



# Cisco Connect

Bled, 11. april 2018



AppDynamics

Josip Zimet

*Data Center, Cloud and managed Services SEE*

Cisco

 AppDynamics is now part of Cisco [Learn More](#) [About Cisco](#)

 [START FREE TRIAL](#)

# Gartner Ranks AppDynamics #1 in all 5 Use Cases in New Research

 AppDynamics is now part of Cisco [Learn More](#) [About Cisco](#)

[APPDYNAMICS](#) [PRODUCT](#) [SOLUTIONS](#) [RESOURCES](#) [CUSTOMERS](#) [COMPANY](#) [START FREE TRIAL](#)

[Support](#) [Sign in](#)

[Press Releases](#) >
   
[Press Release](#) a Leader in Gartner's 2016 Magic Quadrant for Application Performance Monitoring Suites for the Fifth Consecutive Year

[AppDynamics Named a Leader in Gartner's 2016 Magic Quadrant for Application Performance Monitoring Suites for the Fifth Consecutive Year](#)

*Positioned furthest for "Completeness of Vision" for third consecutive year*
  
*AppDynamics also recognized in new Gartner Critical Capabilities for Application Performance Monitoring Suites*

**SAN FRANCISCO – Dec. 27, 2016** – AppDynamics, the leading application intelligence company, today announced it has been recognized as a Leader in [Gartner's 2016 Magic Quadrant for Application Performance Monitoring Suites](#) for the fifth year in a row. As well as being named a leader, the company was positioned furthest to the right on the "Completeness of Vision" axis for the third consecutive year. Assessment criteria for vision include market understanding, marketing strategy, sales strategy, offering (product) strategy, business model, vertical/industry strategy, innovation and geographic strategy.

## Cisco Completes Acquisition of AppDynamics



**298** total shares
 [f](#)
[t](#)
[in](#)
[g+](#)
[e](#)

Together, AppDynamics and Cisco will give customers comprehensive visibility from code to consumer and everything in-between

**MARCH 22, 2017**

SAN JOSE, Calif. – March 22, 2017 – Cisco (NASDAQ: CSCO) today announced that it has completed the acquisition of AppDynamics, a leader in application and business monitoring. Together, Cisco and AppDynamics will deliver visibility across the network, data center, security and applications and drive digital transformation that improves customer experiences.

Cisco paid approximately \$3.7 billion in cash and assumed equity awards.

<https://finance.yahoo.com/news/cisco-appdynamics-recognized-leader-gartner-150900779.html>

## **Cisco (AppDynamics) Recognized as a Leader in Gartner's Magic Quadrant for the Sixth Consecutive Year and Placed Highest on the Ability to Execute Axis**



Business Wire March 21, 2018

SAN FRANCISCO--(BUSINESS WIRE)--

[AppDynamics](#), a [Cisco](#) company and the leader in application intelligence, today announced it has been named a leader in Gartner's 2018 Magic Quadrant for Application Performance Monitoring Suites for the sixth year in a row. In addition to being identified as a leader, the company also placed the highest among 15 vendors on the ability to execute axis. From our perspective, this recognition emphasizes AppDynamics' deep understanding of what companies need to be successful as they reshape their businesses to meet consumer expectations for sophisticated, always-on services. We also feel that it reflects Cisco's support in accelerating AppDynamics' time to market and ability to address the market on a global scale.

[Next Generation of Business iQ](#): Simplifies Complex Business Problems to Power Data-driven Decisions

[Perspica Technologies](#): Machine Learning Speeds Up Transformation of Data into Real-time Insights

[IoT & Network Monitoring](#): Provides Full Visibility from End-User Interactions on Connected Devices to Microservices Performance Across Multiple Clouds

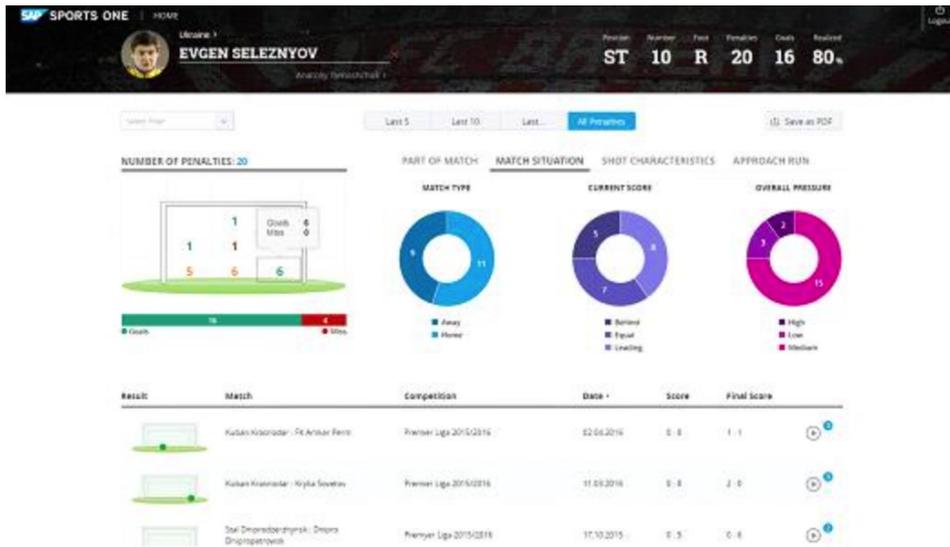
[Developer Toolkit](#): Improves App Team Collaboration with Development Lifecycle Transparency



On February 1, 2013, a TV series called *House of Cards* debuted on the video streaming service Netflix [1](#). It proved an immediate hit. Two years later, it has a [nine out of 10 rating](#) from more than 275,000 reviewers. Those numbers place it in the same category as *Avatar* and *The Sopranos*, but this success, some said, was guaranteed in advance. Why?

As [New York Times explained](#): [Netflix] already knew that a healthy share had streamed the work of David Fincher, the director of *The Social Network*, from beginning to end. And films featuring Kevin Spacey had always done well, as had the British version of “House of Cards.”

[https://thenextweb.com/insider/2016/03/20/data-inspires-creativity/#.tnw\\_UPbyyoz](https://thenextweb.com/insider/2016/03/20/data-inspires-creativity/#.tnw_UPbyyoz)



*‘In just 10 minutes, 10 players with three balls can produce over seven million data points,’ Oliver Bierhoff explained. ‘SAP HANA can process these in real time. With SAP, our team can analyze this huge amount of data to customize training and prepare for the next match.’*

**The team led by the captain Philipp Lahm focused on their passing speed. In 2010 World Cup the average ball possession was 3.4 seconds per touch and by using the “Match Insights” framework they scaled it down to 1.1 seconds. The improvement was very apparent from their 7-1 victory over the mighty Brazil.**

<https://confluo.wordpress.com/2016/02/18/sap-had-a-hand-in-germanys-world-cup-title/>



Budgeted for digital transformation

\$2.1T

Estimated 2019 WW Enterprise spend on digital transformation technologies

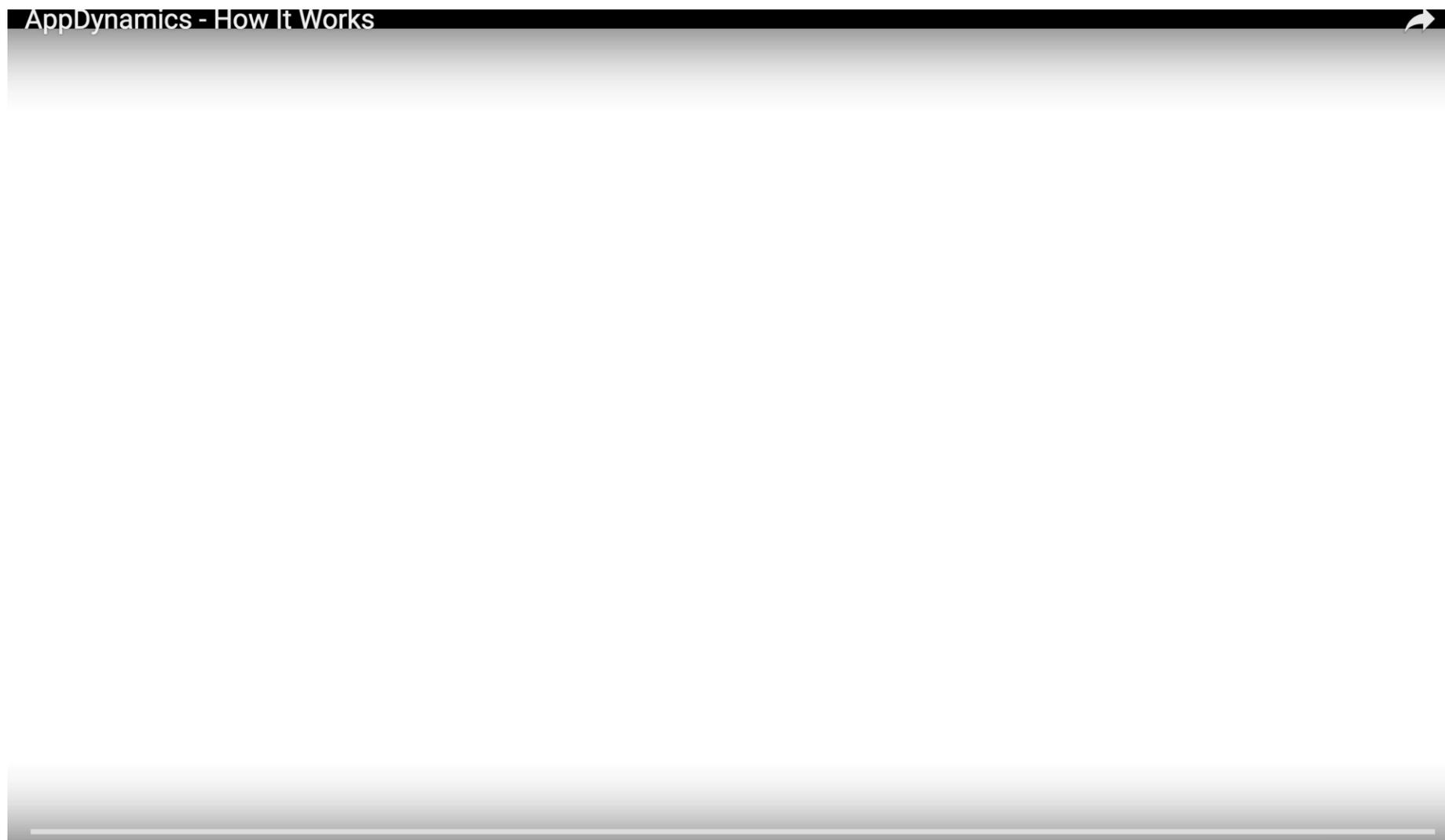
Enterprises lack coherent strategies

73%

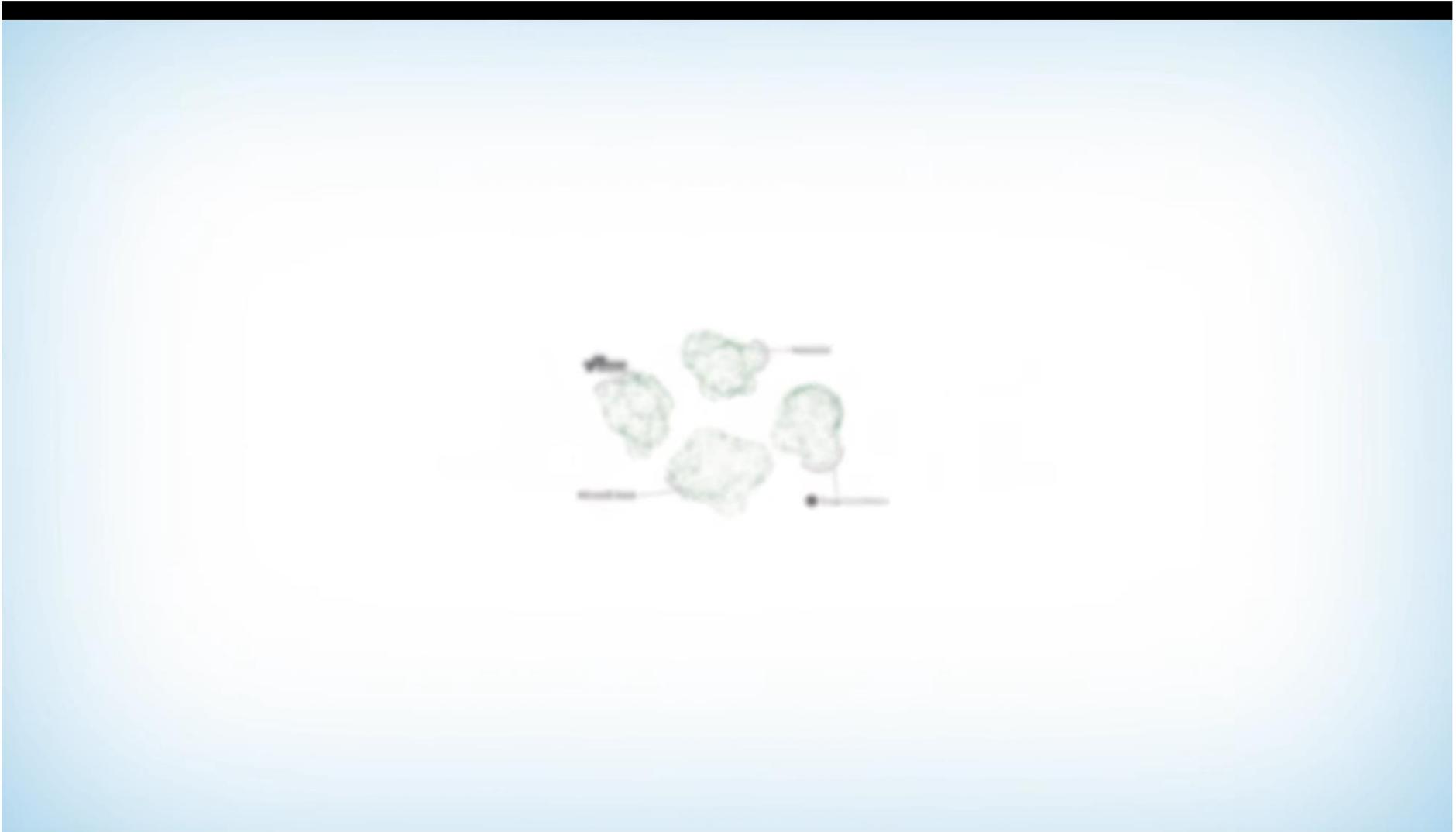
Enterprises polled that didn't have a coherent digital strategy to create customer value



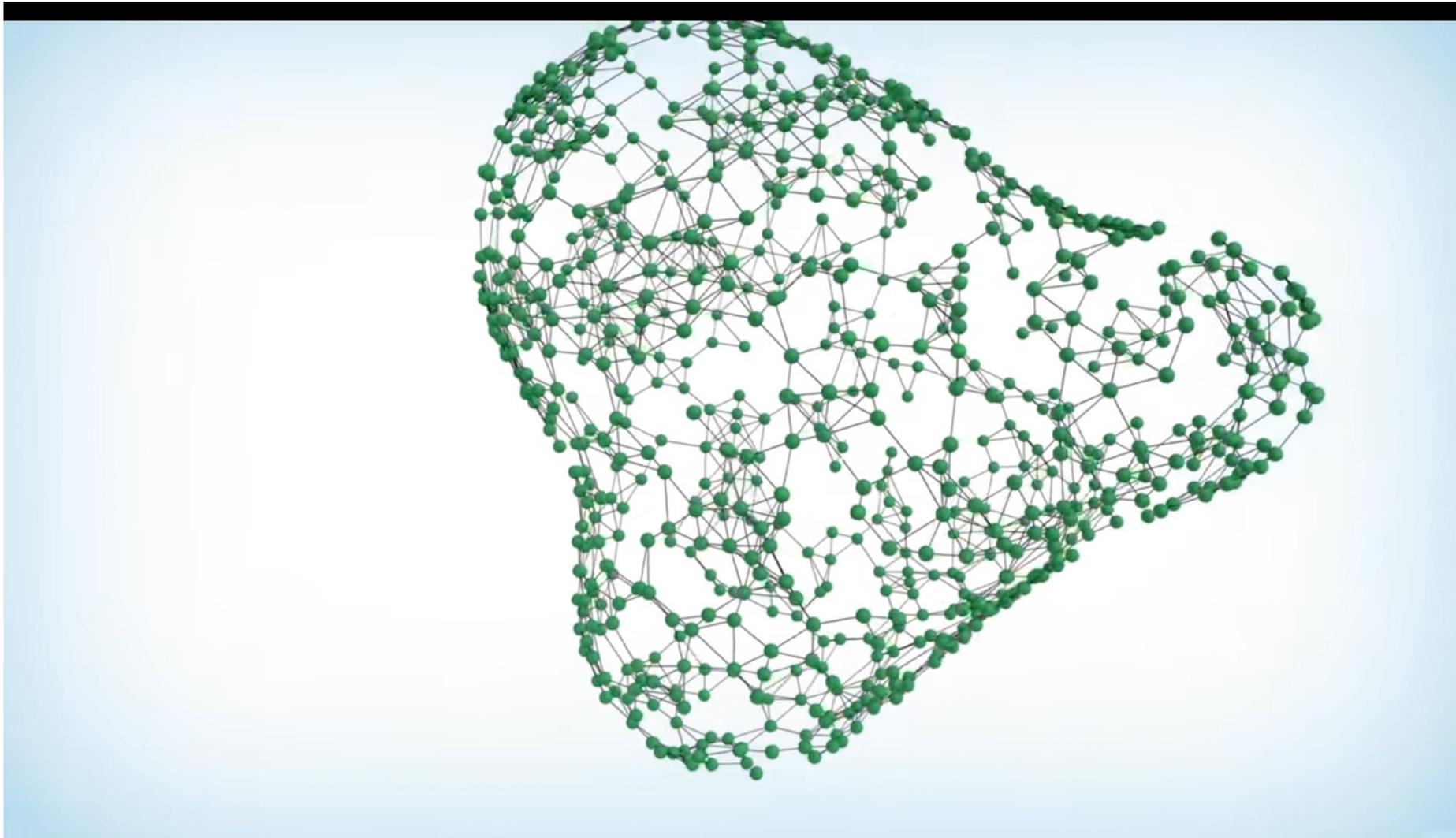
# Agents deployed on any cloud



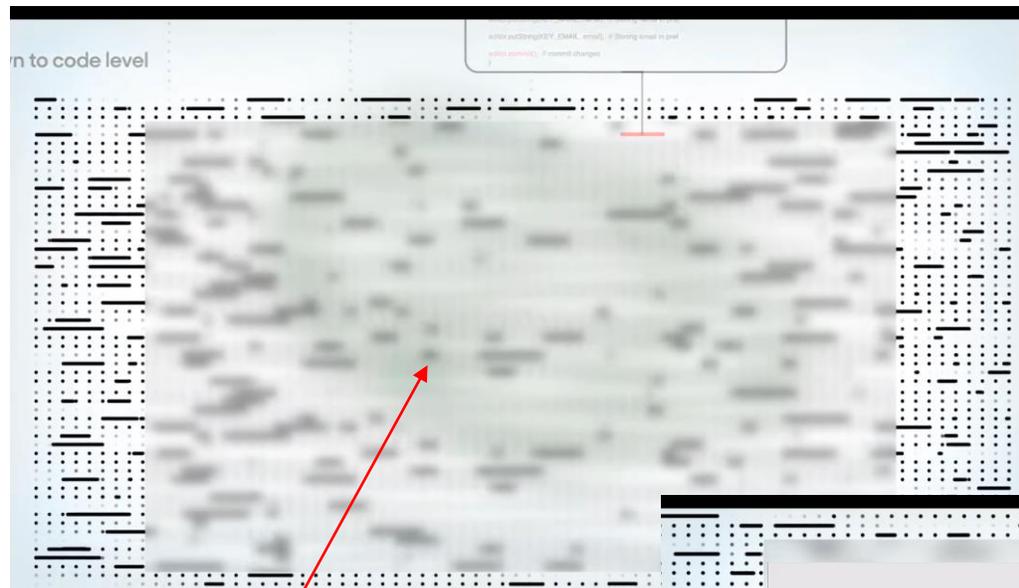
... watch all line of code ...



... And automatically discover business transactions (login) ...

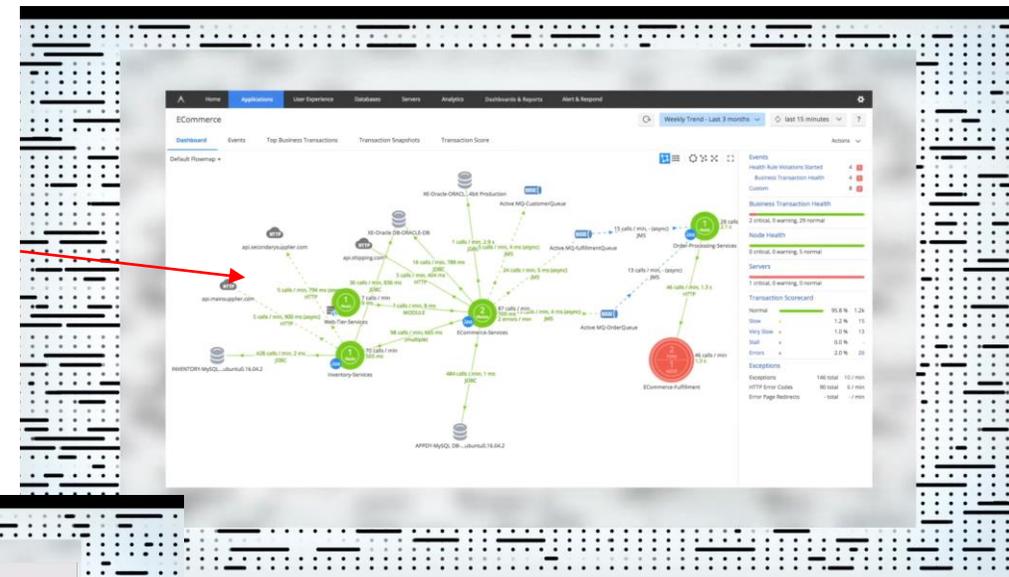


# Developers, operations professional and business owners ...

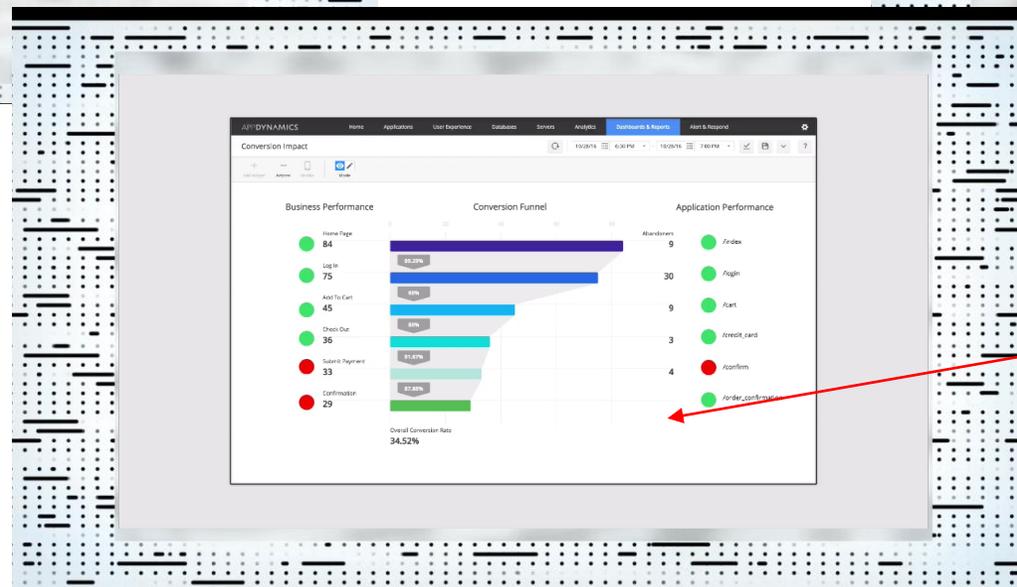


**Developers** are able to quickly diagnose and determine root cause by viewing stack traces, database calls and correlated log statements

**Operations Professionals** can focus on end to end business transaction instead of monitoring individual services



**Business owners** can monitor real time business performance metrics such as revenue and conversion rate driving better business outcome



# https://www.appdynamics.com/free-trial/

APPDYNAMICS Sign in

Get started quickly with AppDynamics

Start your free 15-day trial now, easily, with no credit card required.

Designed for production and pre-production environments, AppDynamics gives you visibility into your entire application topology from a single pane of glass. Instrument your own application or use our sample app.

- Application Performance Management**  
Monitor and manage end-to-end performance of complex distributed applications.
- End User Monitoring**  
Deliver a better user experience with Mobile and Browser Real-User Monitoring and Browser Synthetic Monitoring.
- Infrastructure Visibility**  
Unlock deeper insights by correlating server and database performance with application performance.
- Real-time Business Monitoring**  
Get real-time business awareness into IT operations, customer experience, and business outcomes with Transaction, Log, Browser, and Mobile Analytics.

Sign Up for Free

in G+  f

First Name

Last Name

Company Email

Company

Job Title

Phone Number

Create Password

SaaS  On-Premise

I agree to the [Evaluation Terms and Privacy Policy](#).

[CREATE A FREE ACCOUNT](#)

Already have an account? [Sign In](#).

