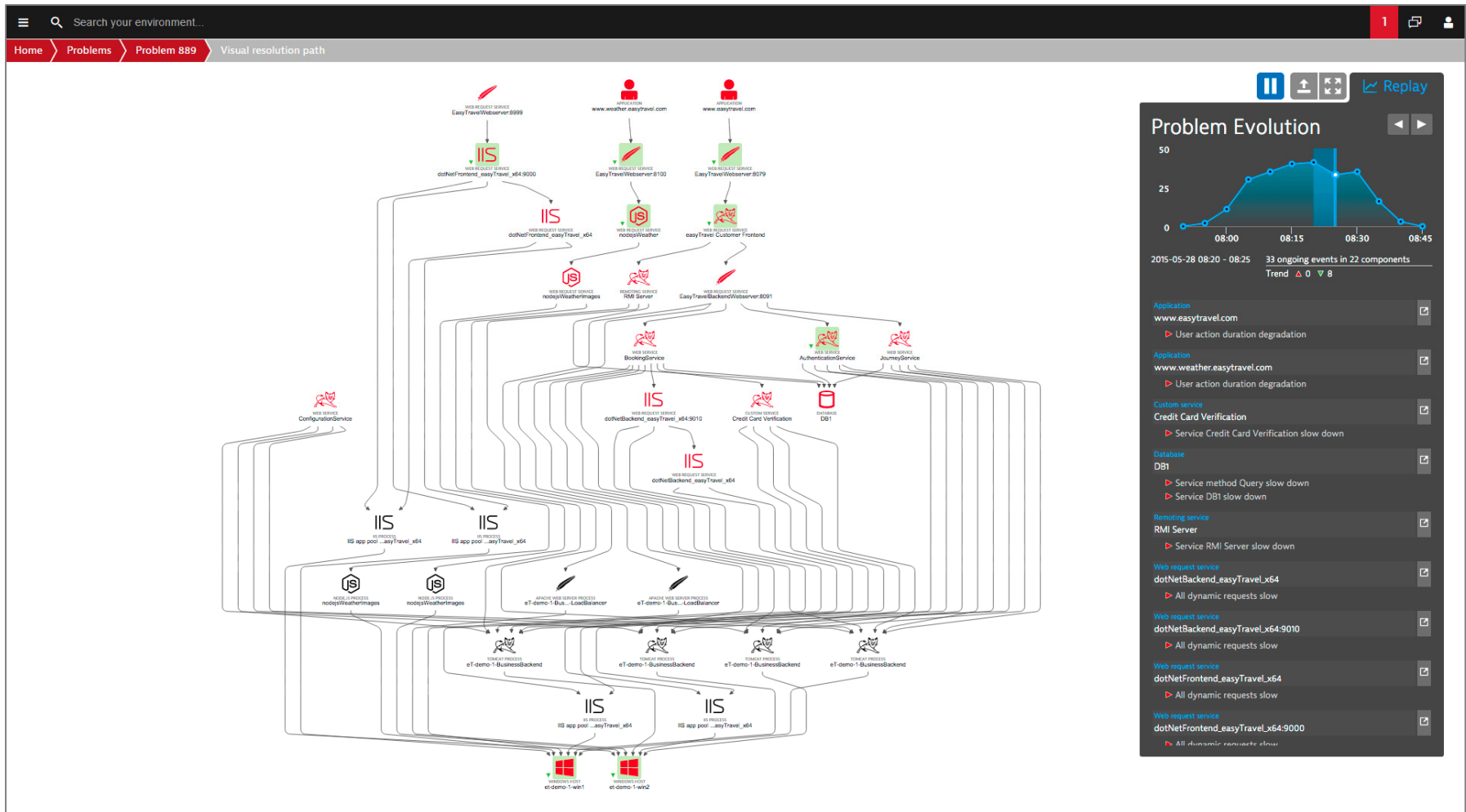
Cost - from last month

Shows how much cost in US Dollar was triggered last month by the active Amazon EC2 instances. Note: this does not include traffic and storage costs.