Copyright © 2009 - 2017 AppDynamics. All Rights Reserved. [English \(USA\)](#) [Privacy Policy](#) [Cookie Policy](#) [Terms of Use](#) [Security](#)





Java



.NET



PHP



Node.js



C++



Python



Apache Web Server



GoLang



iOS



ANDROID



XAMARIN

# AppD Platform

Works with:



## AppDynamics Platform

### Unified Monitoring



Application Performance Management



End User Monitoring



Infrastructure Visibility

### App iQ Platform

### Enterprise iQ



Map iQ  
End-to-End Business Transaction Tracing



Baseline iQ  
Machine Learning - Dynamic Baseline



Diagnostic iQ  
Code-Level Diagnostics With Low Overhead



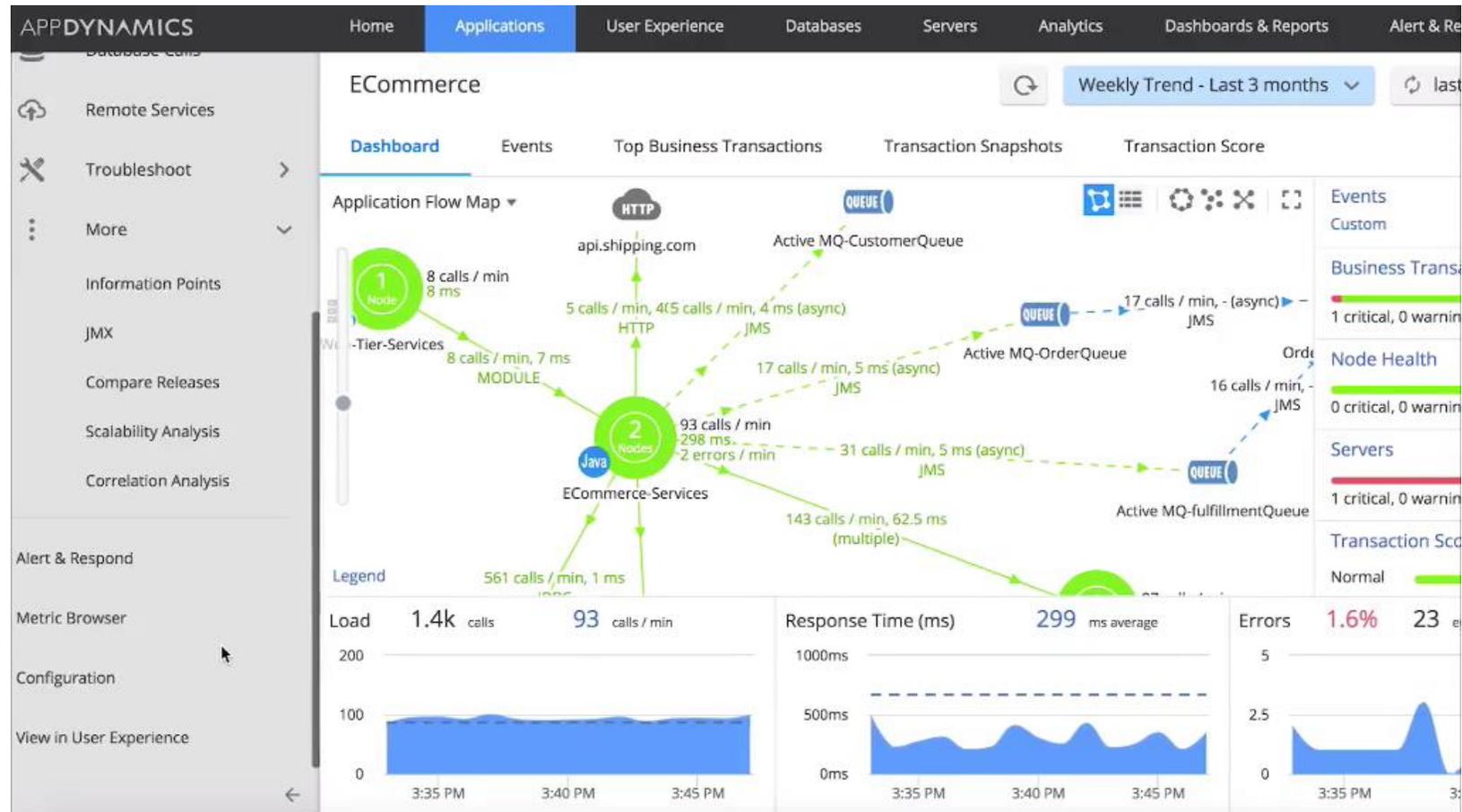
Business iQ  
Track, Baseline and Alert on Business Metrics

Signal iQ

# Smart camera



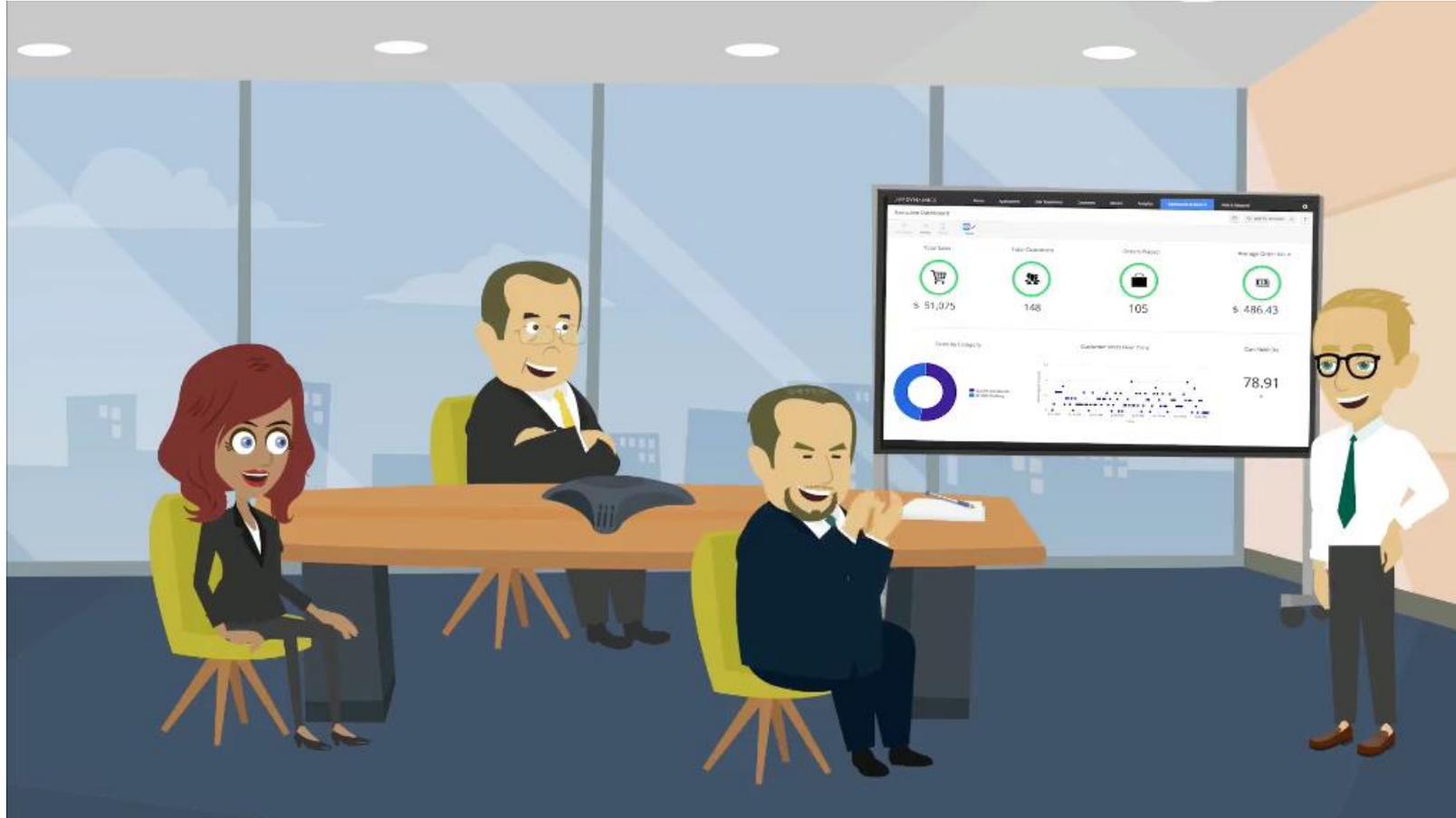
# Development Mode



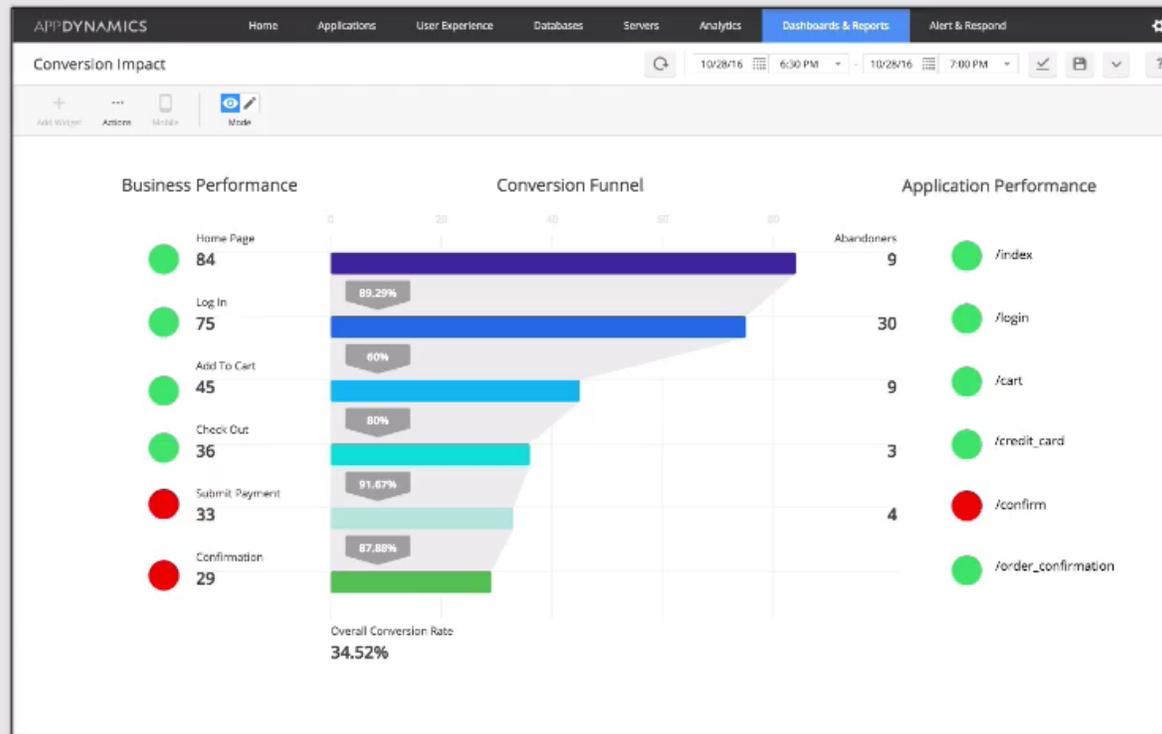
# Mean Time Business Awareness



# Mean Time Business Awareness



# BusinessIQ



# APPDYNAMICS



## Map iQ

Discovery of your application topology



## Baseline iQ

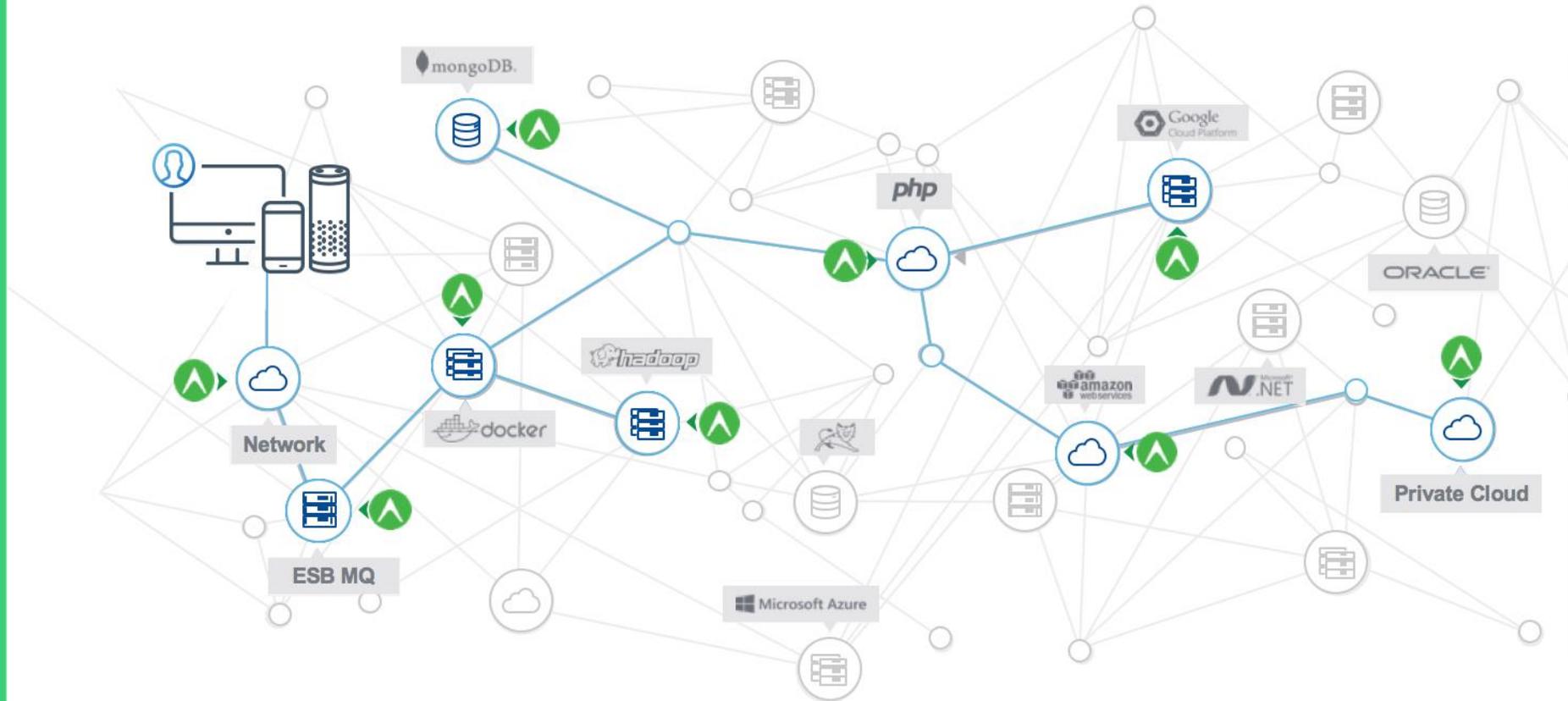
Dynamic baselines and anomaly detection



## Diagnostic iQ

Self-learning diagnostics

### Is your application **fast**?



# APPDYNAMICS



## Map iQ

Discovery of your application topology



## Baseline iQ

Dynamic baselines and anomaly detection

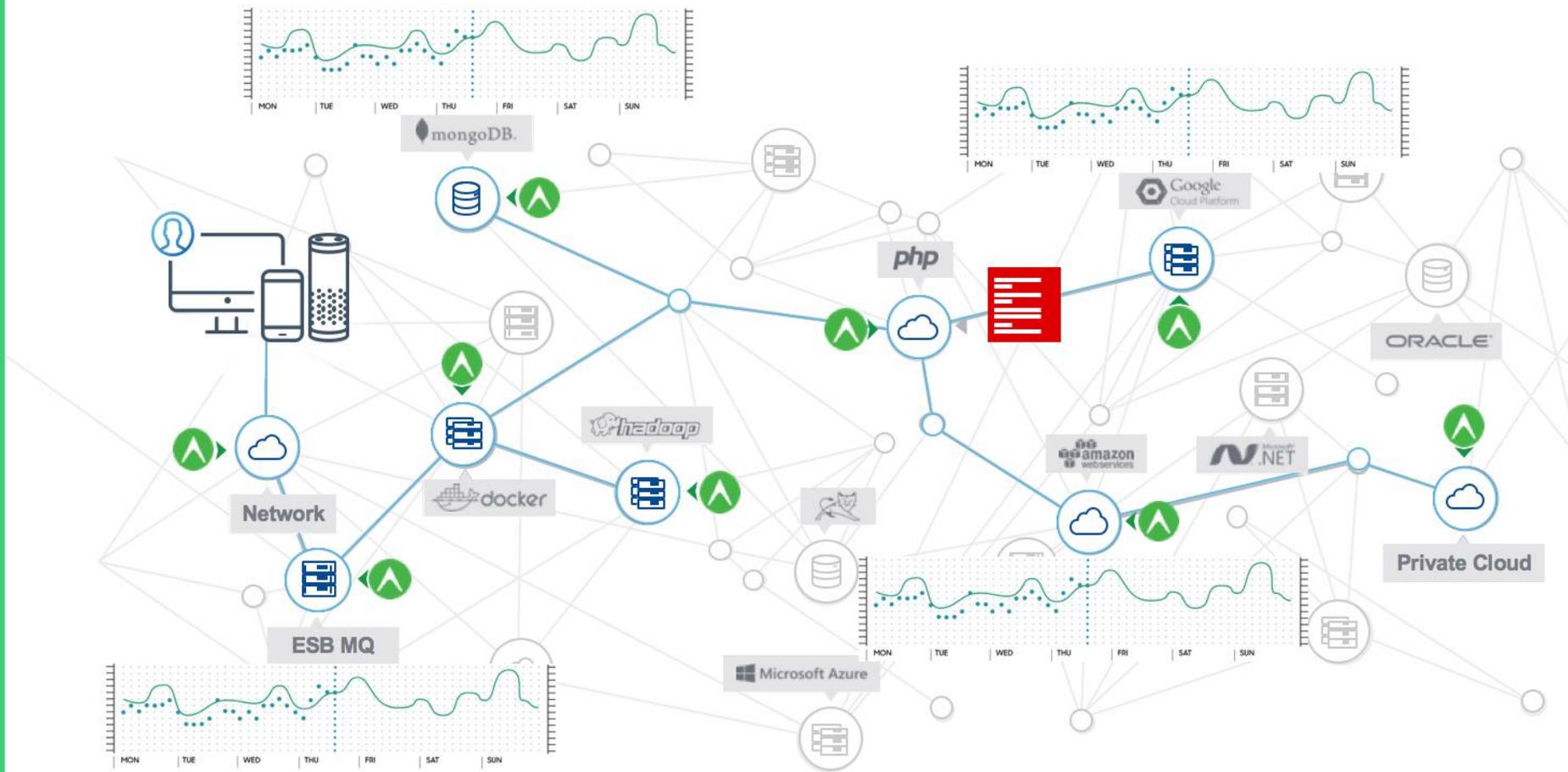


## Diagnostic iQ

Self-learning diagnostics



# Is your application **fast**?



# APPDYNAMICS



## Map iQ

Discovery of your application topology



## Baseline iQ

Dynamic baselines and anomaly detection

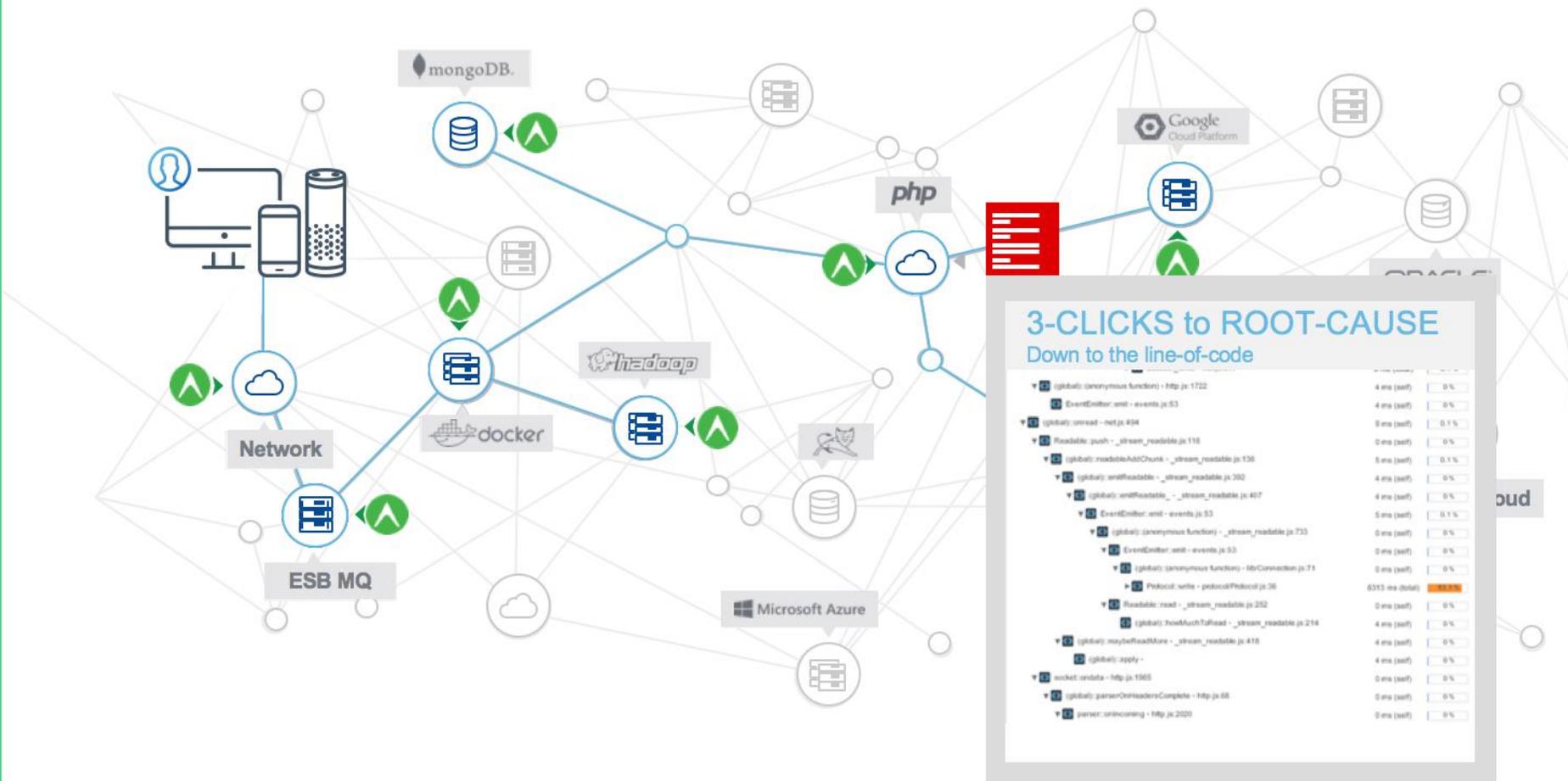


## Diagnostic iQ

Self-learning diagnostics



# Is your application fast?

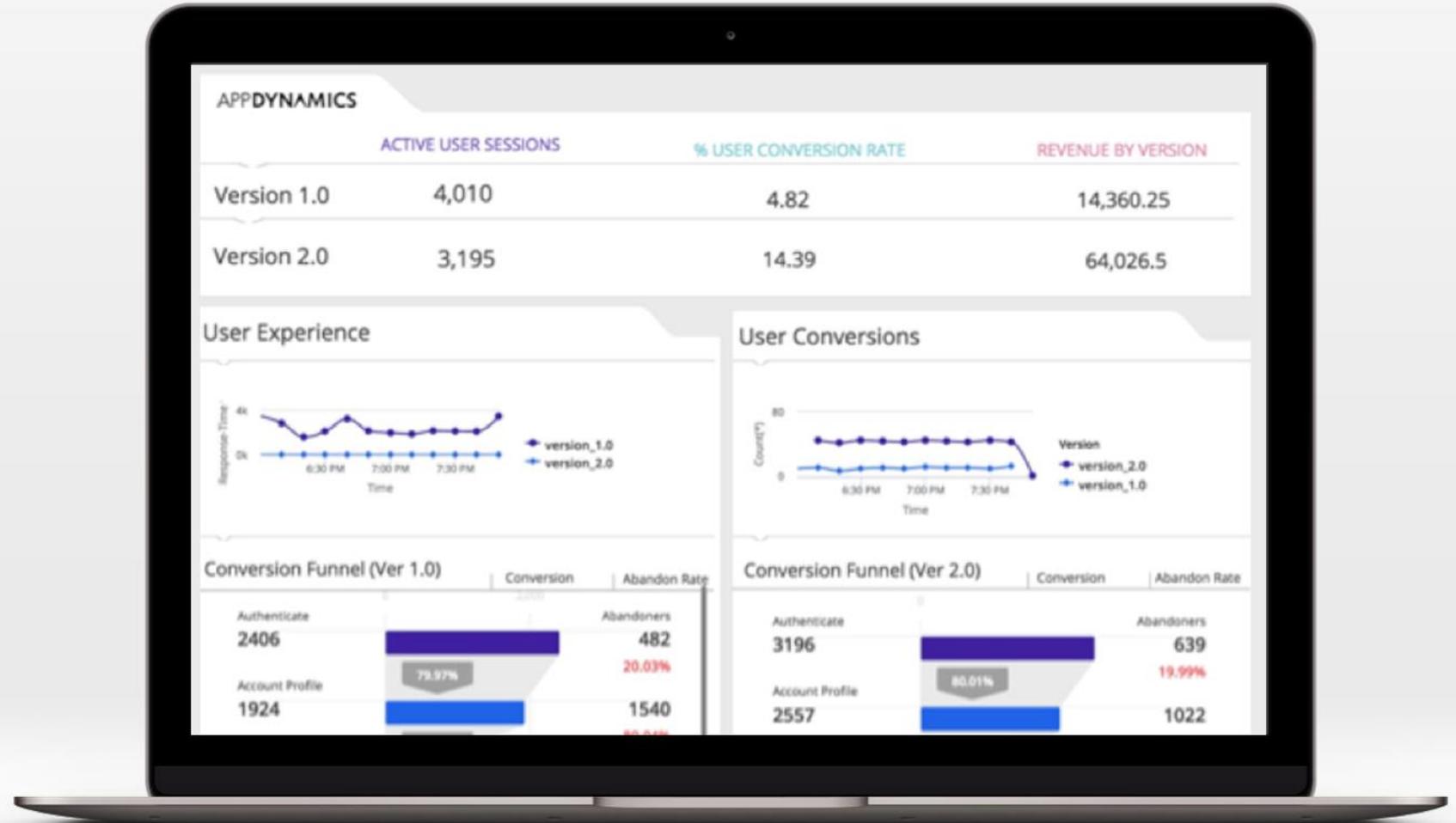




## Business iQ

- Increase business velocity
- Business centric dashboards
- Real-time business monitoring

# Is your application **useful**?

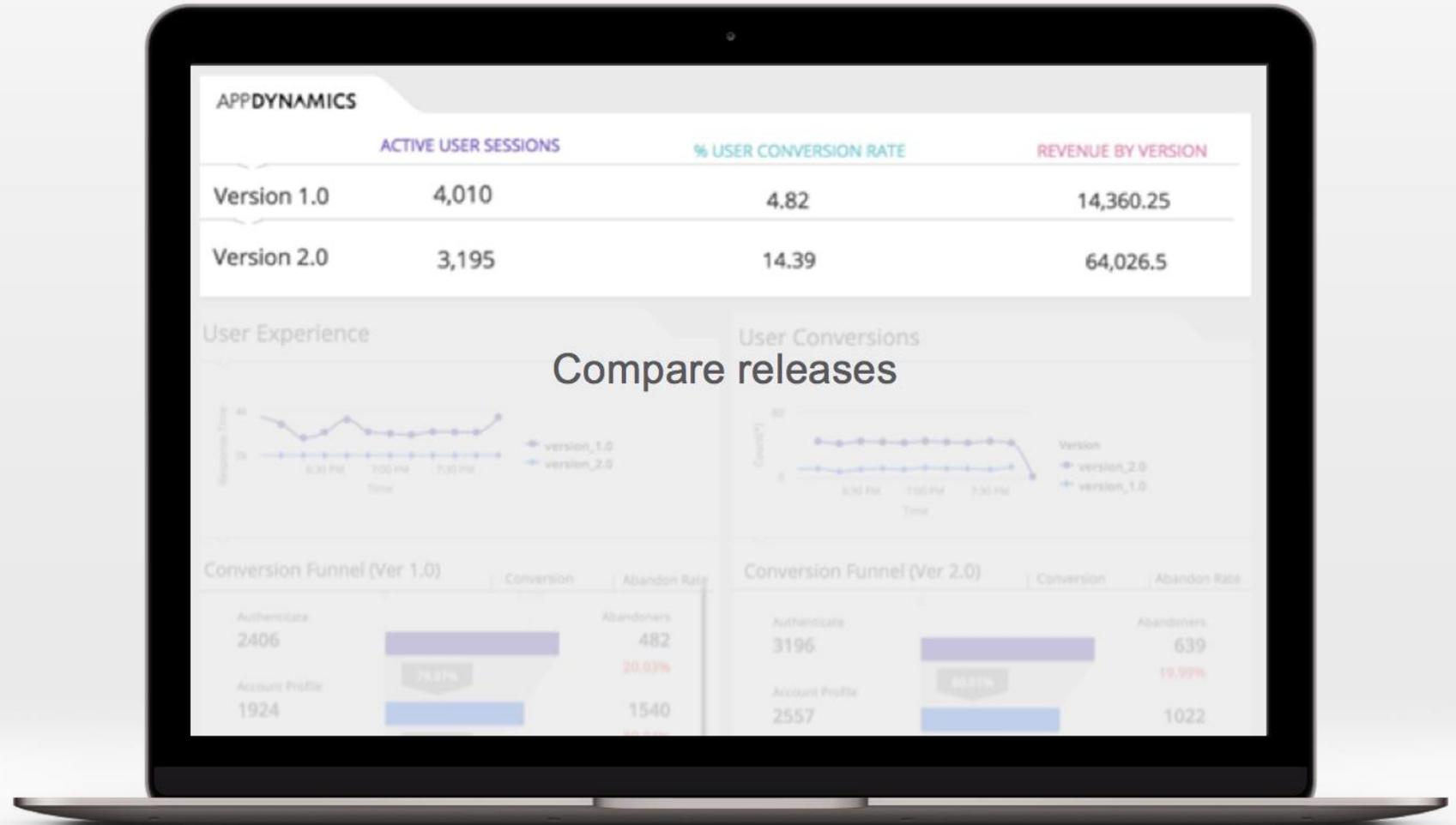




Business iQ

- Increase business velocity
- Business centric dashboards
- Real-time business monitoring

# Is your application **useful**?

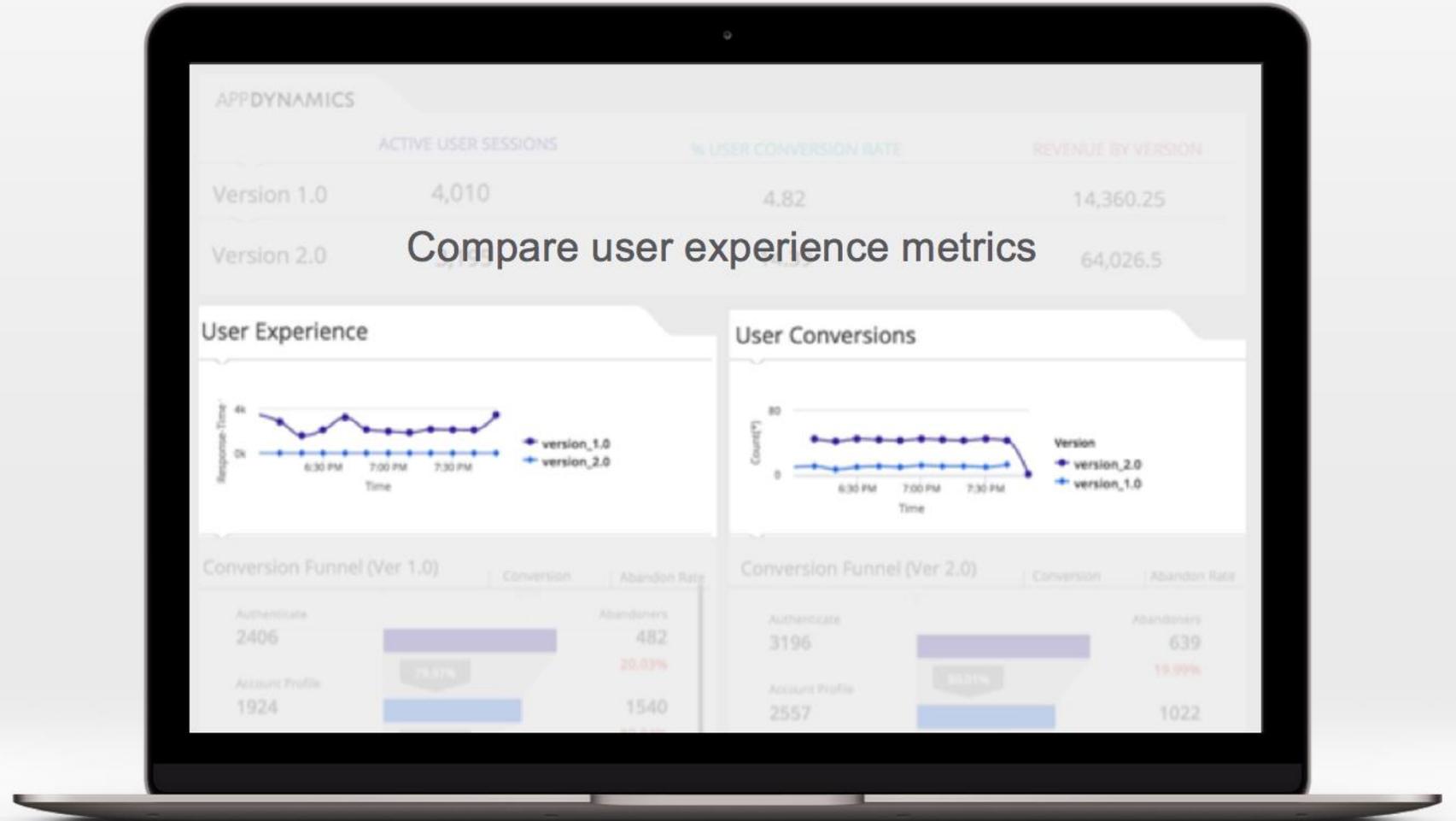




## Business iQ

- Increase business velocity
- Business centric dashboards
- Real-time business monitoring

# Is your application **useful**?

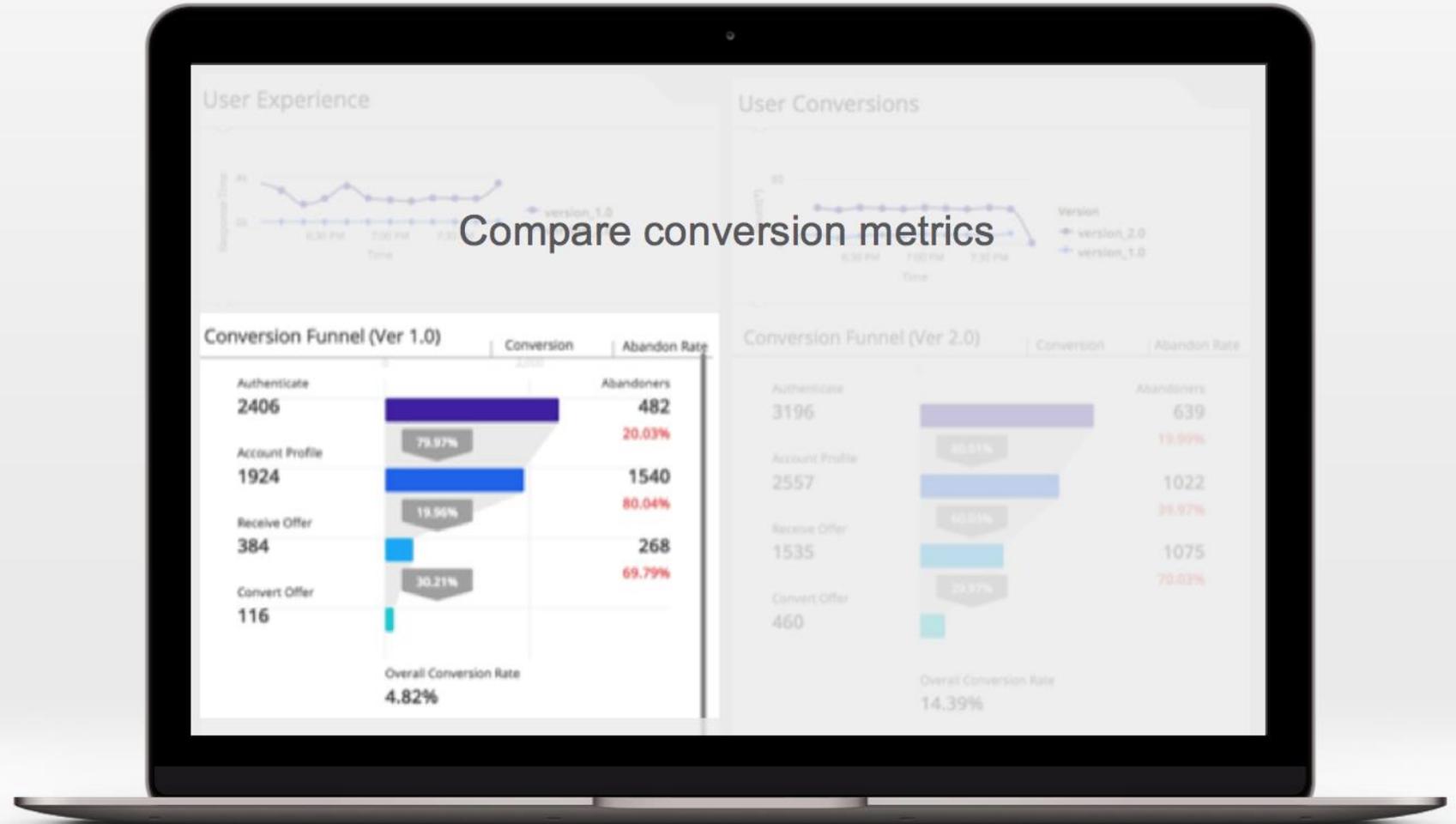




## Business IQ

- Increase business velocity
- Business centric dashboards
- Real-time business monitoring

# Is your application **useful**?

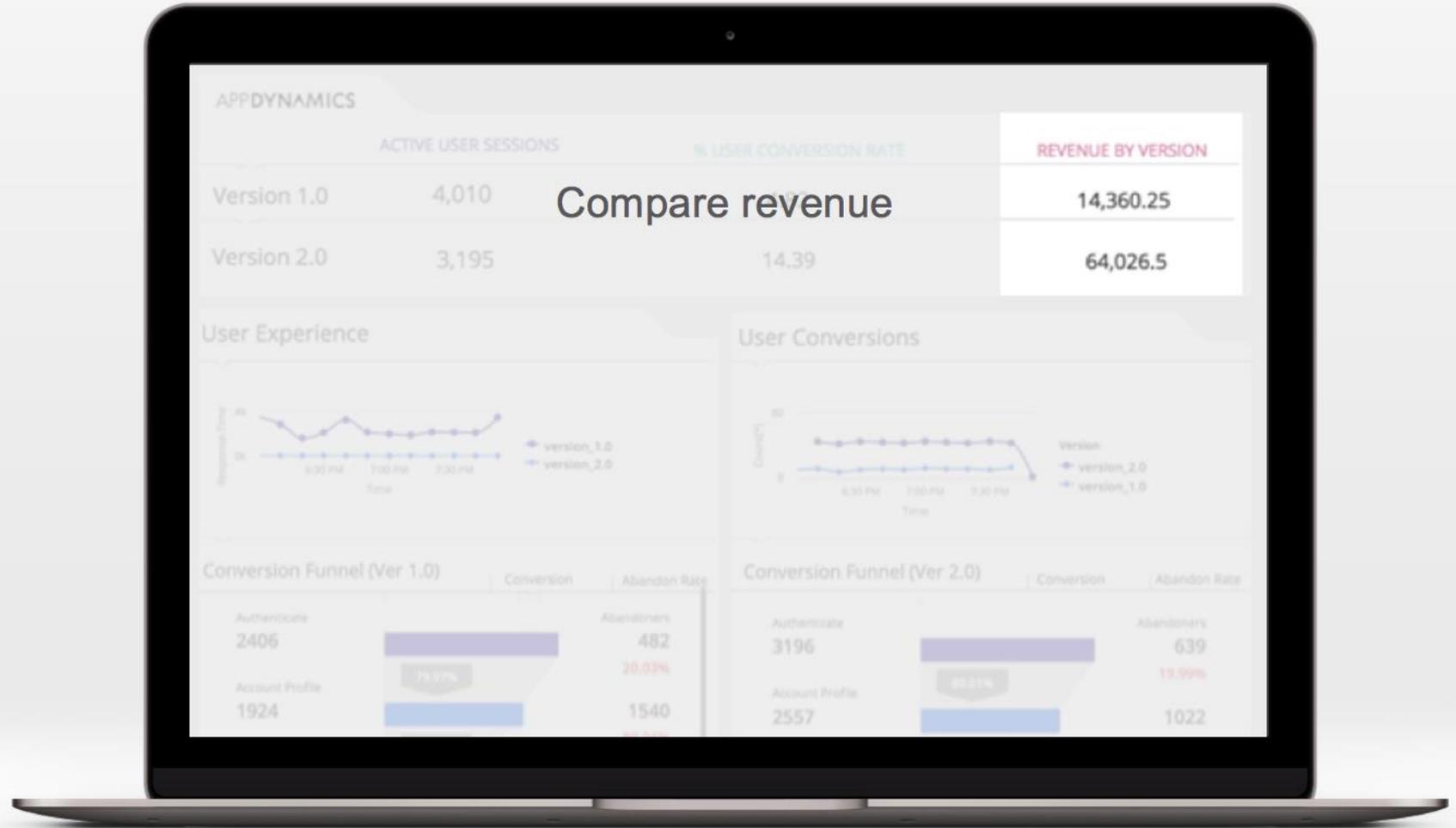




Business iQ

- Increase business velocity
- Business centric dashboards
- Real-time business monitoring

# Is your application **useful**?

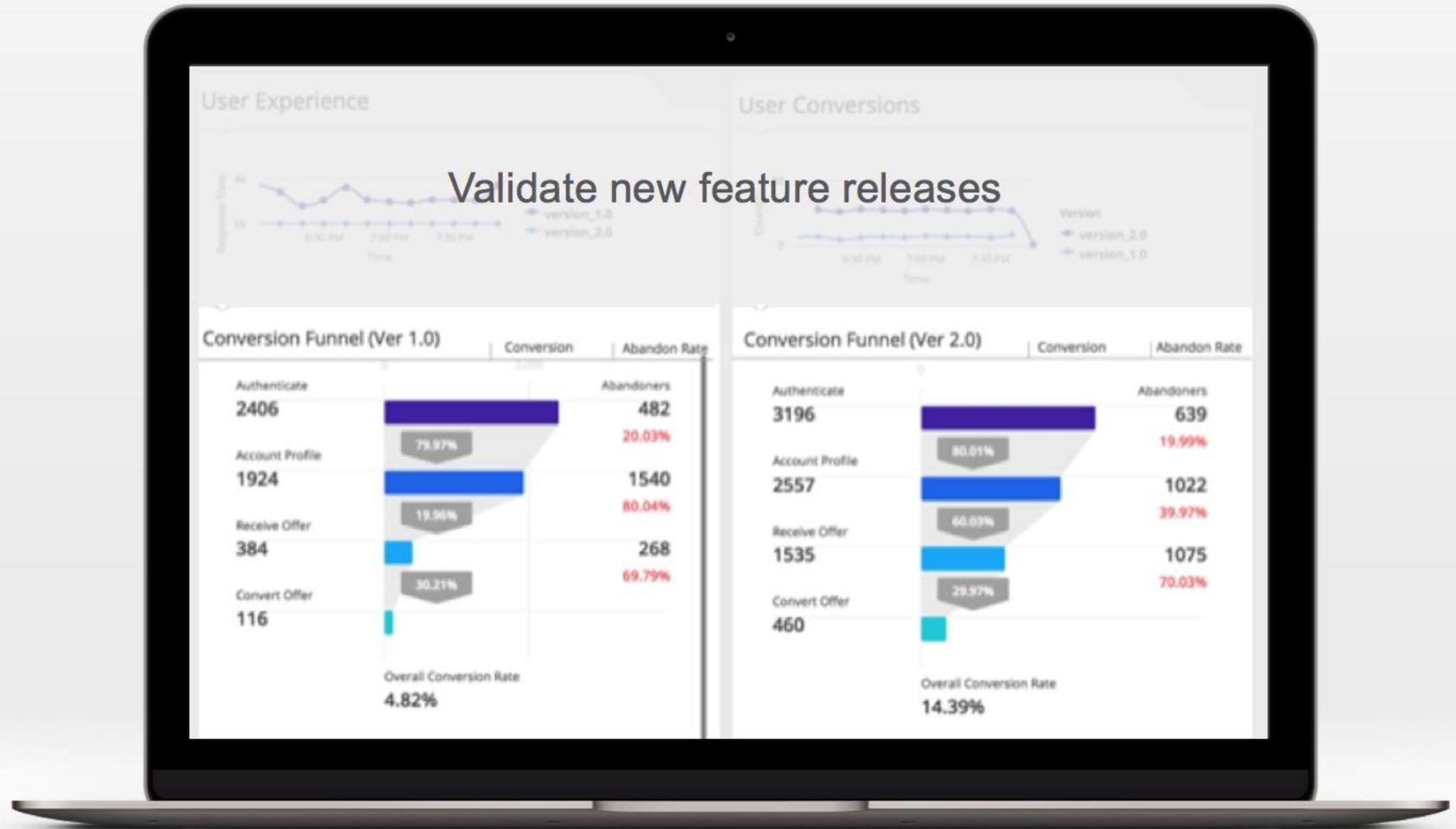




## Business IQ

- Increase business velocity
- Business centric dashboards
- Real-time business monitoring

# Is your application **useful**?





Whitelist  
Policy



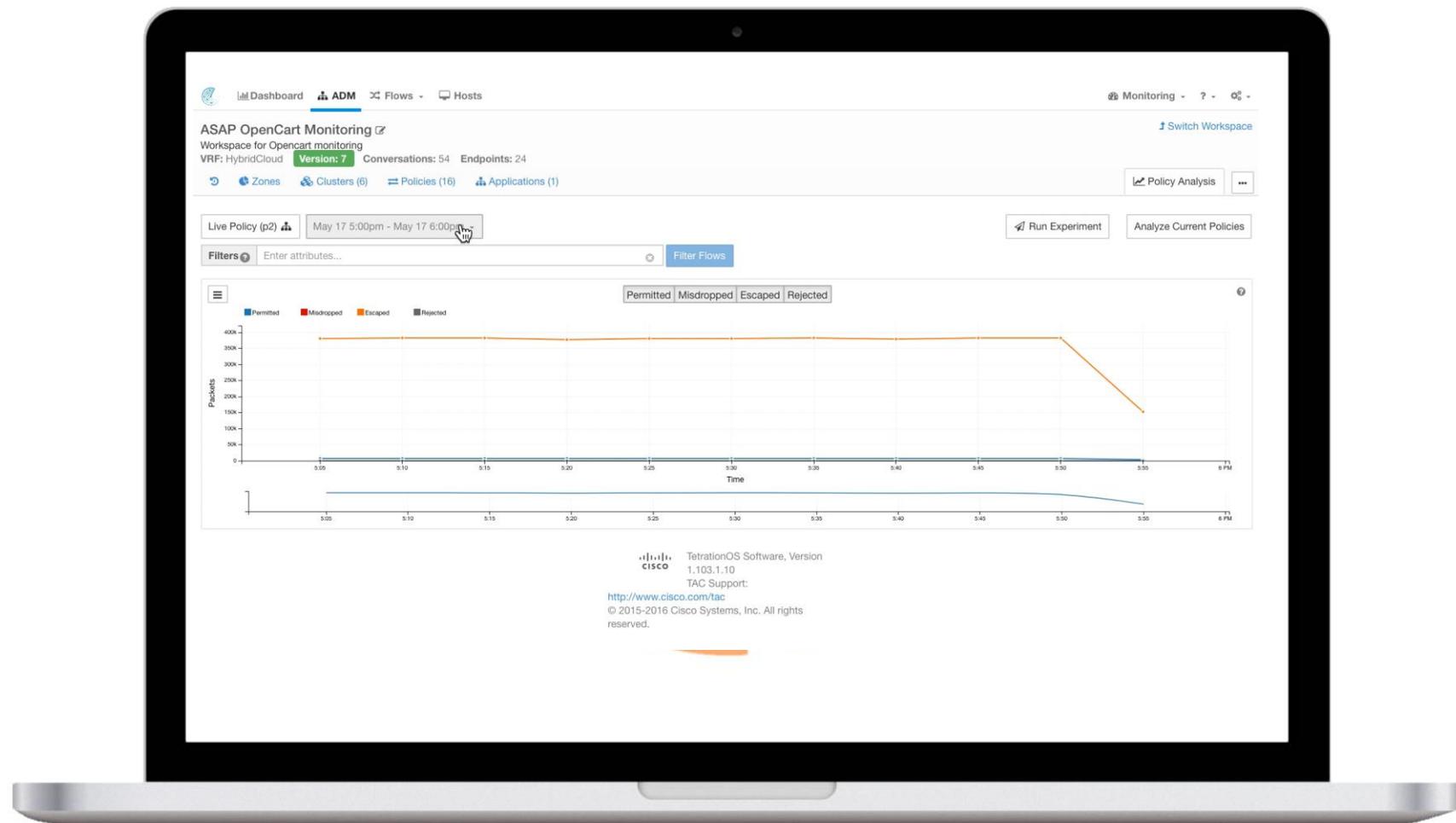
Application  
Segmentation



Policy  
Compliance

# Is your application **secure**?

## Auto Generate Whitelist Policies



Tetration Analytics™



Whitelist  
Policy



Application  
Segmentation

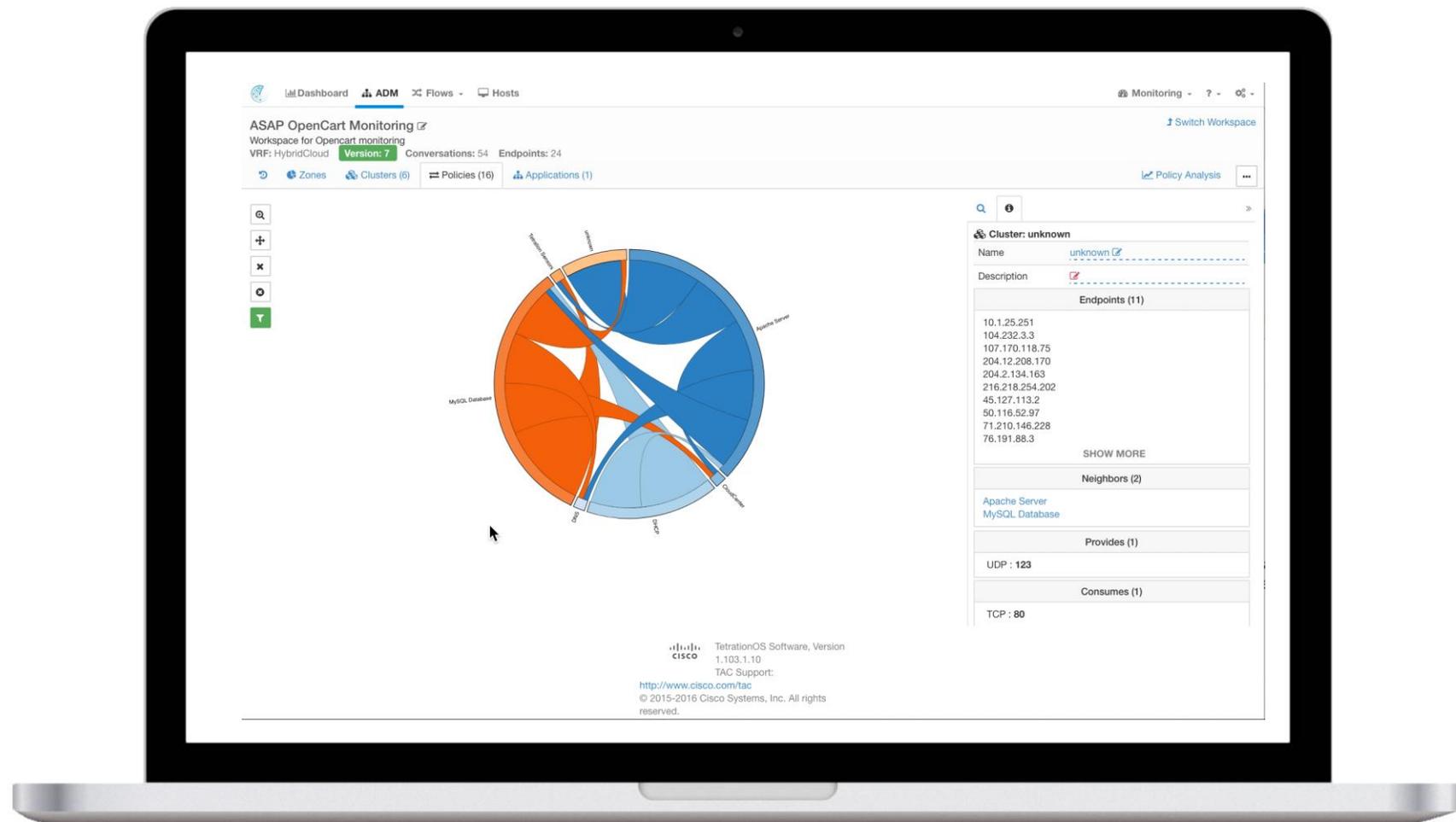


Policy  
Compliance



# Is your application **secure**?

## Auto Generate Whitelist Policies





Model, Deploy and Monitor  
Applications Across Clouds

**Cisco HyperFlex Multicloud Ecosystem**

# HyperFlex - Hyper converged infrastructure

## Quickly deploy HyperFlex with HyperFlex Installer

The screenshot displays the Cisco HyperFlex Installer web interface. At the top, the Cisco logo and "HyperFlex Installer" are visible. The interface is divided into "Progress" and "Summary" sections. The "Progress" section shows a horizontal timeline with eight steps: Start, Validations, UCSM Configuration, Hypervisor Configuration, Deploy Validation, Deploy, Create Validation, and Cluster Creation. All steps are marked with green checkmarks, indicating successful completion. Below the timeline, a green checkmark and the text "Cluster Creation Successful" are displayed, along with a "View Summary" button. The "Summary" section on the right contains a "Configuration" panel with "Credentials" and "Server Selection" details. The "Credentials" section lists UCS Manager Host Name (192.168.165.17), User Name (admin), vCenter Server (192.168.165.19), User Name (Administrator@vsphere.local), and Admin User Name (root). The "Server Selection" section lists eight servers (Server 1-8) with their respective IDs and models. At the bottom of the configuration panel is an "Edit Configuration" button. The main content area shows a detailed view of the "Cluster Creation" process, including an overall status of "Succeeded" and a list of tasks such as "Configuring Cluster Resource Manager", "Preparing Storage Cluster", "updateClusterSEDStatus", and "Configuring NTP Services" for three servers (172.31.165.23, 172.31.165.25, and 172.31.165.25).