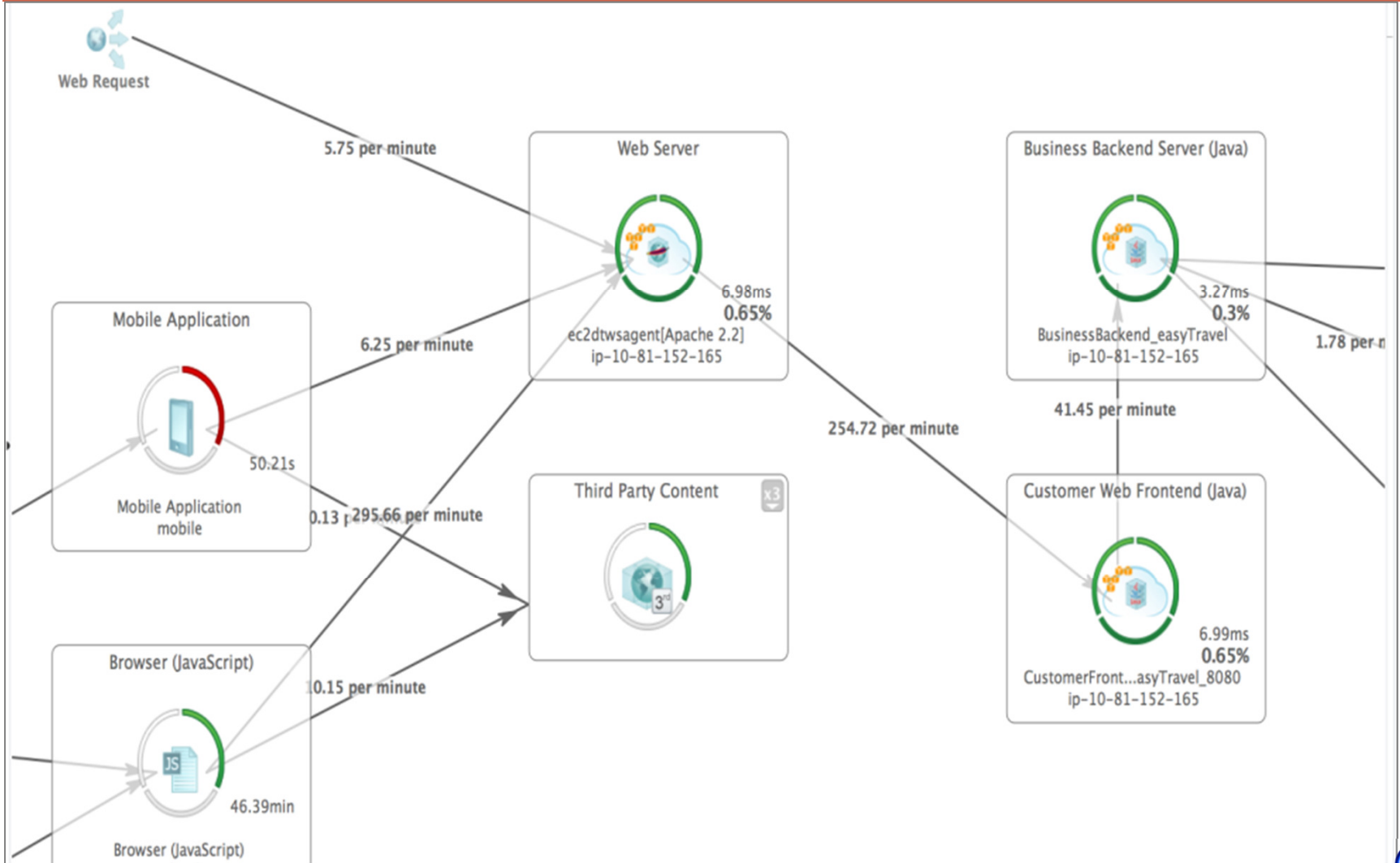
Dynatrace – Problem Evolution

53



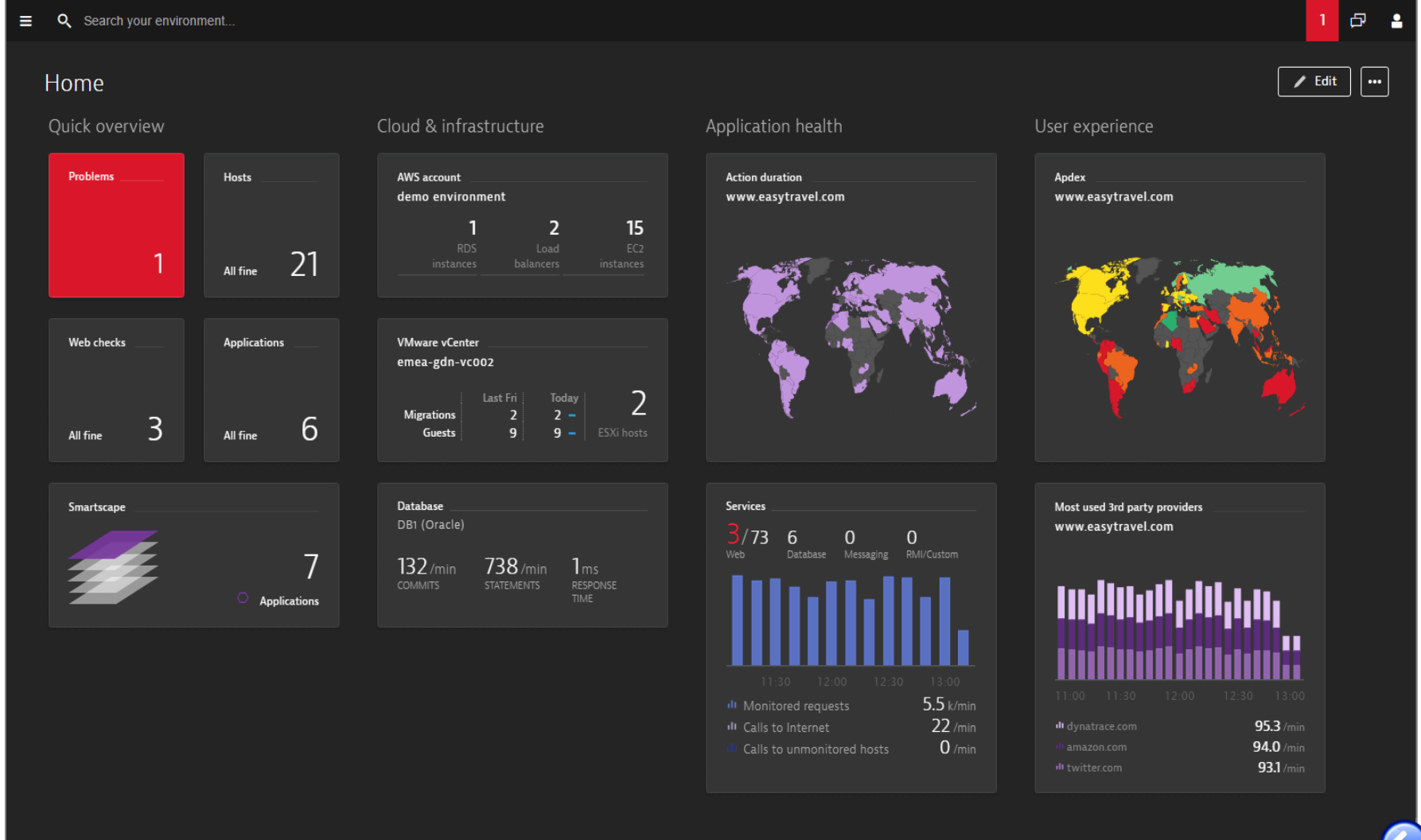
Dynatrace – Topology Visualization

54

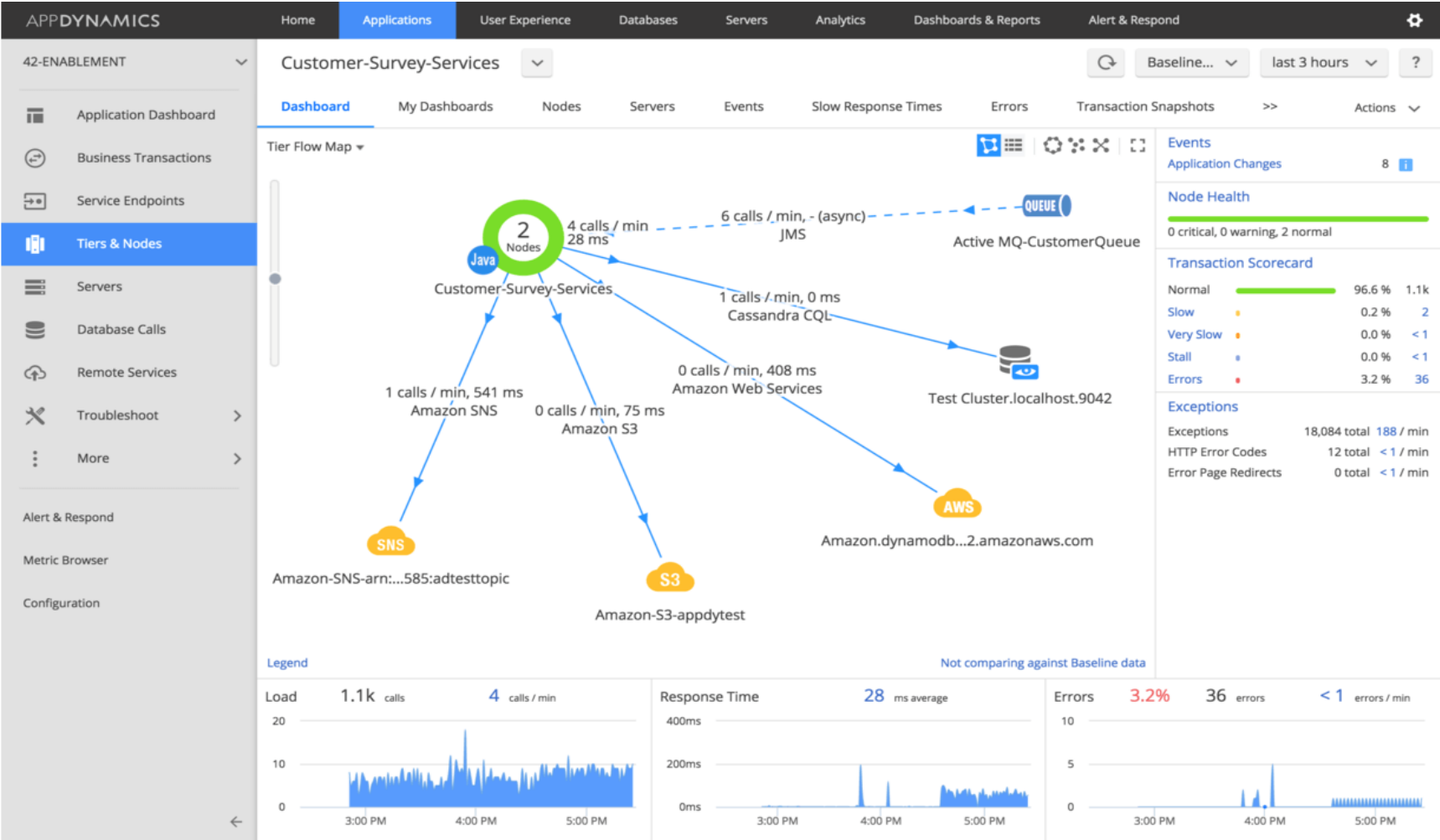