HyperFlex Installer

Progress Summary

Start Validations UCSM Configuration Hypervisor Configuration Deploy Validation Deploy Create Validation Cluster Creation

Cluster Creation Successful [View Summary](#)

Cluster Creation

Cluster Creation - Overall **Succeeded**

- Configuring Cluster Resource Manager Cluster Resource Management
- Preparing Storage Cluster Storage Cluster
- updateClusterSEDStatus

172.31.165.23 **Succeeded**

- Configuring NTP Services

172.31.165.25 **Succeeded**

- Configuring NTP Services

Configuration

Credentials

UCS Manager Host Name 192.168.165.17

User Name admin

vCenter Server 192.168.165.19

User Name Administrator@vsphere.local

Admin User Name root

Server Selection

Server 8 FCH1949V23G / HX220C-M4S

Server 2 FCH1949V2QJ / HX220C-M4S

Server 3 FCH1949V2TH / HX220C-M4S

Server 1 FCH1951V07E / HX220C-M4S

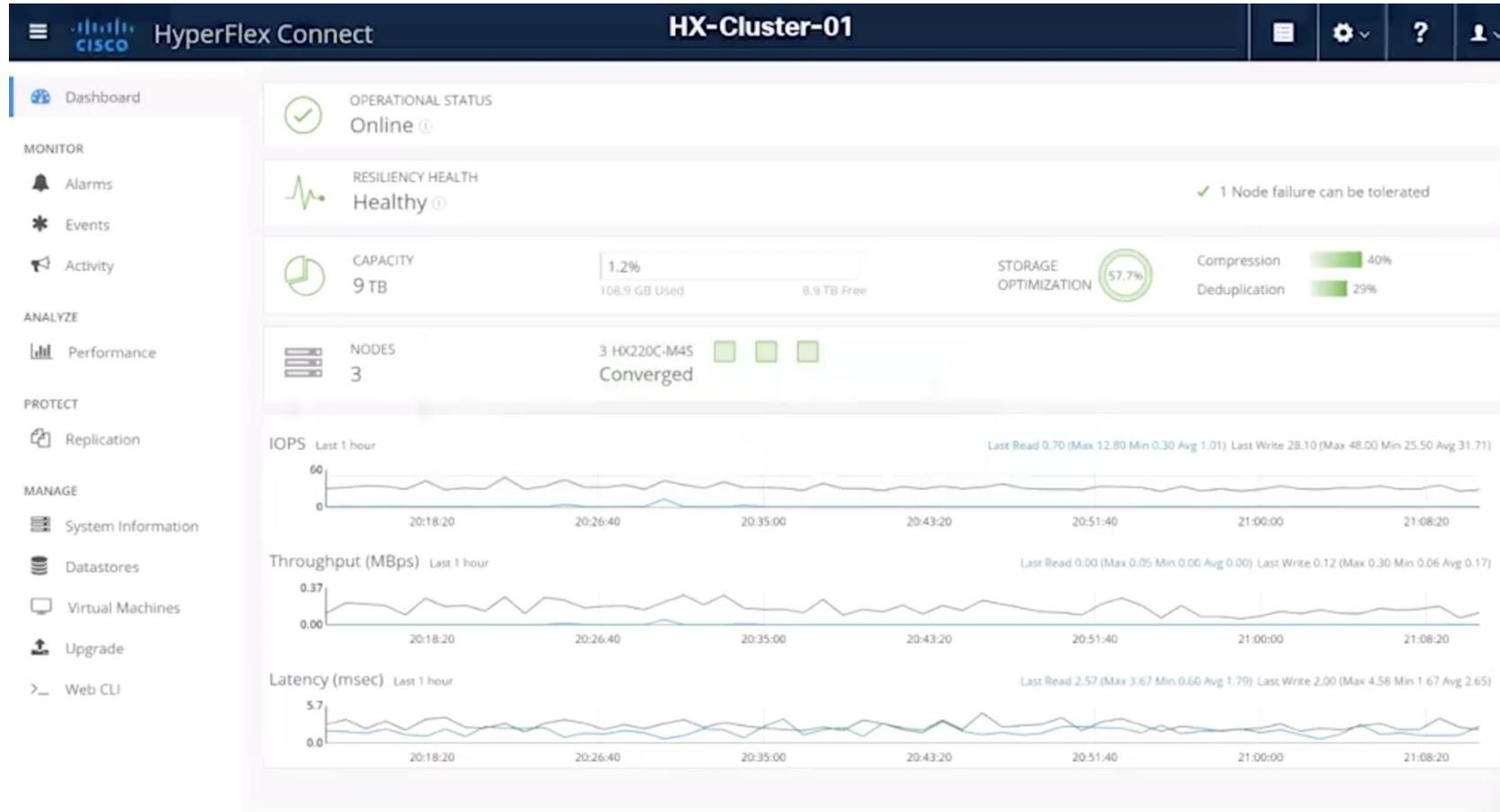
Server 7 FCH1951V06j / HX220C-M4S

Server 4 FCH1951V06A / HX220C-M4S

[Edit Configuration](#)

# HyperFlex - Hyper converged infrastructure

## Monitor HyperFlex health and Performance with HyperFlex Connect



# Model App

CloudCenter – Model  
Application Profile

# Deploy App

CloudCenter – Deploy  
Multi-Cloud

# Monitor App

AppDynamics - Monitor

# Nasdaq, Williams-Sonoma, Cisco

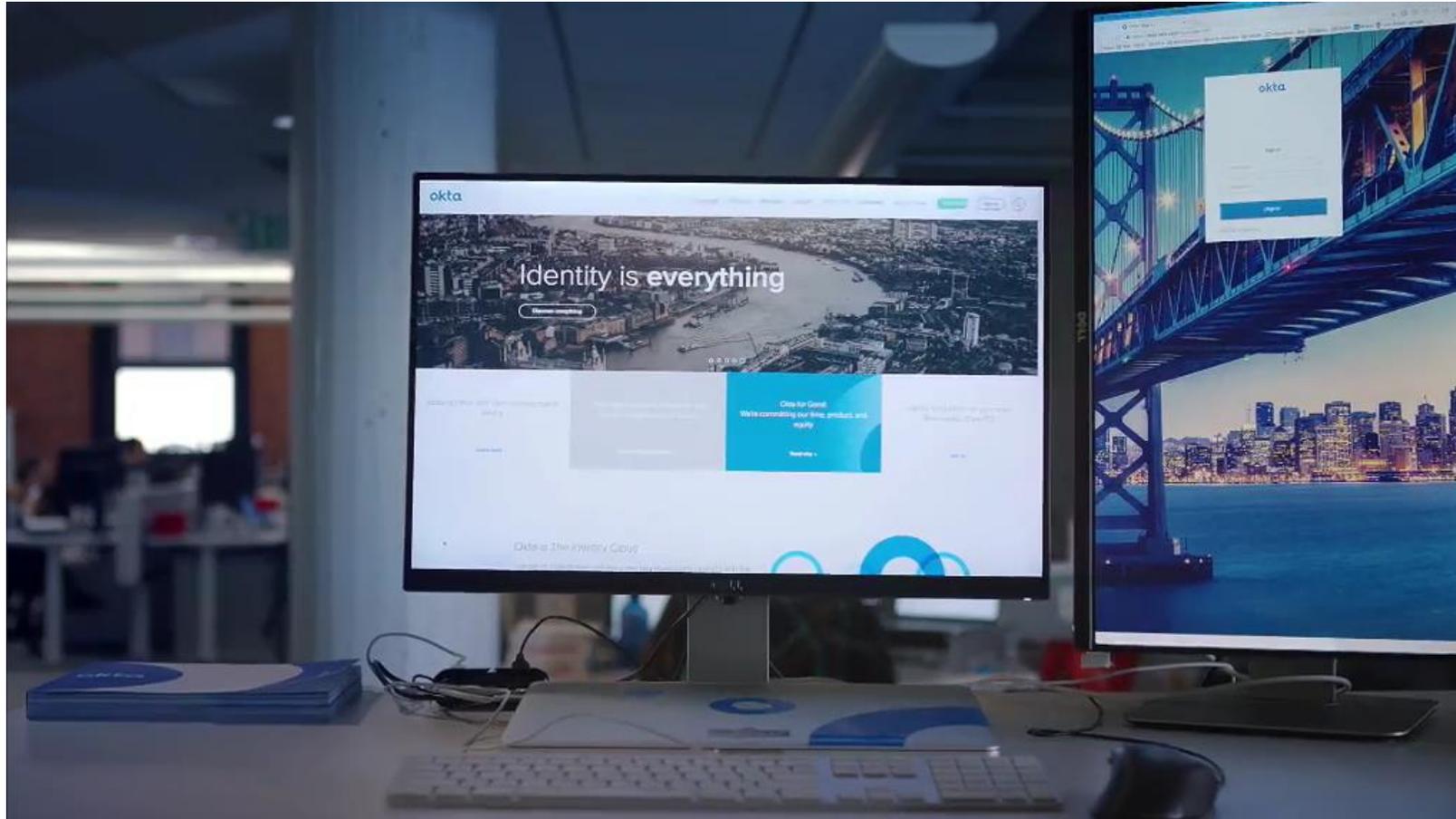




# Lloyds Bank



# Okta



# United Airlines





DevOps Release Analytics v2.0

APPDYNAMICS

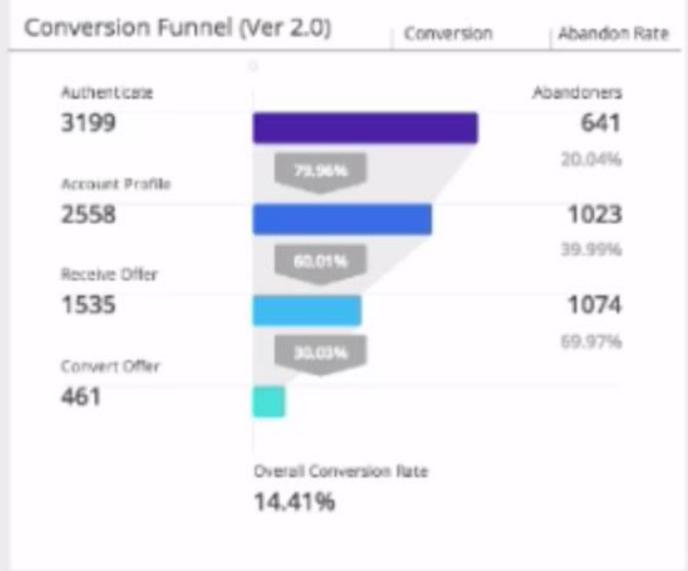
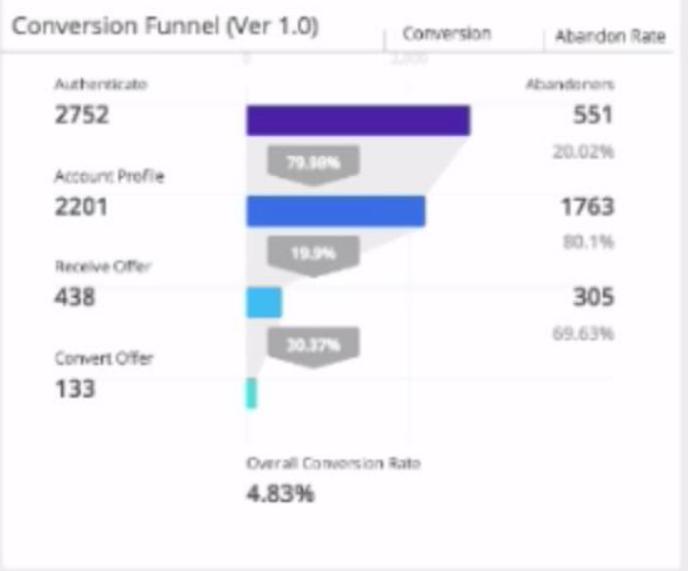
ACTIVE USER SESSIONS      % USER CONVERSION RATE      REVENUE BY VERSION

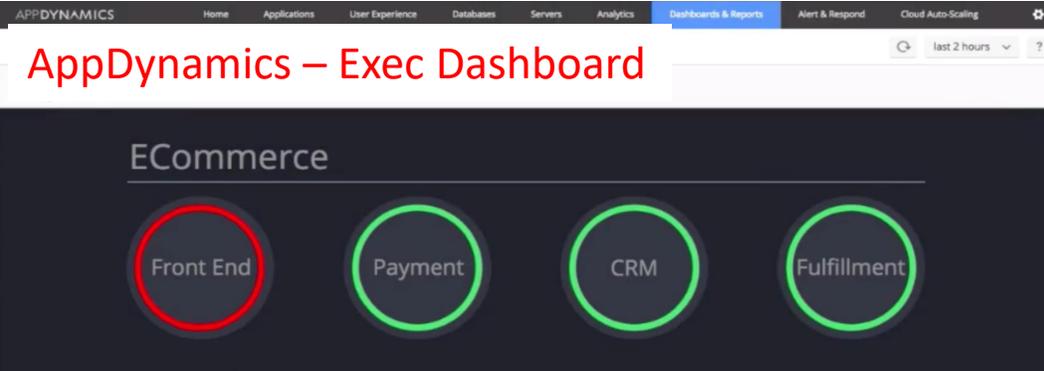
Conversion increased from 4% to 14%)

Version	Active User Sessions	% User Conversion Rate	Revenue
Version 1.0	3,625	4.83	14,148
Version 2.0	3,206	14.41	61,508.5

Fewer users pushed to version 2 (3206)

Revenue increased 4X





AppDynamics - Server Time and Metrics

Server Time: 4,716 ms

HTML Download Time: 16 ms

DOM Building Time: 158 ms

Click on Checkout

The screenshot shows various performance metrics and a sidebar menu. The 'Checkout' item in the sidebar is circled in red with the text 'Click on Checkout' next to it.

AppDynamics - Transaction Flow Map

Transaction Flow Map

Click on Slow

The screenshot displays a detailed transaction flow map for a 'Checkout' transaction. A node labeled 'Inventory Services' is circled in red with the text 'Click on Slow' next to it.

AppDynamics - Call Graph and Slow Call

Transaction: aeed2a0a-c288-4d3d-b937-ce421afda553

8,106 ms - Inventory Serv...

Overview, Call Graph, Slow Call

Execution Time: 8.1 s

Spring Bean - queryExecutor.executeOracleQuery

Self Time: 8103ms (100%)

Total Time: 8103ms (100%)

JDBC Exit Call

Time: 8091ms

From: Inventory-Services

To: XE-Oracle DB-ORACLE-DB

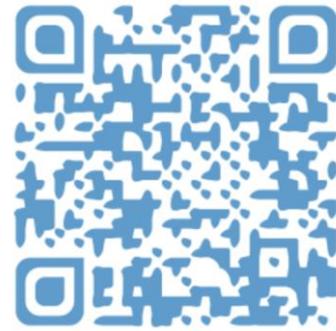
Details

SELECT COMPANY, CITY, COUNTRY, CONTACT, PHONE, TO\_CHAR(SUM(ORDERTOTAL) / 19909999990999M1) AS VALUE\_S  
SHIPPED FROM ( SELECT C.COMPANYNAME AS COMPANY, C.CITY AS CITY, C.COUNTRY AS COUNTRY, C.CONTACTNAME AS  
CONTACT, C.PHONE AS PHONE, O.ORDERID AS DID1 FROM WORLD.ORDERS O JOIN WORLD.CUSTOMERS C ON O.CUSTO  
MERID = C.CUSTOMERID WHERE C.CUSTOMERID = 1 ) JOIN ( SELECT O.ORDERID AS DID2, SUM(OD.QUANTITY \* OD.LINITE  
PRICE \* OD.DISCOUNT) AS ORDERTOTAL FROM WORLD.ORDERS O JOIN WORLD.ORDERS O ON OD.ORDERID = O.  
ORDERID WHERE O.CUSTOMERID = (SELECT \* FROM TABLE(WORLD.GET\_NUM\_T(?, ?)) GROUP BY O.ORDERID ) ON DID1  
= DID2 GROUP BY COMPANY, CITY, COUNTRY, CONTACT, PHONE

The screenshot shows a call graph and a detailed view of a slow call. The 'Slow Call' tab is active, showing a 'Spring Bean - queryExecutor.executeOracleQuery' call with a self time of 8103ms. Below this, a 'JDBC Exit Call' is shown with a time of 8091ms, originating from 'Inventory-Services' and going to 'XE-Oracle DB-ORACLE-DB'. The details section contains a complex SQL query.

.. Reveals the code level details of database and location causing the problem.

# https://learninglabs.cisco.com/labs/tags/AppDynamics/page/1



Account

- My Profile
- Subscriptions
- Sign Out

## Subscriptions

Select a subscription (https://earhart2018031902395049.saas.appdynamics.com)

SAAS - earhart20180...  
EXP 04/03/2018    Start On-Premises Trial >

Subscription Details

Expires in **14 Days**  
on Apr. 3, 2018 12:00 am PST (Apr. 3, 2018 7:00 am GMT)

Show Controller Login In    **Launch AppDynamics**    Contact Sales >

License Usage Report

Onboarding - Application Performance Monitoring    15 Day(s) left in Pro Trial - Upgrade Now

APPDYNAMICS    Home    Applications    User Experience    Databases    Servers    Dashboards & Reports    Alert & Respond

## Getting Started

AppDynamics Unified Monitoring platform gives complete visibility into your entire IT Infrastructure and user experience, applications, databases and servers.

**Getting Started Wizard**    2

What do you want to monitor?

Applications    User Experience    Databases    Analytics

- Java
- .NET
- PHP
- Node.js**    3
- Python

Browser

- Real User
- Synthetic

Mobile

- iOS
- Android

Connected Devices

- IoT SDK

```
JZIMET-M-C099:coding jzimet$ cd appd-101-sample-code/    6
JZIMET-M-C099:appd-101-sample-code jzimet$ ls
README.md    app.js    client.js    node_modules    package.json
JZIMET-M-C099:appd-101-sample-code jzimet$ nano app.js
```

Connect the Agent to the AppDynamics Controller  
Generate load on your instrumented application.

Agent Connected

Waiting for Load (may take a few min)...    10

Your Application    AppDynamics

Continue

It can take a few minutes for the agent to establish connection with the controller and send data related to your application.

Check your Node.js Prerequisites

Verify the Controller Host and Port

Set Application and Tier

Create a new Application:    4

My Sample App

Create a new Tier:    5

My Sample App

```
require("appdynamics").profile({
  controllerHostName: 'earhart2018031902395049',
  controllerPort: 443,

  // If SSL, be sure to enable the next line
  controllerSslEnabled: true,
  accountName: 'earhart2018031902395049',
  accountAccessKey: '5g3lwmeylb3',
  applicationName: 'My Sample App',
  tierName: 'My Sample App',
  nodeName: 'process' // The controller will automatically append the node name with a unique number
});

JZIMET-M-C099:appd-101-sample-code jzimet$ npm start

> app-dynamics-sample-app@1.0.0 start /Users/jzimet/coding/appd-101-sample-code
> node app.js

Example app listening on port 3000!    7    8
```

```
JZIMET-M-C099:~ jzimet$ curl -v http://localhost:3000/500    9
* Trying ::1...
* Connected to localhost (::1) port 3000 (#0)
> GET /500 HTTP/1.1
> Host: localhost:3000
> User-Agent: curl/7.43.0
> Accept: */*
>
< HTTP/1.1 500 Internal Server Error
< X-Powered-By: Express
< Content-Type: text/html; charset=utf-8
< Content-Length: 25
< ETag: W/"19-+xaJTnb0d6oL12z3HaWvea4U500"
< Date: Mon, 19 Mar 2018 10:40:55 GMT
< Connection: keep-alive
```

Onboarding - Application Performance Monitoring    15 Day(s) left in Pro Trial - Upgrade Now

APPDYNAMICS    Home    **Applications**    User Experience    Databases    Servers    Dashboards & Reports    Alert & Respond

## Applications

last 1 hour    ?