Dynatrace – Overview Dashboards

55

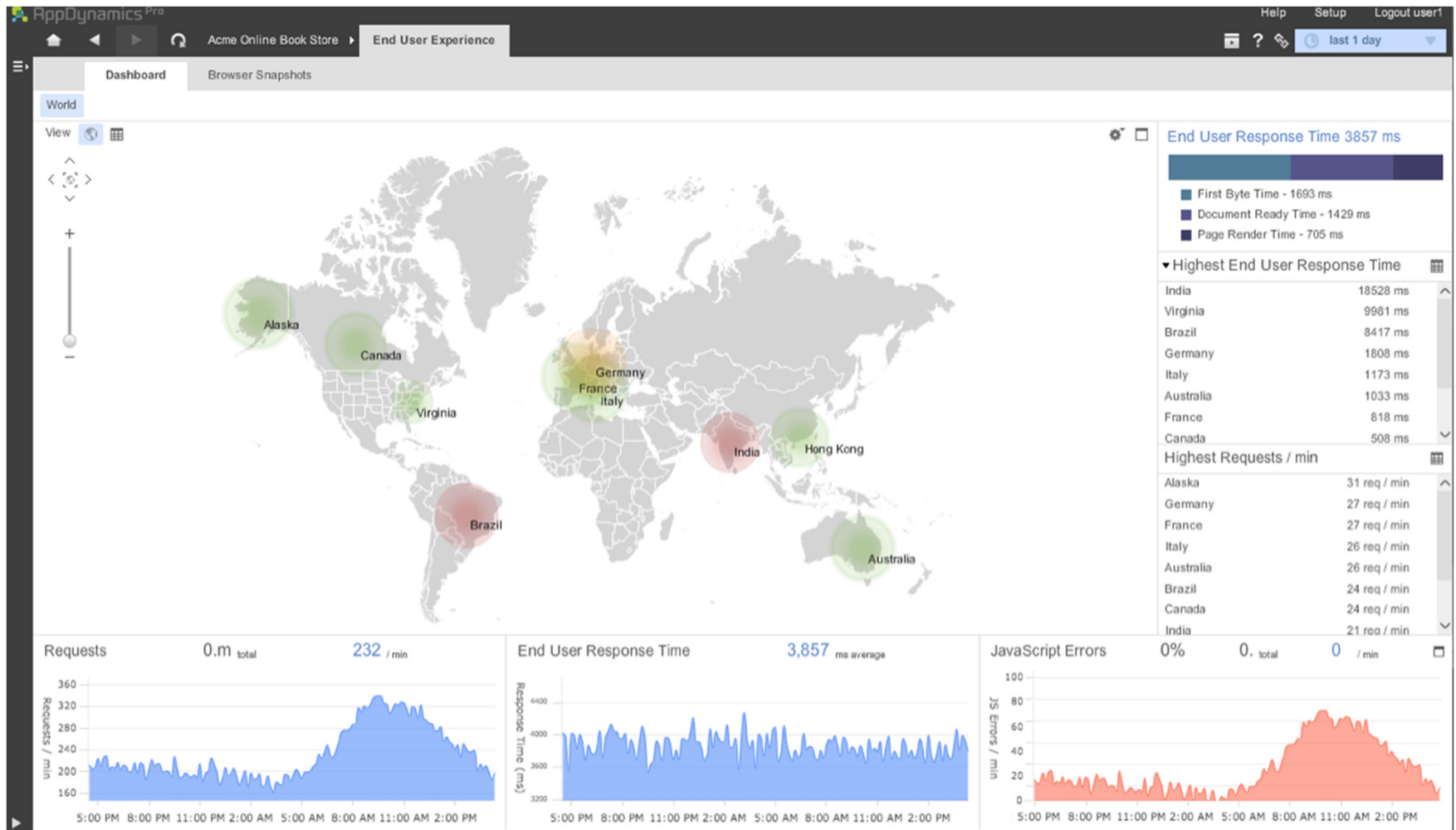