Details	Create Application	Actions	View	Calls / min	Sort
My Sample App >					
32 ms Response Time (ms)	1 Calls	1 Calls / min			
100.0 % Error %	1 Errors	1 Errors / min			
Business Transaction Health					
Node Health					

Showing 1-1 of 1

11

Onboarding - Browser Apps 15 Day(s) left in Pro Trial - Upgrade Now

APPDYNAMICS Home Applications User Experience Databases Servers Dashboards & Reports Alert & Respond

Home last 1 hour

Overview [Unified Monitoring](#) Getting Started

User Experience

Browser Apps >



No Browser Apps

[Get Started](#)

Mobile Apps >



No Mobile Apps

[Get Started](#)

Applications >



1 Application

Databases >



No Databases

[Get Started](#)

Servers >



No Servers

[Get Started](#)

Onboarding - Application Performance Monitoring 15 Day(s) left in Pro Trial - Upgrade Now

APPDYNAMICS Home Applications User Experience Databases Servers Dashboards & Reports Alert & Respond

My Sample App last 1 hour

Application Dashboard **Dashboard** Events Top Business Transactions Transaction Snapshots Transaction Score

Application Flow Map



1 Node  
1 call / min  
3 ms  
1 error / min

My Sample App

Events

Application Changes 1

Business Transaction Health

0 critical, 0 warning, 1 normal

Node Health

0 critical, 0 warning, 1 normal

Server Health

---

Transaction Scorecard

Normal 0.0% 0

Slow 0.0% 1

Very Slow 0.0% 1

Stall 0.0% 1

Errors 100.0 15 %

---

Exceptions

Exceptions 23 total 2 / min

HTTP Error Codes - total / min

Error Page Redirects - total / min

Legend

Load 15 calls 1 calls / min

Response Time (ms) 3 ms average

Errors 100% 15 errors 1 errors / min





Onboarding - Application Performance Monitoring 15 Day(s) left in Pro Trial - Upgrade Now

APPDYNAMICS Home Applications User Experience Databases Servers Dashboards & Reports Alert & Respond

My Sample App last 1 hour

Application Dashboard **Events** Top Business Transactions Transaction Snapshots Transaction Score

Showing 13 of 13

Type	Summary	Time	Act...	Business Transaction	Tier	Node
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
Error	/500	03/19/18 12:...	-	/500	My Sample ...	process-0
New Business Transaction Di...	New Business Transaction Di...	03/19/18 12:...	-	/500	My Sample ...	-
App Server Restart	Proxy was re-started Node: p...	03/19/18 12:...	-	-	My Sample ...	process-0
New Node Discovered	New Node Discovered proce...	03/19/18 12:...	-	-	My Sample ...	process-0
New Tier Discovered	New Tier Discovered My Sam...	03/19/18 12:...	-	-	My Sample ...	-
New Application Discovered	New Application Discovered ...	03/19/18 12:...	-	-	-	-

Onboarding - Application Performance Monitoring 15 Day(s) left in Pro Trial - Upgrade Now

APPDYNAMICS Home Applications User Experience Databases Servers Dashboards & Reports Alert & Respond

My Sample App last 1 hour

Application Dashboard **Top Business Transactions** Transaction Snapshots Transaction Score

By Load >

Name	Calls	Calls / min	H...
/500	15	1	

By Response Time >

Name	Time (ms)	H...
/500	3	

By Errors >

Name	Erro...	H...
/500	15	

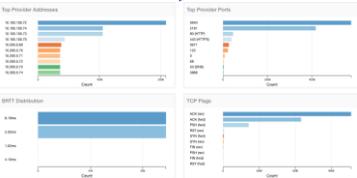
By Slow Transactions >

No Business Transactions

Cisco  
Connect



2 dashboard



7 Search hosts

Search Flows  
Explore Flows

Total Number of Hosts: 164

Filters Enter attributes... Search Hosts

1 HybridCloud multitenancy

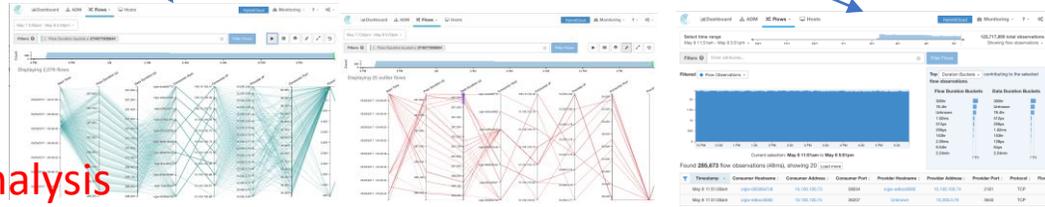
VRF HybridCloud

Tenant HybridCloud (10000)

All  
Default (0)  
Tetration (676767)  
Melab (20000)  
HybridCloud (10000)

6 Player

Outliers



5 Policy Analysis

Permitted Misdropped Escaped Rejected

Policy Analysis

Analyze Current Policies

Run Experiment

Range: 1 hr 6 hr 1 day 1 wk 1 mo 6 mo 1 yr Custom

Granularity: 1 day

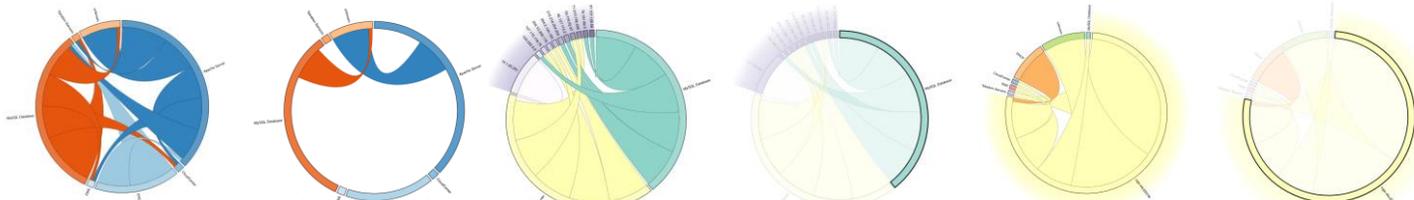
4 Export & search process

Export...

Export Clusters Clusters and Policies JSON XML YAML

Download Cancel

3



Application dependency map

8 Sensor status

Sensors

73 Total Sensors

CPU Overhead

Bandwidth Overhead

Sensor Health

SW Updates to the Latest Revision

Sensor Packet Misses

Sensor Software Version Distribution

Sensor OS Distribution

Displaying 73 of 73 sensors

Hostname	Interfaces	SW Ver (Config Ver)	SW Deployed	OS Platform	Last Check in
W802	<ul style="list-style-type: none"> <li>VRF: Melab IP: 10.107.0.02 Name: Ethernet 2</li> <li>VRF: Melab IP: 10.107.0.00 Name: Ethernet 2</li> <li>VRF: Melab IP: 10.107.0.42 Name: SvcS (87F02202-8411-4E3A-AF5A-C8CA3DF4888)</li> <li>VRF: Melab IP: 10.107.0.00 Name: SvcS (87F02202-8411-4E3A-AF5A-C8CA3DF4888)</li> <li>VRF: Melab IP: 10.107.0.00 Name: SvcS (87F02202-8411-4E3A-AF5A-C8CA3DF4888)</li> <li>VRF: Melab IP: 10.107.0.00 Name: SvcS (87F02202-8411-4E3A-AF5A-C8CA3DF4888)</li> </ul>	1.103.1.10-4064	Jan 12, 5:03 PM	MS-Server2012R2Datacenter	8:24 PM
W816	<ul style="list-style-type: none"> <li>VRF: Tetration IP: 10.200.001-001-001 Name: eth0</li> <li>VRF: Tetration IP: 2.2.1.1 Name: eth1</li> <li>VRF: Default IP: 10.200.0.85 Name: eth0</li> <li>VRF: Default IP: 10.200.0.85 Name: eth0</li> <li>VRF: Default IP: 10.200.0.85 Name: eth0</li> <li>VRF: Tetration IP: 10.200.001-001-001 Name: eth0</li> <li>VRF: Tetration IP: 10.200.001-001-001 Name: eth0</li> </ul>	1.103.1.10-1	Dec 19, 11:16 AM	CentOS-6.5	8:24 PM

Provides (11)

TCP: 22 /usr/sbin/sshd

UDP: 68 /sbin/dhclient

User root

Command /sbin/dhclient -1 -q -lf /var/lib/dhclient/dhclient-eth0-lesser -pf /var/run/dhclient-eth0.pid etc

TCP: 80 /usr/sbin/httpd

TCP: 111 rpcbind

UDP: 111 rpcbind

UDP: 123 ntpd

UDP: 621 /sbin/portreserve

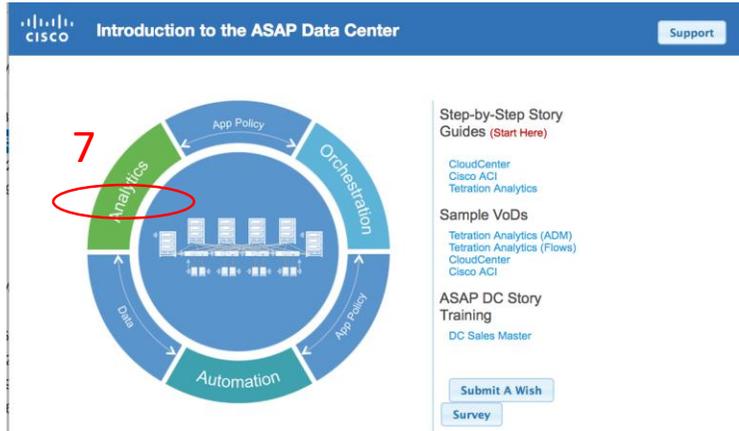
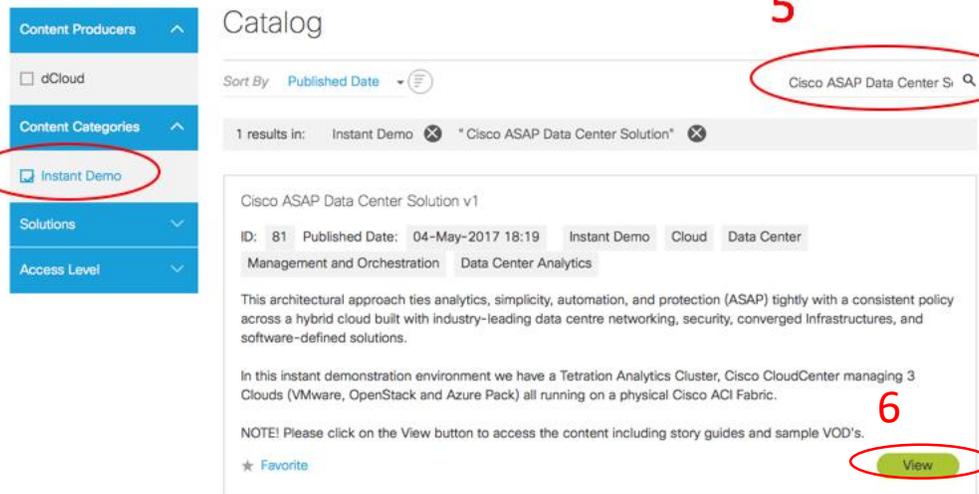
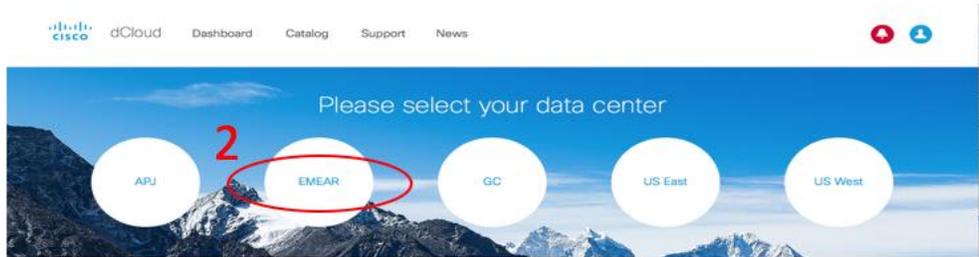
UDP: 763 rpcbind

TCP: 8810 /usr/lib/jvm/java-8-sun/bin/java

UDP: 39237 rpc.statd

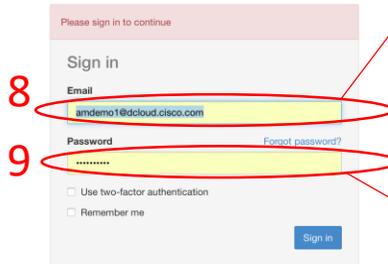
TCP: 88412 rpc.statd

<https://dcloud.cisco.com>

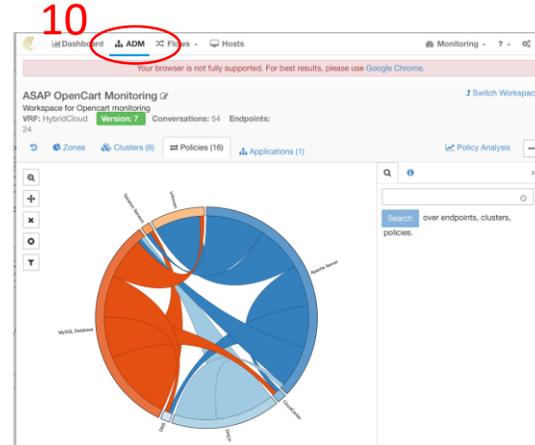


Cisco Tetration Analytics™

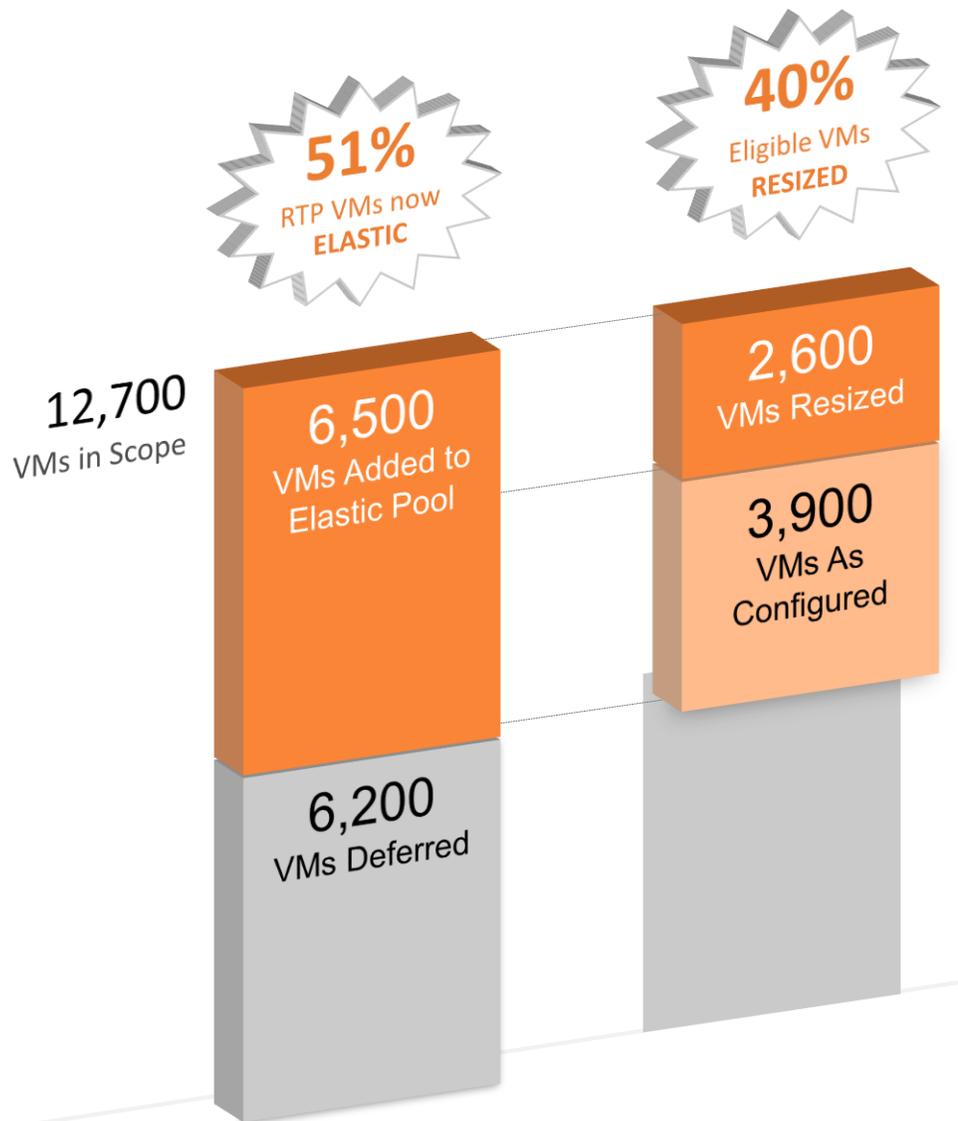
Username :  
[amdemo1@dcloud.cisco.com](mailto:amdemo1@dcloud.cisco.com)



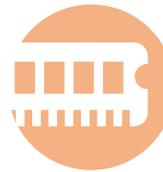
**Password :**  
**C1sco12345**



# Cisco Story

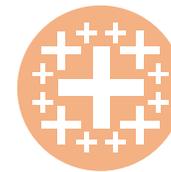


## Outcomes



**20GB+ per VM**

Average resize  
memory recovered



**2,200+ new VMs**

Equivalent capacity recovered



**50% Reduction**

Of over-provisioned  
memory



**80% less contention**

Among Virtual Clusters



**\$1 Million+**

Cost avoidance



**600+ VMs upsized**

Automatically since Feb 10,  
with *no downtime*



**4,200 vCPUs**

Recovered



Q3 RCDN &  
ALLN

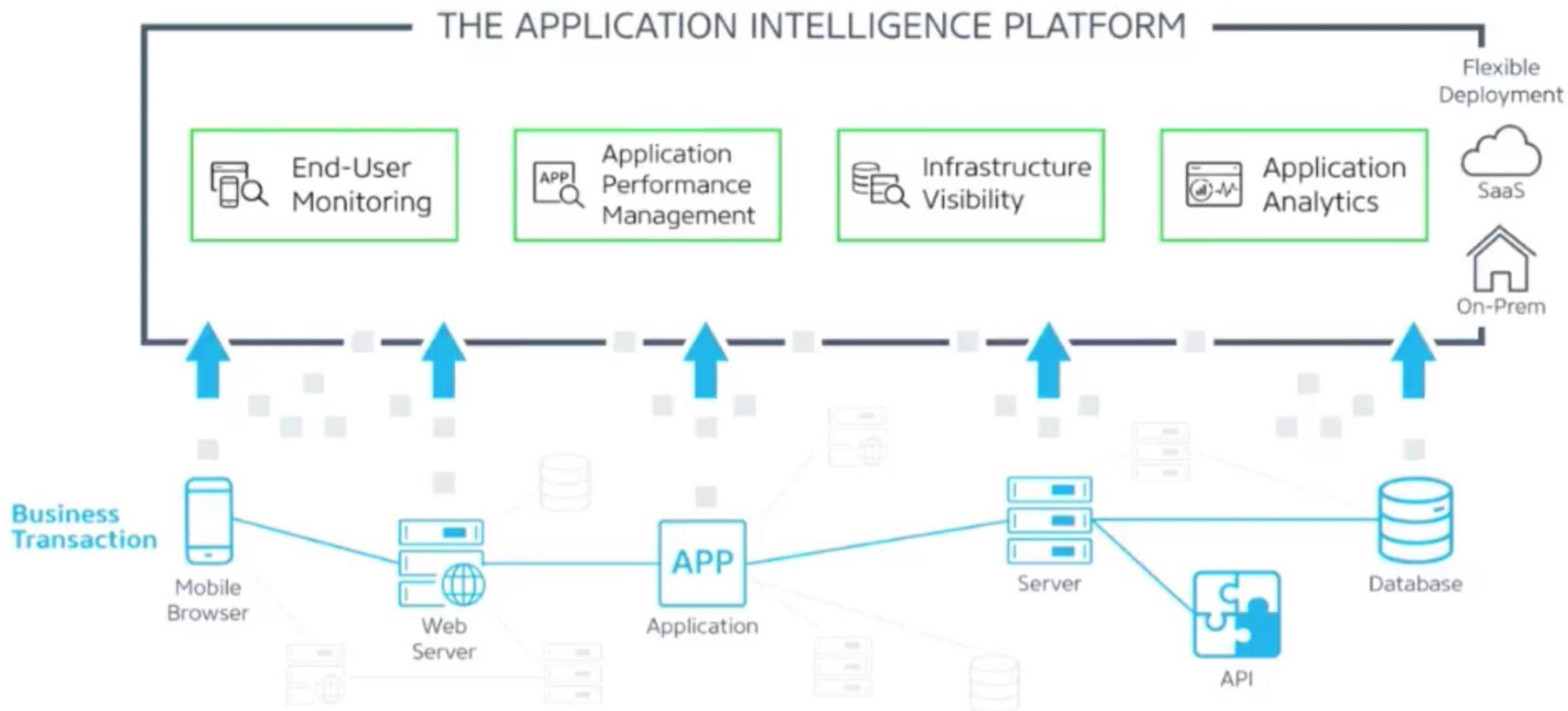


eStore  
option

N E X T S T E P S

# Example 1

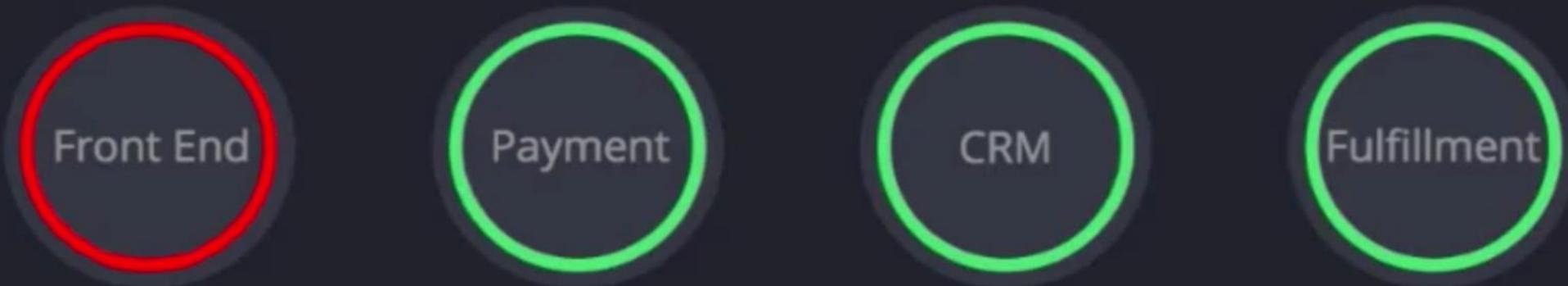
# Application Intelligence Platform Architecture



Executive summary of key components of two enterprise applications. Drilling Down into Front End ...

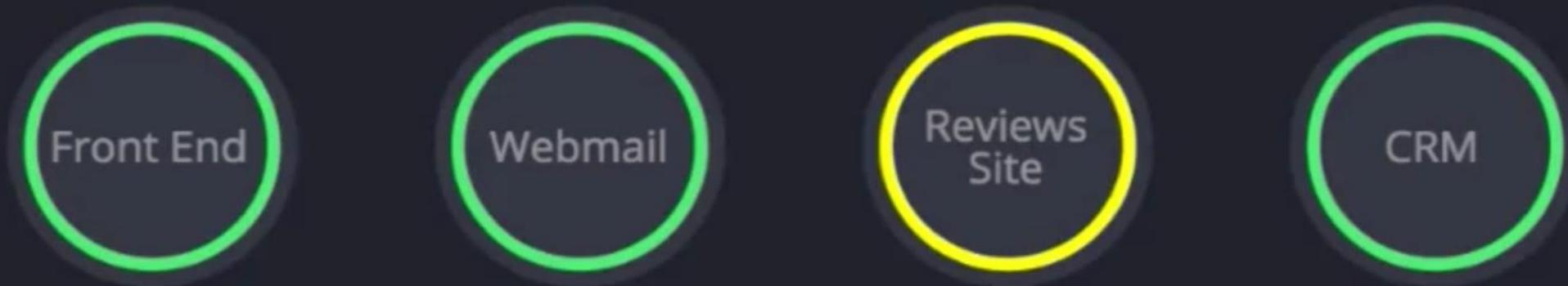
## ECommerce

---



## Movie Tickets

---



# AppDynamics – User Journey Dashboard

... reveals user Journey. Checkout is critical to driving conversions. Drilling down into Checkout ...



ECommerce

www.ecommerce.com/store/viewcart/senditems.action

last 2 hours

Browser App Dashboard

Dashboard Browser Snapshots

Analyze

REAL USERS

Sessions

**Pages & AJAX Requests**

Analyze

SYNTHETIC

Jobs

Sessions

Pages

On-Demand

Health Rule Violations

Events

Alert & Respond

Metric Browser

Configuration

Summary

Time to Service End User Request = 5142 ms

END USER RESPONSE TIME

5,142 milliseconds

LOAD

865 Reqs 7 Req / min

CACHE HIT

0 % Cache Hits

PAGE VIEWS WITH JS ERRORS

Errors Errors / min

... reveals detailed timing and navigation data for checkout. Scroll down to Checkout Details.

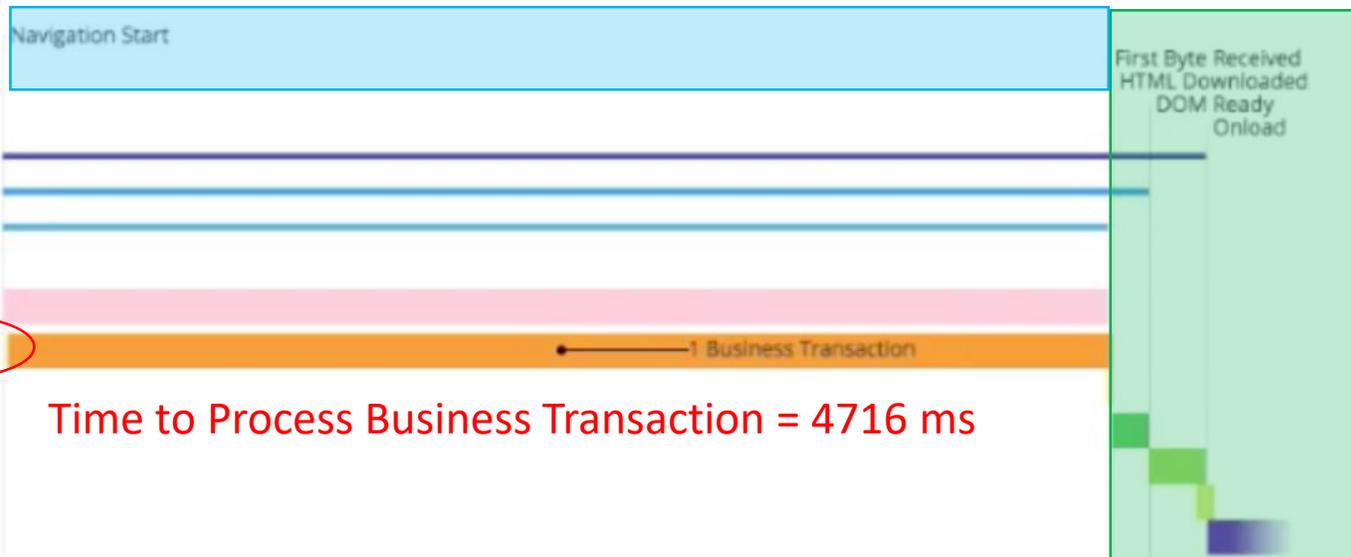
Timing Breakdown

Server-Side Events 4716 ms

Client-Side Events 426 ms

Trends/Details

<input checked="" type="checkbox"/>	End User Response Time	5142 ms
<input checked="" type="checkbox"/>	DOM Ready Time	4897 ms
<input checked="" type="checkbox"/>	First Byte Time	4723 ms
<input checked="" type="checkbox"/>	Server Connection Time	0 ms
<input checked="" type="checkbox"/>	Response Available Time	4723 ms
<input checked="" type="checkbox"/>	Server Time	4716 ms
<input checked="" type="checkbox"/>	HTML Download Time	16 ms
<input checked="" type="checkbox"/>	DOM Building Time	158 ms
<input checked="" type="checkbox"/>	Resource Fetch Time	244 ms
<input checked="" type="checkbox"/>	Onload	
<input checked="" type="checkbox"/>	Post Page Load	



Time to Process Business Transaction = 4716 ms

ECommerce

www.ecommerce.com/store/viewcart/senditems.action

last 2 hours

Browser App Dashboard

Dashboard Browser Snapshots

Analyze

REAL USERS

Sessions

Pages & AJAX Requests

Analyze

SYNTHETIC

Jobs

Sessions

Pages

On-Demand

Health Rule Violations

Events

Alert & Respond

Metric Browser

Configuration

View in Applications

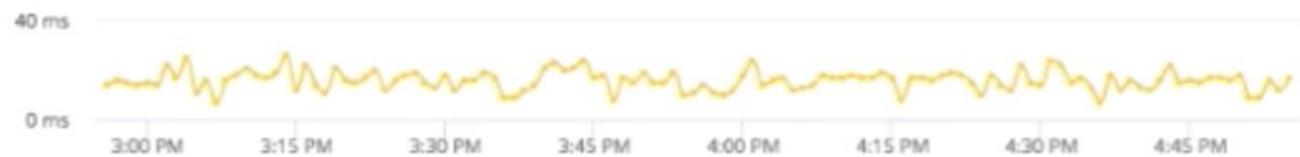
### Server Time

Server Time 4,716 ms



### HTML Download Time

HTML Download Time 16 ms



Checkout details. Drill Down to Checkout ...

### Related Business Transactions

Show Business Transactions Without Load

Checkout

Average Time 4,716 ms

Average Load 7 Req/Min

863 Requests

827 Normal, 10 Slow, 53 Very Slow, 0 Stalls



### DOM Building Time

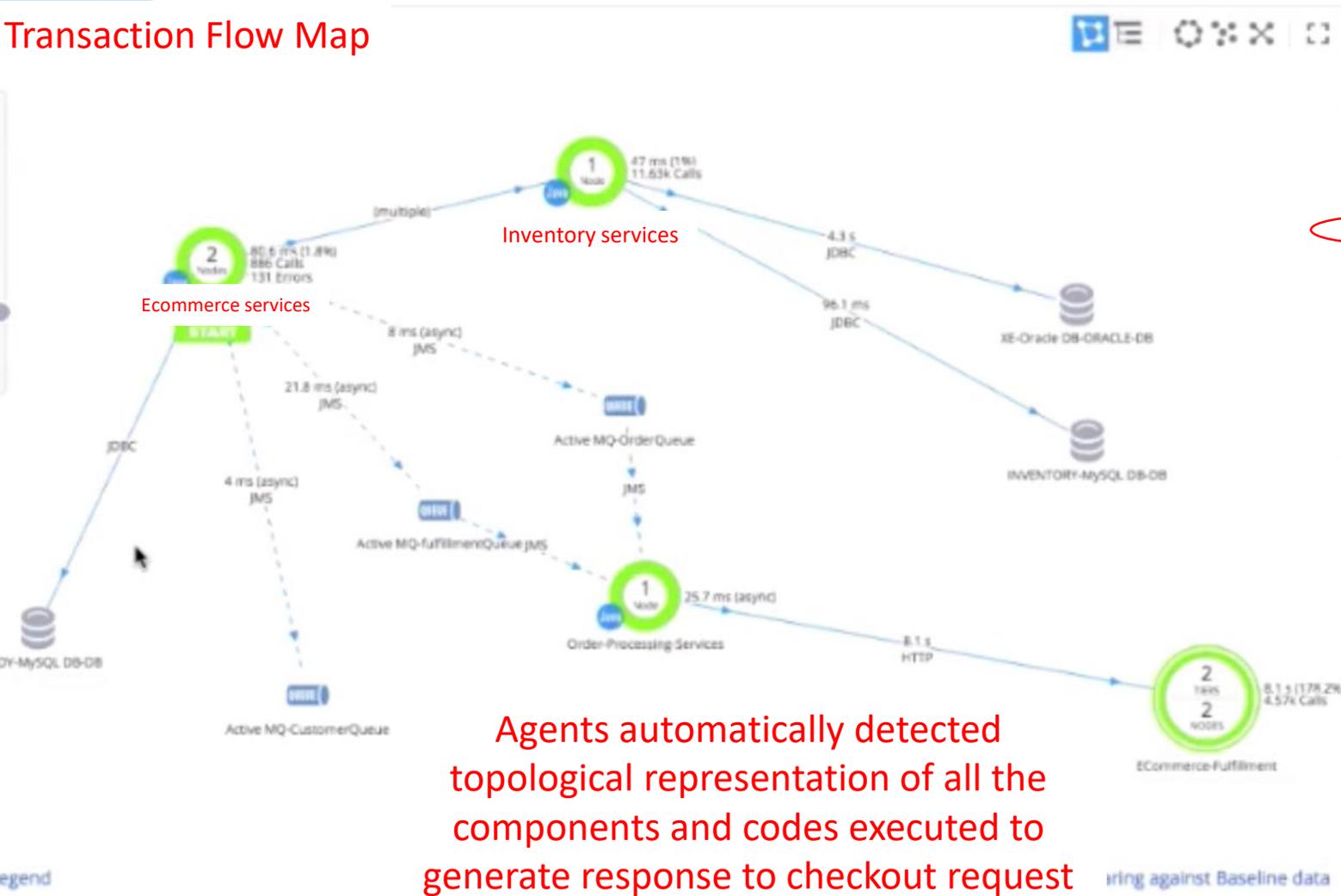
DOM Building Time 158 ms

- ECommerce
- Application Dashboard
- Business Transactions**
- Service Endpoints
- Tiers & Nodes
- Servers
- Database Calls
- Remote Services
- Troubleshoot
- More

Checkout

Weekly Trend - Last 3 months | last 2 hours

Dashboard | Events | Slow Response Times | Errors | Transaction Snapshots | Transaction Score



### Transaction Flow Map

Ecommerce services

Inventory services

Agents automatically detected topological representation of all the components and codes executed to generate response to checkout request

Events		
Health Rule Violations Started	3	⚠️
Business Transaction Health	3	⚠️
Transaction Scorecard		
Normal	78.1 %	692
Slow	1.1 %	10
Very Slow	6.0 %	53
Stall	0.0 %	-
Errors	14.8 %	131

Based on dynamic baseline, flow map shows categorization of transaction in the respective performance ranges by percentile. We can drill down in particular range to quickly diagnose what is affecting those transactions.

Legend

Load: 886 calls, 7 calls / min

Response Time: 4,552 ms average

Errors: 14.8% 131 errors, 1 errors / min

ECommerce



Checkout



last 2 hours



Application Dashboard

Dashboard

Events

Slow Response Times

Errors

Transaction Snapshots

Transaction Score

Business Transactions

All Snapshots

Slow and Error Transactions

Diagnostic Sessions

Periodic Collection



Details



Filters



Analyze



Actions



Configure



Showing 53 of 53

+ Add Criteria

User Experience: (1) Very Slow



Business Transaction: (1)



	Time ↓	Exe Time (ms)	URL	Business Transaction	Tier	Node
🚨	16/06/16 4:17:58 PM	32,581	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB2_...
🚨	16/06/16 4:17:48 PM	57,364	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB1_...
🚨	16/06/16 4:16:29 PM				ervices	ECommerce_WEB1_...
🚨	16/06/16 4:15:31 PM				ervices	ECommerce_WEB1_...
🚨	16/06/16 4:13:09 PM				ervices	ECommerce_WEB2_...
🚨	16/06/16 4:12:55 PM	24,820	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB2_...
🚨	16/06/16 4:12:50 PM	41,381	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB2_...
🚨	16/06/16 4:11:50 PM	57,532	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB2_...
🚨	16/06/16 4:09:45 PM	40,963	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB1_...
🚨	16/06/16 4:08:47 PM	41,111	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB1_...
🚨	16/06/16 4:05:52 PM	41,026	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB2_...
🚨	16/06/16 4:05:40 PM	24,704	/appdynamicspilot/ViewCartItem...	Checkout	ECommerce-Services	ECommerce_WEB2_...

Drilling down into a transaction with very slow user experience we see individual snapshots of each transactions

Transaction: aeed2a0a-c288-4d3d-b937-ce421a6daf53

Overview Slow Calls and Errors Waterfall View Segment List

Summary

Server Browser

User Experience: Very Slow

Execution Time  
32.6 s

Timestamp  
16/06/16 4:17:58 PM

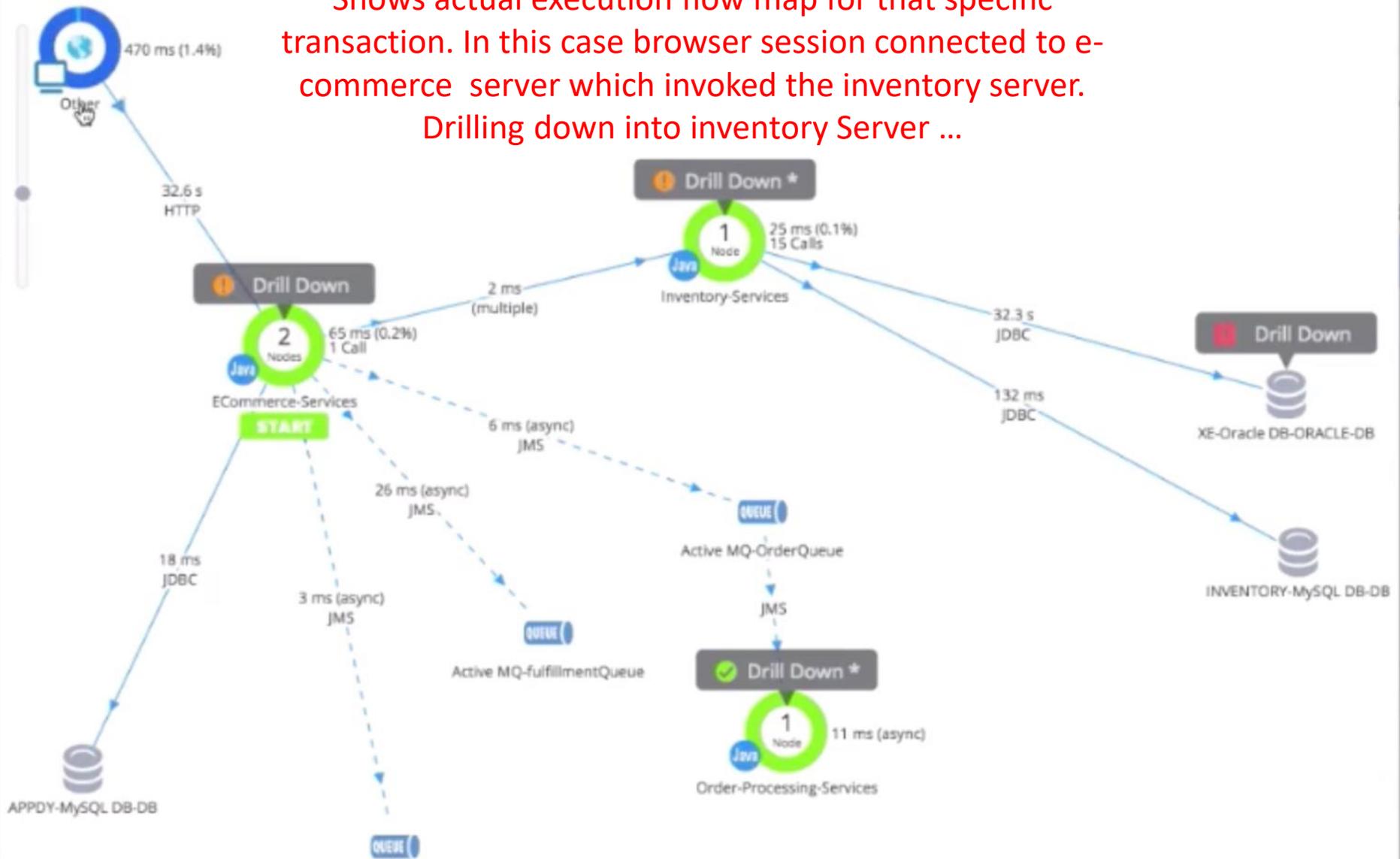
Business Transaction  
Checkout

More Details >

Potential Issues

com.appdynamics.pilot.webservicelient.SoapUtils.raisePO	32.5 s
com.appdynamics.inventory.QueryExecutor.executeOracleQuery	32.3 s
OrderServiceImplService.createOrder	8.1 s
SELECT COMPANY, CITY, COUNTRY, CONTACT, PHONE, TO_CHAR(SUM(ORDERTOTAL), 'L99G999G999D99MI') AS VALUE...	8.1 s
OrderServiceImplService.createOrder	8.1 s
OrderServiceImplService.create	8.1 s

Transaction Flow Map



Shows actual execution flow map for that specific transaction. In this case browser session connected to e-commerce server which invoked the inventory server. Drilling down into inventory Server ...

Transaction: aeed2a0a-c288-4d3d-b937-ce421a6daf53

Overview Slow Calls and Errors Waterfall View Segment List

Summary Transaction Flow Map

Server Browser



User Experience: Very Slow

Execution Time 32.6 s

Timestamp 16/06/16 4:17:58 PM

Business Transaction Checkout

More Details >

Potential Issues

com.appdynamics.pilot.websecclient.SoopUtils.raisePO

com.appdynamics.inventory.CeryExecutor.executeOracleQuery

OrderServiceImplService.createOrder

SELECT COMPANY, CITY, COUNTRY, CONTACT, PHONE, TO\_CHAR(SUM(ORDERTOTAL), 'L99G9L999D99M') AS VALUE...

OrderServiceImplService.createOrder 8.1 s

OrderServiceImplService.createOrder 8.1 s

### Select a Call to Drill Down into

Drill Down Show: All Calls Originating From: Show all

Exe Time (ms)	Called from	Exit Calls	Start Time
0	ECommerce-Services [HT	No Exit Calls Made	16/06/16 4:17:58 PM
19	ECommerce-Services [We	Exit Calls Made: 6 JDBC calls (3.2 ms avg, 12 ms max)	16/06/16 4:17:58 PM
13	ECommerce-Services [We	Exit Calls Made: 6 JDBC calls (2.3 ms avg, 7 ms max)	16/06/16 4:17:58 PM
1	ECommerce-Services [HT	No Exit Calls Made	16/06/16 4:17:58 PM
8106	ECommerce-Services [We	Exit Calls Made: 8 JDBC calls (1013 ms avg, 8091 ms max)	16/06/16 4:17:58 PM
16	ECommerce-Services [We	Exit Calls Made: 6 JDBC calls (2.3 ms avg, 7 ms max)	16/06/16 4:18:06 PM
1	ECommerce-Services [HT	No Exit Calls Made	16/06/16 4:18:06 PM
8084	ECommerce-Services [We	Exit Calls Made: 8 JDBC calls (1009.9 ms avg, 8066 ms max)	16/06/16 4:18:06 PM

.. shows code path of transactions and drill down show ...

Close

Drill Down

XE-Oracle DB-ORACLE-DB

INVENTORY-MySQL DB-DB

APPDY-MySQL DB-DB

Node 11 ms (async)  
Order-Processing-Services

Transaction: aeed2a0a-c288-4d3d-b937-ce421a6daf53

8,106 ms - Inventory-Serv... X



Overview

Call Graph

Slow Calls &amp; Errors

DB &amp; Remote Service Calls

Server

Data Collectors

More

Actions

Execution Time 8.1 s Node ECommerce\_WS\_NODE Timestamp 16/06/16 4:17:58 PM



Details



Filters



Set Root



Reset Root



Export



Showing 5 of 5

Name	Time (ms)	Percent %	Exit Calls / Threads...
Servlet - jaxws-servlet:doPost	3 ms (self)	0%	
Web Service - com.sun.xml.ws.server.InvokerTube\$2:invoke:149	0 ms (self)	0%	
Spring Bean - orderService:createOrder:27	0 ms (self)	0%	
Spring Bean - OrderDao:createOrder:59	0 ms (self)	0%	
Spring Bean - queryExecutor:executeOracleQuery:52	8,103 ms (se...)	100%	JDBC

.. Slow JDBC flow contributed to poor performance. Finally drilling down into the JDBC call ...

Transaction: aeed2a0a-c288-4d3d-b937-ce421a6daf53

8,106 ms - Inventory-Serv...

Overview Call Graph Slow Call

Execution Time 8.1 s Node ECommer

Details Filters Set Root Resets Root

Name

- Servlet - jaxws-servlet:doPost
  - Web Service - com.sun.xml.ws.server
    - Spring Bean - orderService:create
      - Spring Bean - OrderDao:creat
        - Spring

executeOracleQuery:52

Name	Spring Bean - queryExecutor:executeOracleQuery	
Type	Spring Bean	
Class	com.appdynamics.inventory.QueryExecutor	
Method	executeOracleQuery	
Line Number	52	
Self Time	8103ms	100%
Total Time	8103ms	100%

Exit Calls

Type	Details	Count	Time (ms)	% Time	From	To
JDBC	SELECT COMPANY, CITY, COUNTRY, CONTACT,...	1	8091	99.8%	Inventory-S...	XE-Orade D...

JDBC Exit Call

Time 8091ms  
 From Inventory-Services  
 To XE-Oracle DB-ORACLE-DB  
 Details SELECT COMPANY, CITY, COUNTRY, CONTACT, PHONE, TO\_CHAR(SUM(ORDERTOTAL), 'L99G999G999D99MI') AS VALUE\_S  
 HIPPED FROM ( SELECT C.COMPANYNAME AS COMPANY, C.CITY AS CITY, C.COUNTRY AS COUNTRY, C.CONTACTNAME AS  
 CONTACT, C.PHONE AS PHONE, O.ORDERID AS Oid1 FROM WORLD.ORDERS O JOIN WORLD.CUSTOMERS C ON O.CUSTO  
 MERID = C.CUSTOMERID WHERE C.CUSTOMERID = ? ) JOIN ( SELECT O.ORDERID AS Oid2, SUM(OD.QUANTITY \* OD.UNITP  
 RICE \* OD.DISCOUNT) AS ORDERTOTAL FROM WORLD.ORDERDETAILS OD JOIN WORLD.ORDERS O ON OD.ORDERID = O.  
 ORDERID WHERE O.CUSTOMERID = (SELECT \* FROM TABLE(WORLD.GET\_NUM\_T(?, ?)) GROUP BY O.ORDERID ) ON Oid1  
 = Oid2 GROUP BY COMPANY, CITY, COUNTRY, CONTACT, PHONE

Properties

Query Type	Query
Statement Type	String
Session Id	139

.. Reveals the code level details of database and location causing the problem.

## Recently Visited

- 🔍 ECommerce - Dashboard
- 🔍 Checkout - Transaction Snapshots
- 🔍 Checkout - Dashboard
- 🌐 www.ecommerce.com/store/viewcartisendite...
- 📊 AppDynamics - User Journey Dashboard
- 📊 AppDynamics - Exec Dashboard

## Applications 7

2 critical, 1 warning, 4 normal

- 🟢 AppDynamicsPCFOpsMetrics\_AppD
- 🔴 ECommerce
- 🟢 ECommerce-Fulfillment
- 🔴 Movie Tickets Online
- 🟢 MyDistributedApp

## User Experience

## Browser apps 3

- 🌐 ECommerce
- 🌐 Movie Tickets Online
- 🌐 Online\_Retail

## Mobile apps 2

- 📱 EcommAndroid
- 📱 Ecommerce-iOS

## Databases 8

0 critical, 0 warning, 8 normal

- 🟢 E-Commerce MySQL
- 🟢 E-Commerce Oracle
- 🟢 MovieTicketsOnline MSSQL

## Servers 16

1 critical, 0 warning, 13 normal

- 🟢 Commerce
- 🌐 ECommerce-ibr

## Analytics

🔍 10.8k  
Transactions🔍 0.1m  
Log Events

AppDynamics provide comprehensive platform for monitoring performance of complex distributed applications from **native mobile and browser based applications** all the way to **back end server side applications** written in different languages using wide variety of **infrastructure, network and databases** providing advanced application **analytics** capabilities for capturing the business outcomes achieved through the applications

# Example 2

### DevOps Leadership Dashboard

+ Add Widgets   -- Actions   📱 Mobile   🛠️ Mode

#### APPDYNAMICS

<b>UNIQUE SESSIONS</b> <div style="text-align: center; font-size: 24px; font-weight: bold;">5,237</div> Unique Sessions	<b>% OFFERS GIVEN</b> <div style="text-align: center; font-size: 24px; font-weight: bold;">15.95</div> % Users Seeing Offers	<b>% OFFER CONVERSION</b> <div style="text-align: center; font-size: 24px; font-weight: bold;">4.73</div> % Conversion	<b>OFFERS CONVERTED</b> <div style="text-align: center; font-size: 24px; font-weight: bold;">167</div> Offers Converted	<b>OFFERS REVENUE</b> <div style="text-align: center; font-size: 24px; font-weight: bold;">22,967</div> Offers Revenue
--	---	---	--	---

#### User Experience



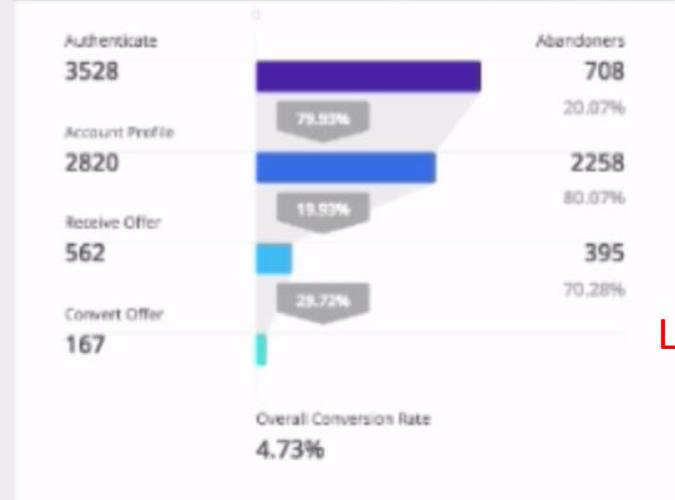
#### User Conversions



#### Transactions

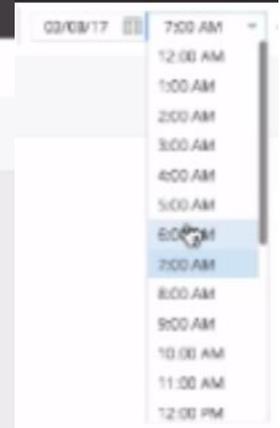
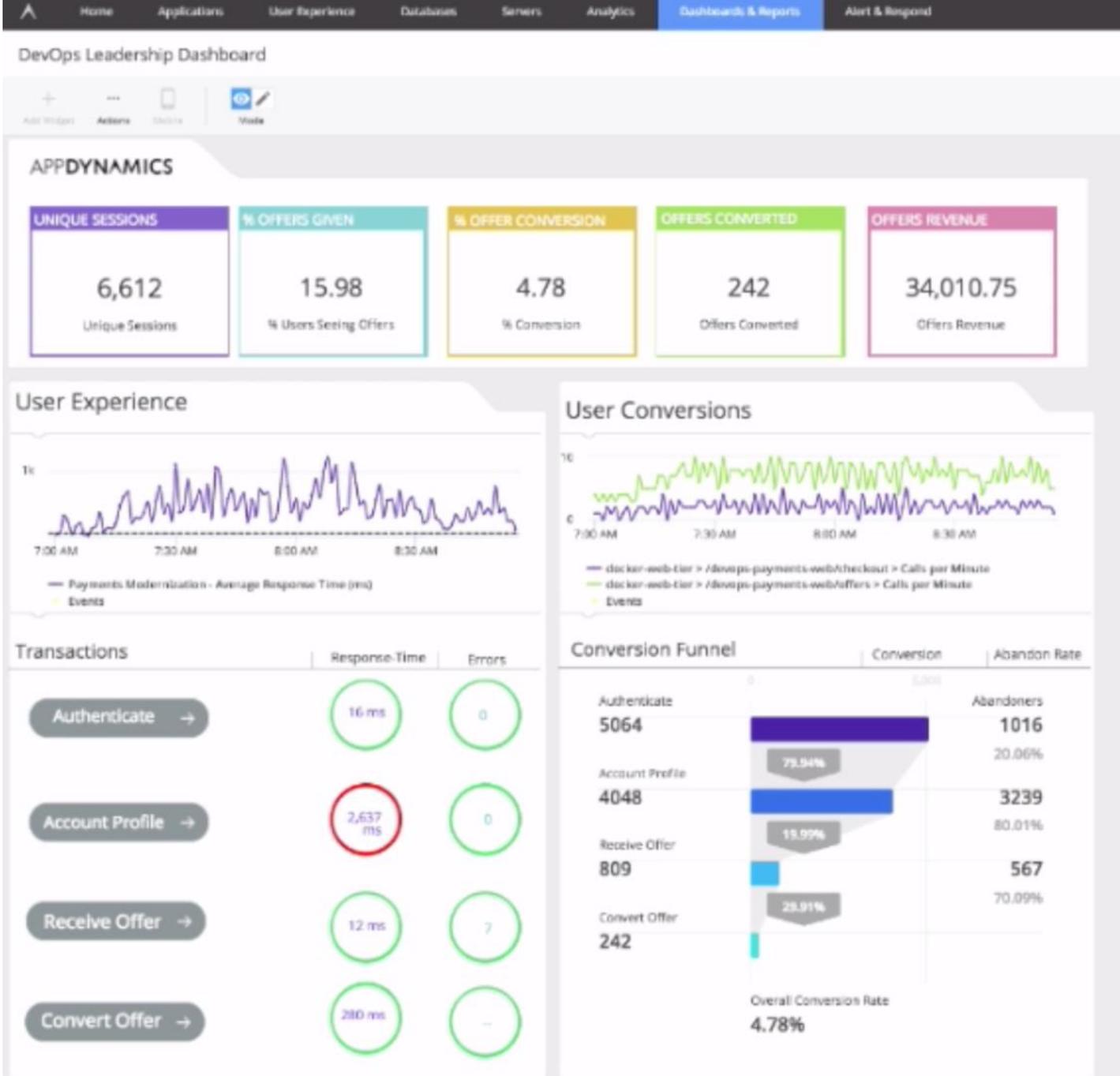
	Response-Time	Errors
Authenticate →	16 ms	0
Account Profile →	1,048 ms	0
Receive Offer →	12 ms	14
Convert Offer →	282 ms	-