AppDynamics – Topology Visualization



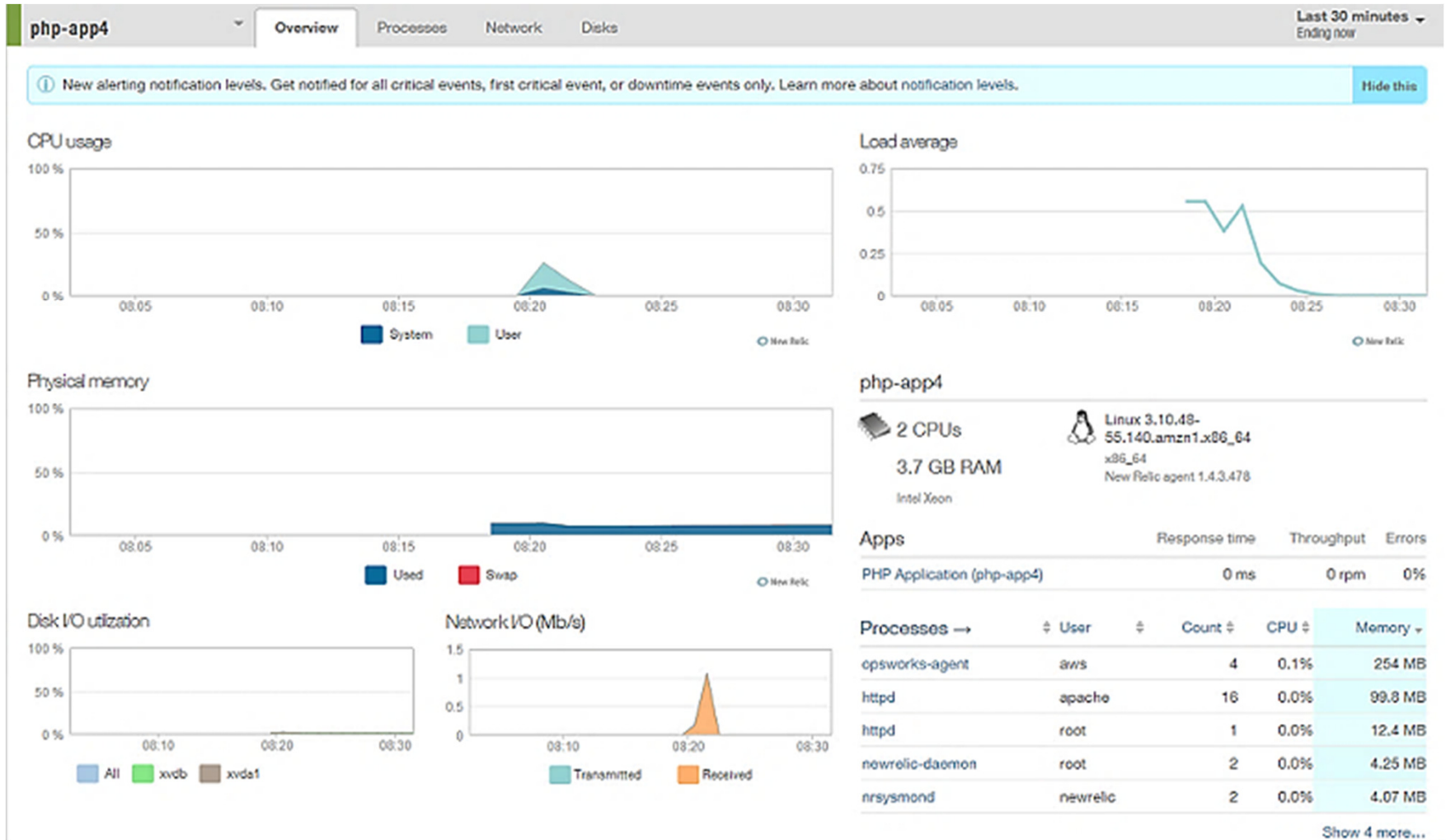
AppDynamics – User Experience

57



New Relic – Infrastructure Overview

58



New Relic – Instance Cost Monitoring

59