#### Conversion Funnel



5 Minutes  
15 Minutes  
30 Minutes  
1 Hour  
**2 Hours**  
3 Hours  
4 Hours  
5 Hours  
12 Hours  
1 Day  
3 Days  
1 Week  
2 Weeks  
1 Month  
3 Month  
6 Month  
1 Year  
Custom  
Saved Time Ranges  
application-deploy-v2.0.537  
application-deploy-v1.0.530  
application-deploy-v2.0.528  
application-deploy-v1.0.531  
application-deploy-v2.0.529  
**application-deploy-v1.0.532**  
Manage Custom Time Ranges

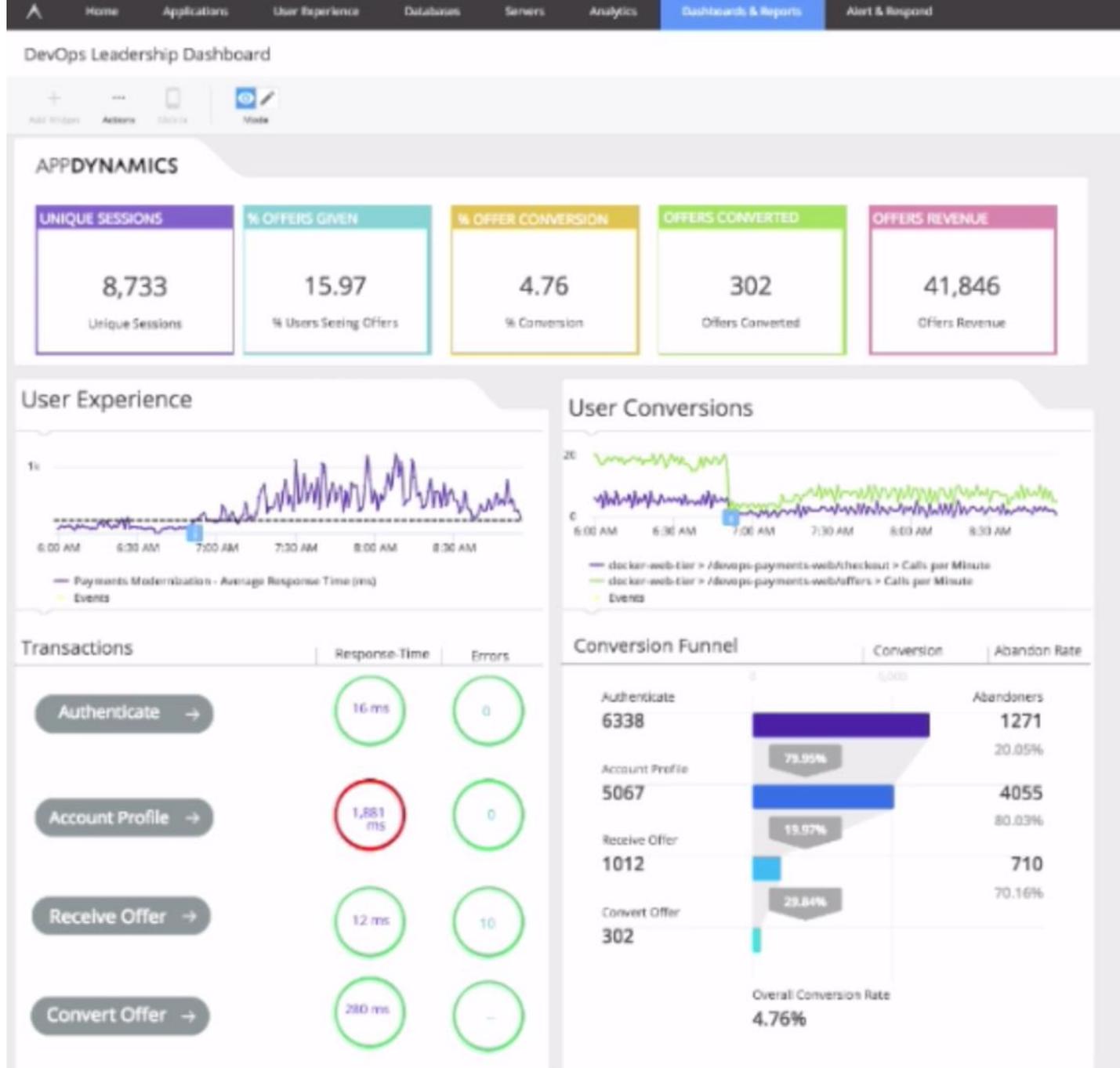
Latest version 1 code release



Tools in place are siloed and unconnected. Marketing teams might have visibility into conversions with google analytics and they can't connect that data with what is happening with IT and understand how new releases or application performance affects business conversion.

Jump back 1 hour  
Start time = 6 AM

Lack of real time visibility in terms of business conversion, user experience and application health

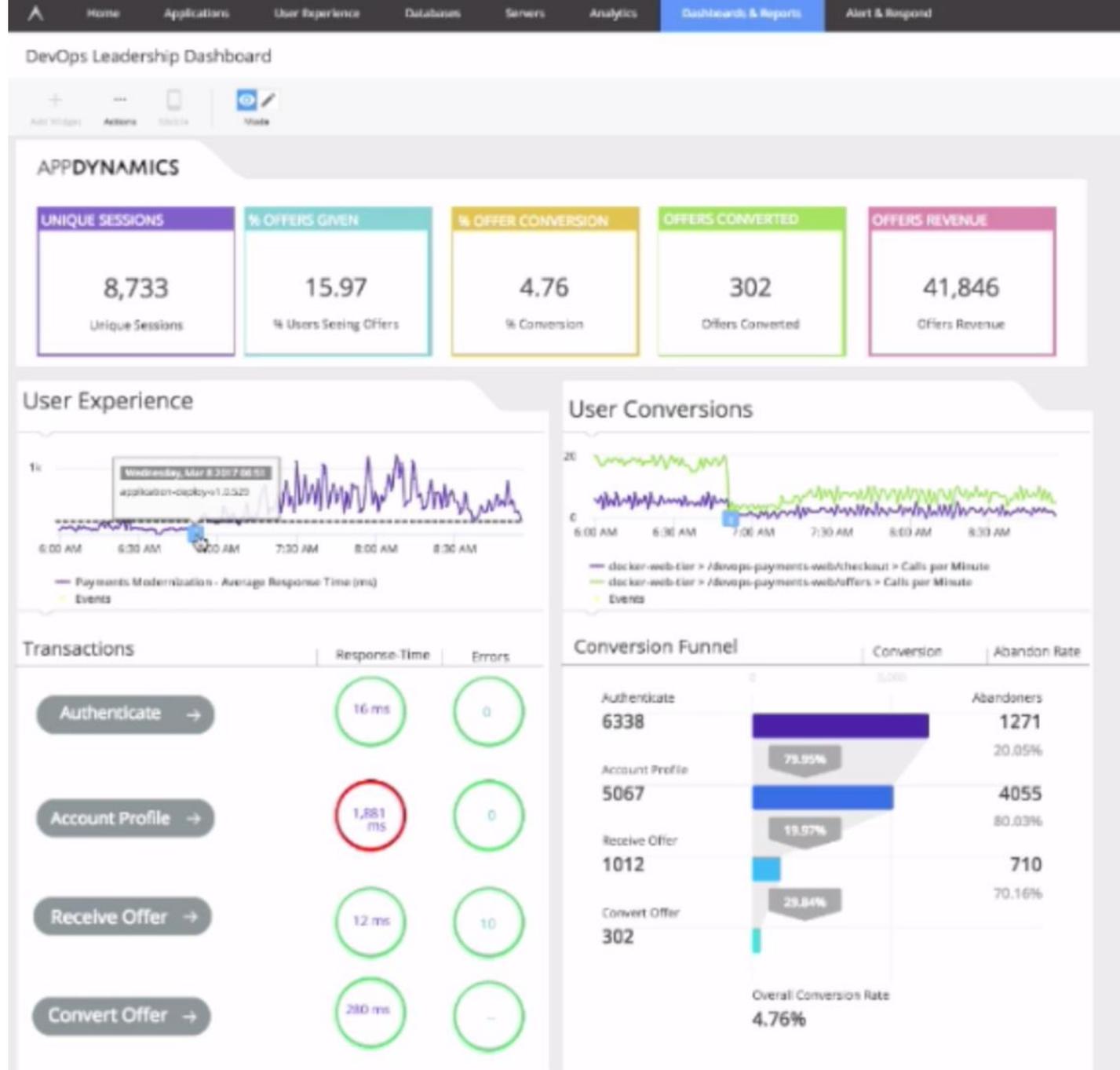


1. two primary functions for generating revenue : online store and providing 3<sup>rd</sup> party marketing offers to it's user.

2. Break down organizational boundaries end enable all teams to understand how they can contribute to business goals.

3. Real time visibility into marketing offers subsystem.

4. Track key business performance indicator : Number of active users, percentage of users that are receiving and converting on marketing offers and overall revenue generated



5. Integration with CI platform (Jenkins). Operations team can see when dev or release team push new version of application

6. New release push has major impact on user experience and user conversion

7. Majority of users authenticate and begin to sign in account profile and there is 80% abandon rate in funnel before user receive marketing offer

8. There is critical issue affecting response time with account profile

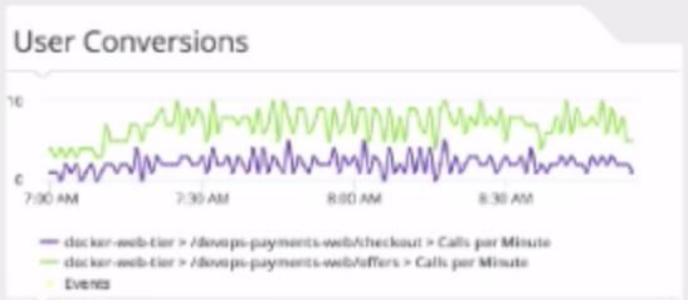
9. IT needs to investigate problem which seems driving down business conversion

### DevOps Leadership Dashboard

+ Add Widgets   - Actions   📱 Mobile   🗨️ Voice

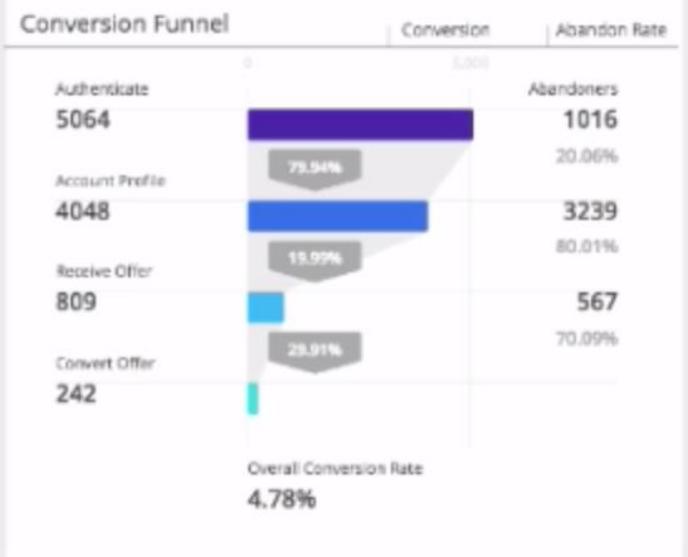
#### APPDYNAMICS

<b>UNIQUE SESSIONS</b> 6,612 Unique Sessions	<b>% OFFERS GIVEN</b> 15.98 % Users Seeing Offers	<b>% OFFER CONVERSION</b> 4.78 % Conversion	<b>OFFERS CONVERTED</b> 242 Offers Converted	<b>OFFERS REVENUE</b> 34,010.75 Offers Revenue
--	---	---	--	--



#### Transactions

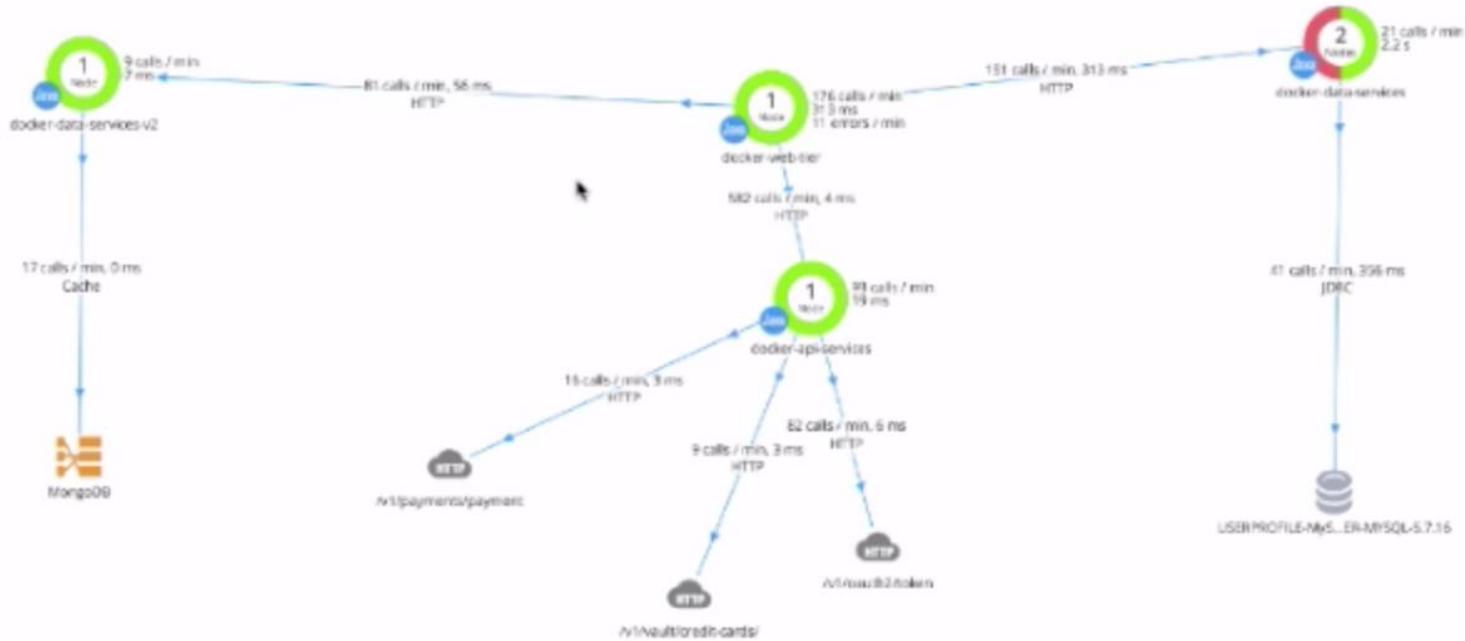
	Response Time	Errors
Authenticate →	16 ms	0
Account Profile →	2,637 ms	0
Receive Offer →	12 ms	7
Convert Offer →	280 ms	-



Double click User Experience

# Application Dashboard

### Application Flow Map



#### Events

Health Rule Violations Started	52
Overall Application Performance	1
Business Transaction Health	20
Node Health	31
Application Changes	2,412

#### Business Transaction Health

1 critical, 0 warning, 7 normal

#### Node Health

1 critical, 0 warning, 9 normal

#### Servers

0 critical, 0 warning, 1 normal

#### Transaction Scorecard

Normal	62.6%	30.1k
Slow	0.4%	171
Very Slow	0.6%	246
Stall	0.0%	<1
Errors	6.4%	2.7k

#### Exceptions

Exceptions	2,741 total	11 / min
HTTP Error Codes	53 total	<1 / min
Error Page Redirects	0 total	<1 / min

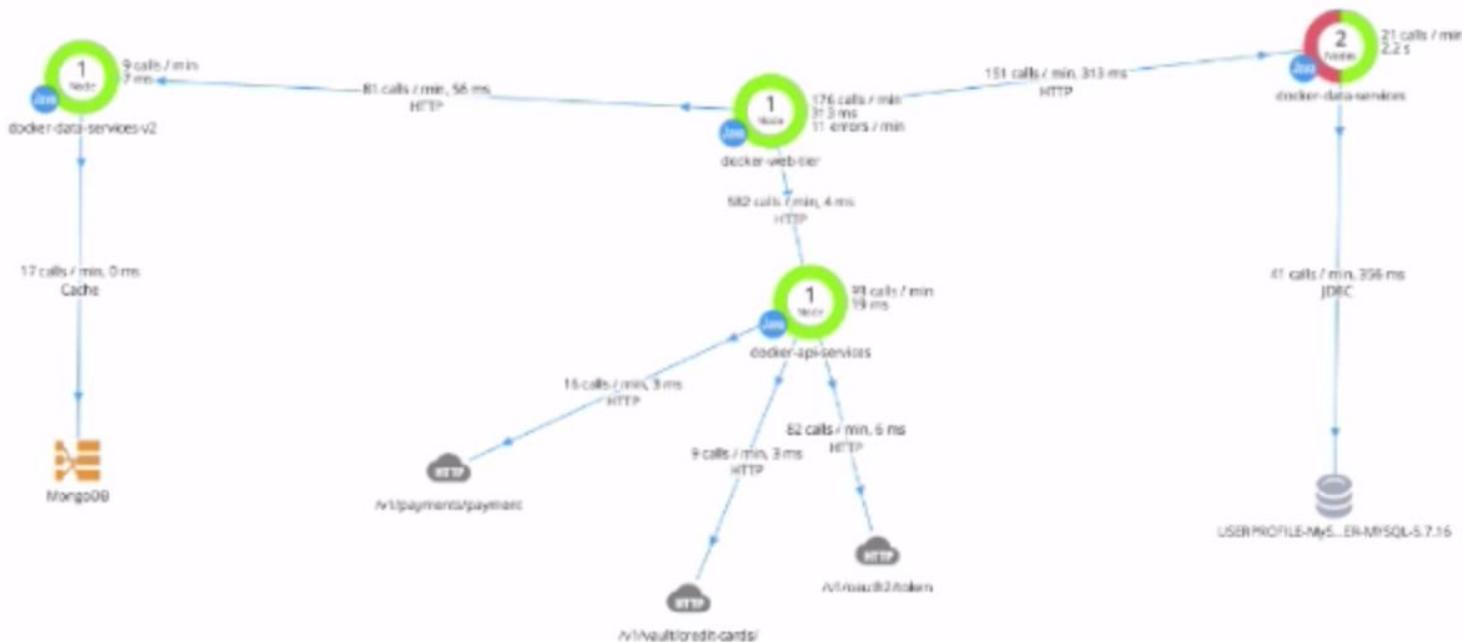
### Legend



Not comparing against: Baseline data

Application Flow Map

Go to custom timeframe and bring version 1 release



Timeframe selection menu:

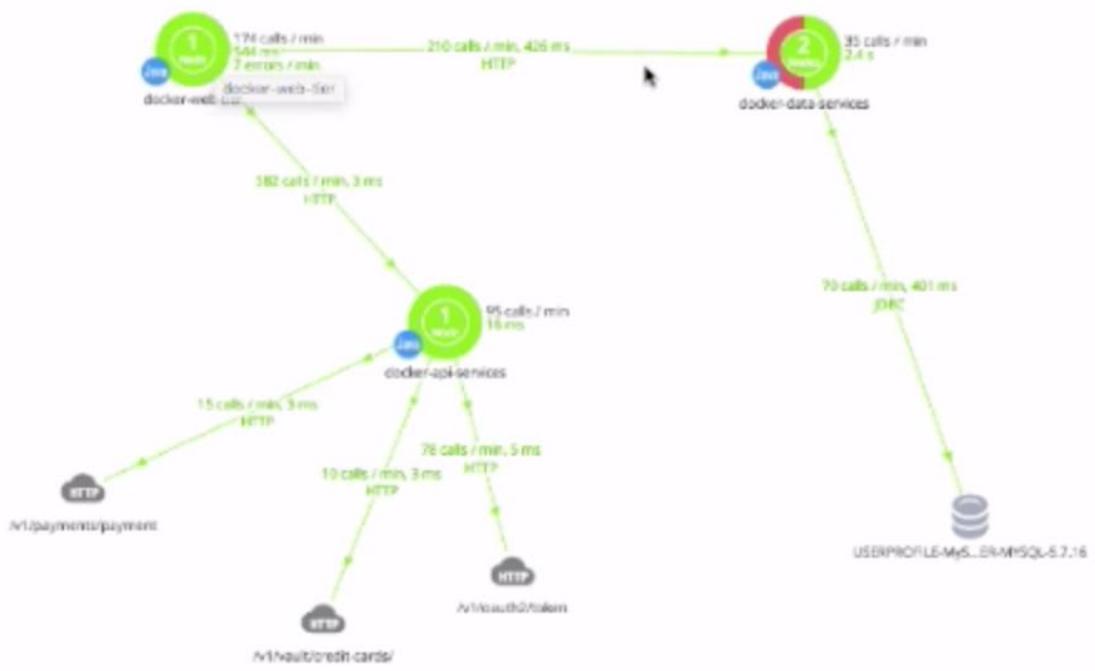
- 5 Minutes
- 15 Minutes
- 30 Minutes
- 1 Hour
- 2 Hours
- 3 Hours
- 4 Hours
- 5 Hours
- 12 Hours
- 1 Day
- 3 Days
- 1 Week
- 2 Weeks
- 1 Month
- 3 Month
- 5 Month
- 1 Year
- Custom

Exception list:

- Saved Time Ranges
- application-deploy-v1.0.530
- application-deploy-v2.0.528
- application-deploy-v1.0.521
- application-deploy-v2.0.529
- application-deploy-v1.0.532
- application-deploy-v2.0.530
- Manage Custom Time Ranges



Application Flow Map



Events

Health Rule Violations Started	45
Business Transaction Health	15
Node Health	30
Application Changes	1,160



Transaction Scorecard

Normal	64.1%	18.9k
Slow	0.8%	151
Very Slow	1.1%	226
Stall	0.0%	< 1
Errors	4.1%	815

Exceptions

Exceptions	859 total	7 / min
HTTP Error Codes	30 total	< 1 / min
Error Page Redirects	0 total	< 1 / min

Back end services : java micro services running inside Docker containers in amazon ec2 environment. AppD automatically discover and create dynamic maps in real time showing how transactions are flowing end to end.





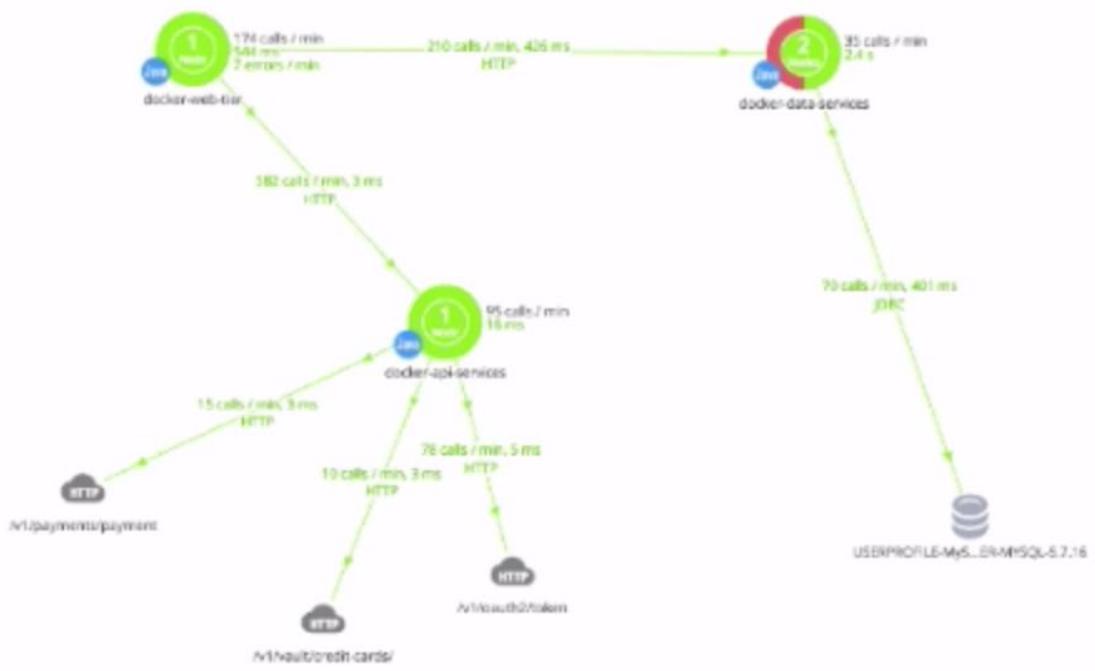
One of the customer in online payment space on average 95 distinct services are invoked for every customer request

Application Flow Map

### docker-data-services

Overview Nodes Servers Slowest DB and Remote Calls Business Transactions Incoming

- docker-data-node-1
- docker-data-node-2



#### Events

Health Rule Violations Started	45
Business Transaction Health	15
Node Health	30
Application Changes	1,160



#### Transaction Scorecard

Normal	64.1%	18.9k
Slow	0.8%	151
Very Slow	1.1%	226
Stall	0.0%	< 1
Errors	4.1%	815

#### Exceptions

Exceptions	859 total	7 / min
HTTP Error Codes	30 total	< 1 / min
Error Page Redirects	0 total	< 1 / min

Appd is the only platform which provide and to end picture of your system : googlemap for your won application



Application Flow Map

### docker-data-services

Overview Nodes Servers Slowest DB and Remote Calls Business Transactions Incoming

Breakdown of calls by Business Transaction

[View Business Transaction Dashboard](#)

Name	Res..	Calls	Call..	Err..	Err..
▼ /devops-payments-web/account/lookup					
docker-web-tier → (HTTP) → docker-data-services		2447	4031	35	4



8/17 7:01 AM - 08/08/17 8:59 AM

Events

- Health Rule Violations Started: 45
- Business Transaction Health: 15
- Node Health: 30
- Application Changes: 1,166

Business Transaction Health

1 critical, 6 warning, 7 normal

Node Health

1 critical, 6 warning, 5 normal

Servers

0 critical, 0 warning, 1 normal

Transaction Scorecard

Normal	64.1%	18,39
Slow	3.8%	151
Very Slow	1.1%	236
Stall	3.0%	< 1
Errors	4.1%	215

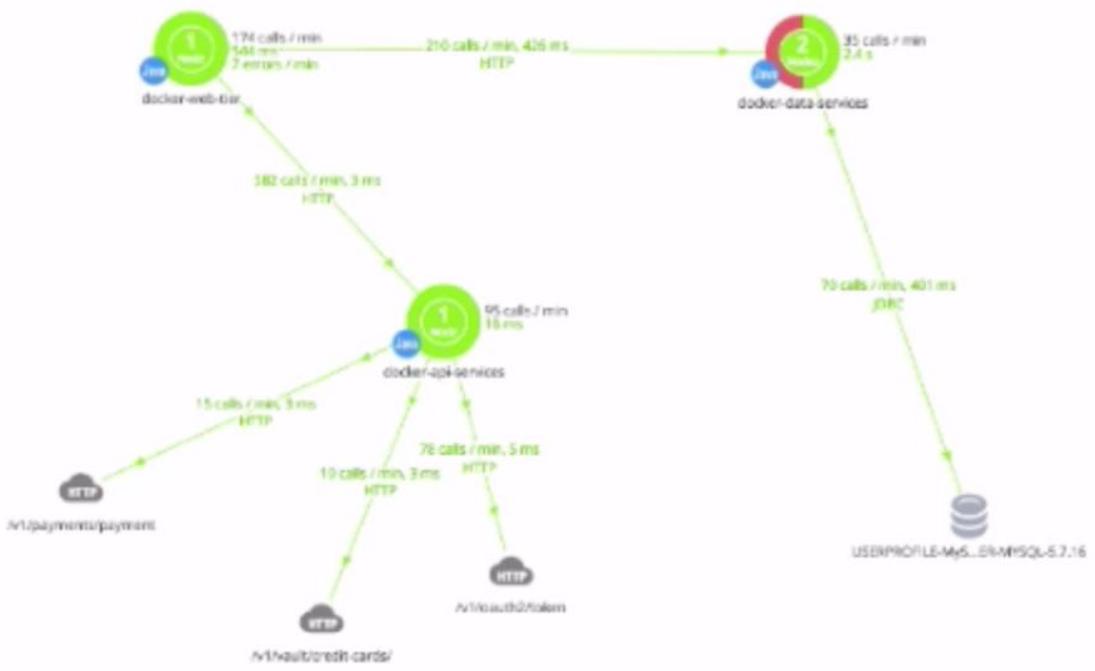
Exceptions

- Exceptions: 859 total / 7 / min
- HTTP Error Codes: 30 total / < 1 / min
- Error Page Redirects: 0 total / < 1 / min



# Payments Modernization

## Application Flow Map



### Events

Health Rule Violations Started	45
Business Transaction Health	15
Node Health	30
Application Changes	1,166

### Business Transaction Health

1 critical, 0 warning, 7 normal

### Node Health

1 critical, 0 warning, 9 normal

### Servers

0 critical, 0 warning, 1 normal

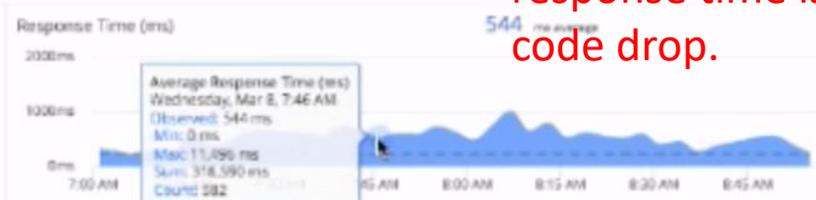
### Transaction Scorecard

Normal	64.1%	18/34
Slow	3.8%	151
Very Slow	1.1%	226
Stall	3.0%	< 1
Errors	4.1%	815

### Exceptions

Exceptions	859 total	7 / min
HTTP Error Codes	30 total	< 1 / min
Error Page Redirects	0 total	< 1 / min

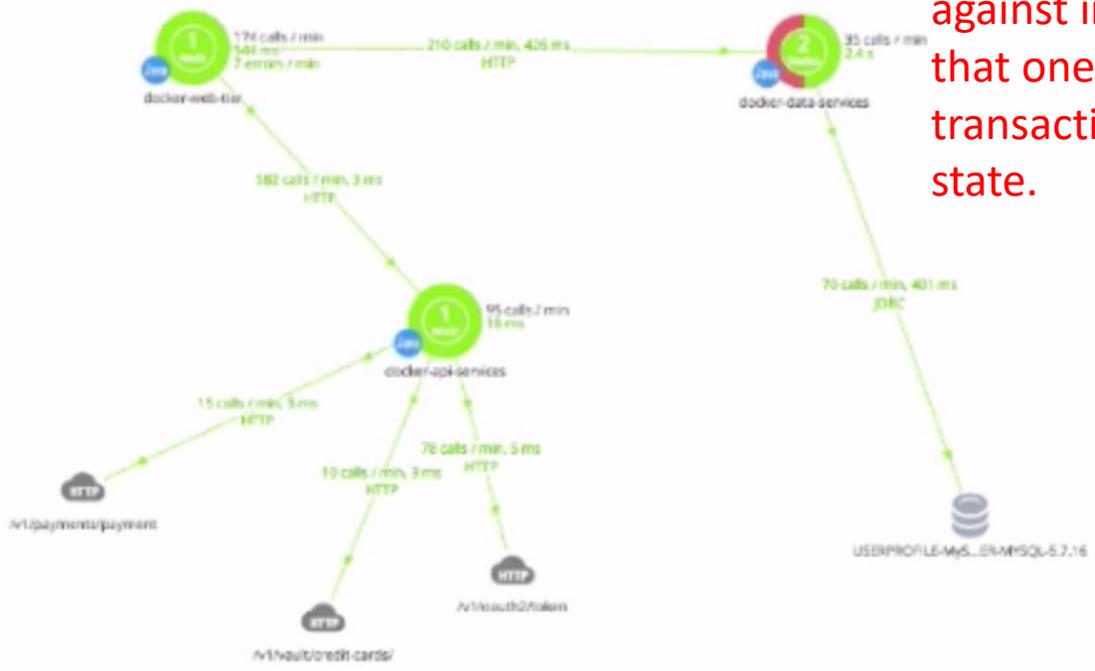
## Legend



AppD baseline technology allows to intelligently learn and proactively alert performance problems. In this example response time is above baseline after new code drop.

# Payments Modernization

## Application Flow Map



Real-time scoring of transactions against individual baseline show that one of our business transaction is in currently in critical state.

### Events

Health Rule Violations Started	45
Business Transaction Health	15
Node Health	30
Application Changes	1,160

### Business Transaction Health

1 critical, 0 warning, 7 normal

### Node Health

1 critical, 0 warning, 5 normal

### Servers

0 critical, 0 warning, 1 normal

### Transaction Scorecard

Normal	64.1%	18.9k
Slow	3.8%	151
Very Slow	1.1%	226
Stall	3.0%	< 1
Errors	4.1%	815

### Exceptions

Exceptions	859 total	7 / min
HTTP Error Codes	30 total	< 1 / min
Error Page Redirects	0 total	< 1 / min



Home Applications User Experience Databases Servers Analytics Dashboards & Reports Alert & Respond

Business Transactions

03/08/17 7:01 AM 03/08/17 8:50 AM

Details Filters Actions View Options Configure

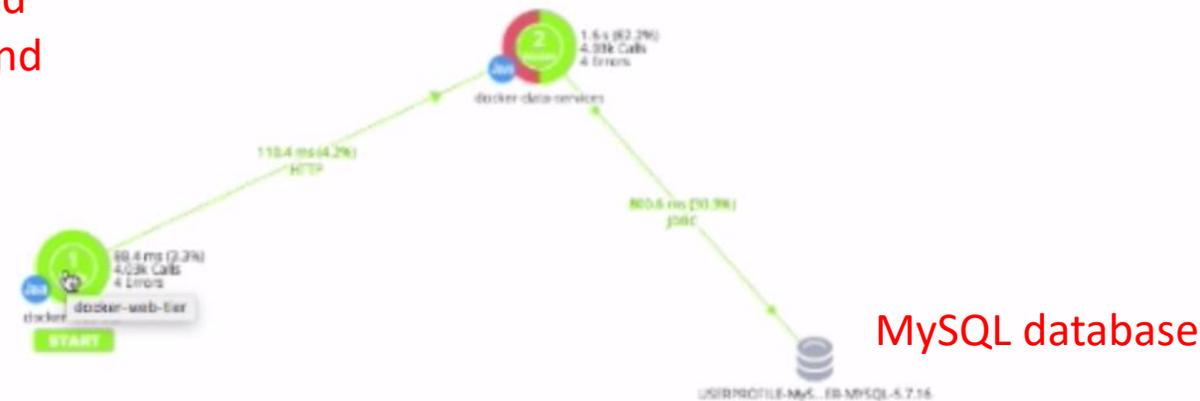
Showing 8 of 8

Name	Health	Response Time (ms) ↓	Calls / min	Errors / min	% Errors	% Slow Transactions	% Very Slow Transactions	% Failed Transactions	Block Time (ms)	Wait Time (ms)	Tier	Type
iceops-payments-web/accountLookup	<span style="color: red;">■</span>	2,644	35	0	0.1	3.5	5.3	0	0	0	€ docker-web-tier	Servlet
iceops-payments-web/checkout	<span style="color: green;">■</span>	280	2	-	-	-	1.2	-	0	0	€ docker-web-tier	Servlet
iceops-payments-web/accountHistory	<span style="color: green;">■</span>	31	13	0	0.1	0.1	0.2	-	0	0	€ docker-web-tier	Servlet
iceops-payments-web/addcard	<span style="color: green;">■</span>	29	10	0	0.2	0	0.3	-	0	0	€ docker-web-tier	Servlet
iceops-payments-web/authenticate	<span style="color: green;">■</span>	16	44	0	0	0.2	0.1	0	0	0	€ docker-web-tier	Servlet
iceops-payments-web/offers	<span style="color: green;">■</span>	12	7	7	100	-	-	-	0	0	€ docker-web-tier	Servlet
iceops-payments-web/logs-page	<span style="color: green;">■</span>	1	53	-	0	0	0	0	0	0	€ docker-web-tier	Servlet
/	<span style="color: grey;">■</span>	0	0	0	0	0	0	0	0	0	€ docker-web-tier	Servlet

AppD automatically discovers and categorize incoming requests into appropriate business transactions. This is critical because allows us to place emphasize on what is really important – how users experiencing and using site. Users are locking into their account information's, checking out, looking account history ... Each business transaction represent different work in system and each one is baseline individually. If there is problem appd tell us exact business function (transaction) which is degraded. We can drill in ...

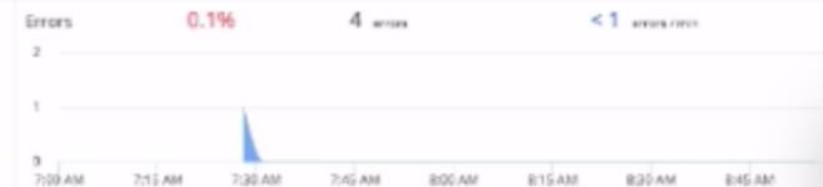
When we drill down into account lookup transaction appd shows us end to end transaction. Front end web tier

Back end data service running a docker



.. and understand what is happening with that transactions. If we need to engage support or dev team we can go to right team who owns account lookup service as opposed to get all dev team involved in war room.

Legend



Events	
Health Rule Violations Started	19
Business Transaction Health	19
Transaction Scorecard	
Normal	91.2% 3.7k
Slow	3.5% 141
Very Slow	5.3% 212
Stall	0.0% -
Errors	0.1% 4

Transaction Flow Map



Events

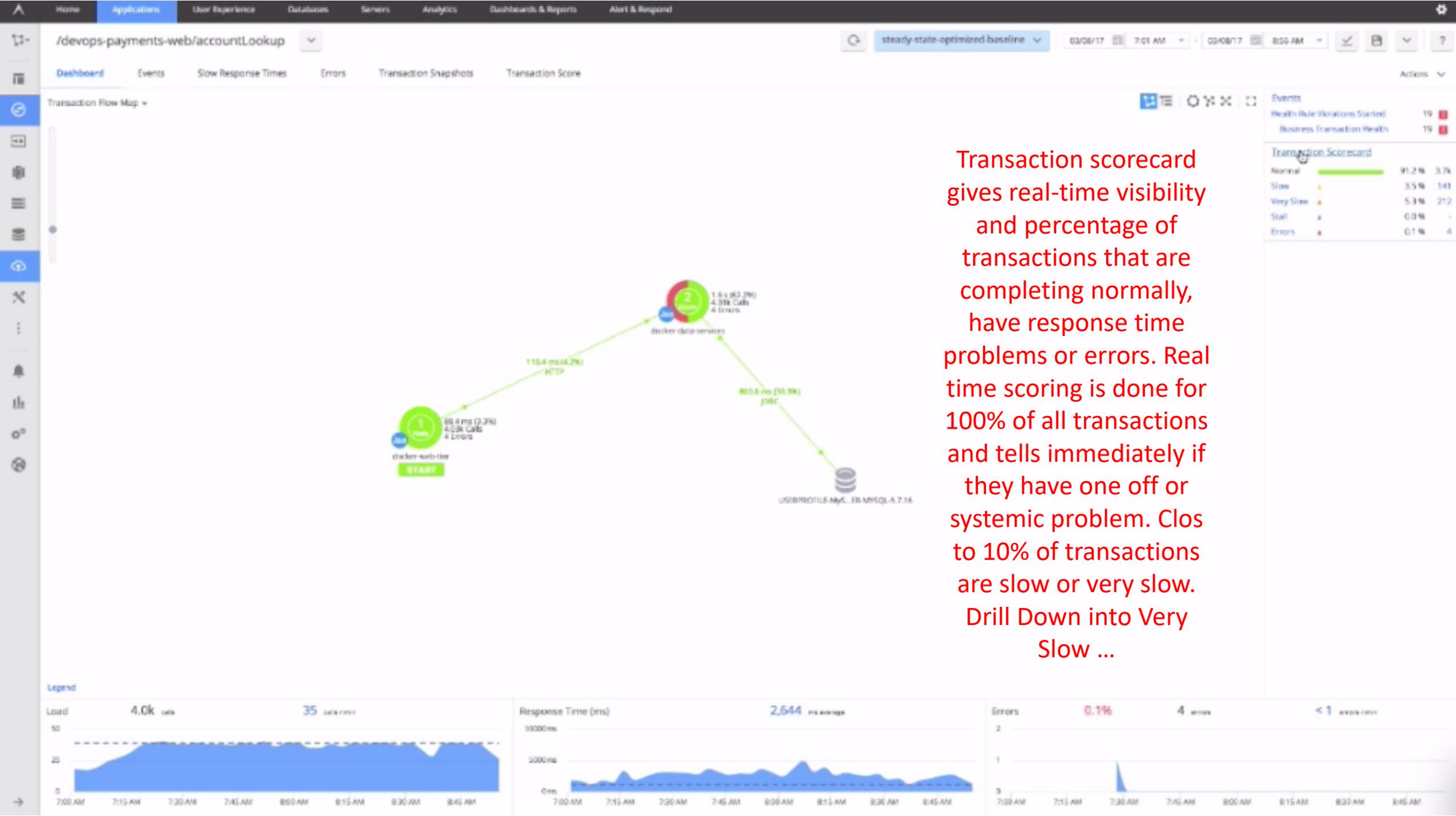
Health Rule Violations Started	19
Business Transaction Health	19

Transaction Scorecard

Normal	91.2%	3.7k
Slow	5.5%	141
Very Slow	5.3%	212
Stall	0.8%	-
Errors	0.1%	4

Average response time is not bad 2.6 Sec





Transaction scorecard gives real-time visibility and percentage of transactions that are completing normally, have response time problems or errors. Real time scoring is done for 100% of all transactions and tells immediately if they have one off or systemic problem. Clost to 10% of transactions are slow or very slow. Drill Down into Very Slow ...

Dashboard Events Slow Response Times Errors Transaction Snapshots Transaction Score

All Snapshots Slow and Error Transactions Diagnostic Sessions Periodic Collection

Loading..

Showing 212 of 212

+ Add Criteria User Experience (1) View Slow Business Transactions (1)

Time	Fast Time (ms)	URL	Business Transaction	Tier	Node
03/08/17 8:07:22 AM	18,755	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:42:04 AM	18,523	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:42:02 AM	17,912	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:40:37 AM	17,422	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:08:47 AM	17,141	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:07:23 AM	17,040	/devops-payments-web/accountLo	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:30:53 AM	16,840	/devops-payments-web/accountLo	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:10:05 AM	16,782	/devops-payments-web/accountLo	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:40:37 AM	16,589	/devops-payments-web/accountLo	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:08:48 AM	16,250	/devops-payments-web/accountLo	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:41:09 AM	16,095	/devops-payments-web/accountLo	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:07:24 AM	15,994	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:30:56 AM	15,752	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:30:56 AM	15,704	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:40:41 AM	15,406	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:10:07 AM	15,171	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:51:20 AM	15,090	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:42:06 AM	15,073	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:21:44 AM	15,063	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:08:48 AM	14,999	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 8:21:47 AM	14,877	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2
03/08/17 7:30:54 AM	14,842	/devops-payments-web/accountLookup	/devops-payments-web/accountLookup	docker-web-tier	docker-web-node-2

Average transaction response time is 2.6 sec. Looking into Transactions snapshot we see number of outliers running at 15+ sec all the way up to 19 sec. Drill down into specific transactions.

Transaction: 4d069eb-9be1-4ed8-8328-8cd011c2b022

Overview Slow Calls and Errors Waterfall View Segment List Actions

Summary Transaction Flow Map

User Experience: Very Slow

Execution Time: 18.8 s

Time Stamp: 03/08/17 8:07:22 AM

Business Transaction: id-evops-payments-web/accountLookup

Potential Issues

com.appdynamics.sample.servic.PaymentModernizationService.invokeService	18.7 s
http://devops-data-services-mysql:8080/devops-legacy-data-services/data/v1/accountProfile/95248	18.3 s
com.appdynamics.sample.service.AccountProfileMySQL.getAccountProfile	18.3 s
select c.* from customers c, add_rest_book a where c.customer_id = a.customer_id and c.first_name like 'kg%' limit 50248	500 ms
org.apache.jsp.jsp_response_jsp.jspService	32 ms

Transaction Flow Map

```
graph TD; docker-web-tier[1 Node] -- "11 ms (0.1%) HTTP" --> docker-data-services[2 Nodes]; docker-data-services -- "999 ms (2.2%) JDBC" --> USERPROFILE-MySQL-DB-MYSQL-5.7.16[1 Call];
```

Legend

Drill Down

Down

11 ms (0.1%) HTTP

999 ms (2.2%) JDBC

USERPROFILE-MySQL-DB-MYSQL-5.7.16

1 Node

2 Nodes

1 Call

1 Call

AppD automatically scanned through each segment of transaction request. Operations or support engineers who might not written the code. AppD provide intelligence to automatically highlight things like : Degraded device indication, long running methods, poor SQL queries. Drill Down into call graph to see source code ...

Transaction: 4d409eb-0be1-4ed8-8320-80d011c29020 | 18,715 ms - docker-web-tier

Overview | Call Graph | Slow Calls & Errors | DD & Remote Service Calls | Server | Data Collectors | More

Execution Time: 18.8 s | Wait Time: 0 ms | Rock Time: 0 ms | Node: docker-web-node-2 | Timestamp: 03/08/17 8:07:22 AM

Details | Filter | Set Icon | Reset Icon | Export

Name	Time (ms)	Percent %	Thread State	Exit Calls / Threads...
Servlet - com.appdynamics.sample.servlet.AccountProfileServlet.doGet	1 ms (self)	0%		
com.appdynamics.sample.servlet.PaymentsModernizationServlet.processUserRequest:236	0 ms (self)	0%		
com.appdynamics.sample.servlet.PaymentsModernizationServlet.processRequest:246	0 ms (self)	0%		
Servlet - com.appdynamics.sample.servlet.AccountProfileServlet.processRequest:130	0 ms (self)	0%		
Servlet - com.appdynamics.sample.servlet.AccountProfileServlet.invokeAccountService:88	0 ms (self)	0%		
com.appdynamics.sample.servlet.PaymentsModernizationServlet.invokeRestService:127	18,721 ms (99.8%)	99.8%		HTTP (0)
HTTPServletService:731	0 ms (self)	0%		
HTTPServletService:731	0 ms (self)	0%		
JSPBaseServletService:70	0 ms (self)	0%		
Servlet - response_ppi_jspService:12	32 ms (self)	0.2%		

Some packages have been excluded from this Call Graph

Start with the web tier and we can see all the method executed ... web tier is making a downstream rest service to one or more service

Transaction: 4d4629e0-9be1-4ede-8329-0c0d011c28020 18,715 ms - docker-web-cl...

Overview Call Graph Slow Calls & Errors DB & Remote Service Calls Server Data Collectors More

Execution Time 18.8 s Wait Time 0 ms Rock Time 0 ms Node docker-web-node-2 Timecamp 03/08/17 8:07:22 AM

Detail Filter Set Size Server Post Export Showing 10 of 10

Name	Time (ms)	Percent %	Thread State	Exit Calls / Threads...
Servlet - com.appdynamics.sample.servlet.AccountProfileService.doGet	1 ms (self)	0%		
com.appdynamics.sample.servlet.PaymentsModernizationServlet.processUserRequest:235	0 ms (self)	0%		
com.appdynamics.sample.servlet.PaymentsModernizationServlet.processRequest:246	0 ms (self)	0%		
Servlet - com.appdynamics.sample.servlet.AccountProfileServlet.processRequestImpl:136	0 ms (self)	0%		
Servlet - com.appdynamics.sample.servlet.AccountProfileServlet.invokeAccountService:8	0 ms (self)	0%		
com.appdynamics.sample.servlet.PaymentsModernizationServlet.invokeRestService:	18,721 ms (98.8%)	98.8%		HTTP (0)
HTTPService:service:731	0 ms (self)	0%		
HTTPService:service:731	0 ms (self)	0%		
JSPBaseServlet:service:70	0 ms (self)	0%		
Servlet - response.jsp_jsp:service:412	32 ms (self)	0.2%		

InvokeRestService:127

Name: com.appdynamics.sample.servlet.PaymentsModernizationServlet.invokeRestService  
 Type: POJO  
 Class: com.appdynamics.sample.servlet.PaymentsModernizationServlet  
 Method: invokeRestService  
 Line Number: 127

Self Time: 18,721 ms 98.8%  
 Total time: 18,721 ms 98.8%

Details: CPU: 146 ms, Block: 0 ms, Wait: 0 ms

Exit Calls

Type	Details	Count	Time (ms)	% Time	From	To
HTTP	http://devops-data-services-mysql:8080/devop...	1	18,805	92.6%	docker-web...	docker-data...
HTTP	http://devops-data-services-mysql:8080/devop...	1	0	0%	docker-web...	docker-data...
HTTP	http://devops-data-services-mysql:8080/devop...	1	0	0%	docker-web...	docker-data...

HTTP Exit Call

Time: 18,205ms  
 From: docker-web-tier  
 To: docker-data-services  
 Details: http://devops-data-services-mysql:8080/devops-legacy-data-services/data/v1/accountProfile/96248

Drill Down Into Downstream Call

Some packages have been excluded from this Call Graph

03/08/17 7:30:54 AM 14,842 devops-payments-web-cl... docker-web-tier docker-web-node-2

We can drill into http calls and see that one call is critically degraded. 98% of time or 18 sec spent calling account profile services. Drill Down into the call.

Transaction: 46667e0-9be1-4ede-8328-96d011c2b559 | 18,755 ms - docker-web-c... | 18,294 ms - docker-data-s...

Overview | Call Graph | Slow Calls & Errors | DB & Remote Service Calls | Server | Data Collectors | More

Execution Time: 18.3 s | Wait Time: 0 ms | Back Time: 0 ms | Node: docker-data-node-2 | Timecamp: 05/08/17 8:07:22 AM

Details | Filter | Set Acc | View Acc | Export

Name	Time (ms)	Percent %	Thread State	Exit Calls / Threads...
com.appdynamics.sample.service.DataServices.getUserProfile	0 ms (self)	0%		
com.appdynamics.sample.service.MySQLDataServices.getUserProfileImpl:54	0 ms (self)	0%		
com.appdynamics.sample.service.AccountProfileMySQL.getAccountProfile:74	18,293 ms (s...)	100%		3/3C (1)

com.appdynamics.sample.service.AccountProfileMySQL.getAccountProfile:74

Some packages have been excluded from this Call Graph

05/08/17 7:30:54 AM | 14,842 | Adveps-payments-web/accountlookup | Adveps-payments-web/accountlookup | docker-web-001 | docker-web-node-2

Looking into the code running at downstream service code stack is very simple and we spent a lot of time to get account profile method at line 74.

It seems we are making two distinct calls back to mySQL database. Drilling into a call ...

Transaction: 46469e0-0be1-4ed9-8328-0c0011c2b003 | 18,755 ms - docker-web-cl... | 18,204 ms - docker-data-s...

Overview | Call Graph | Slow Calls & Errors | DB & Remote Service Calls | Server | Data Collectors | More

Execution Time: 18.3 s | Wait Time: 0 ms | Rock Time: 0 ms | Node: docker-data-node-2 | Timestamp: 05/08/17 8:07:22 AM

Details | Filters | Set Icon | Smart Scroll | Export

Name | Time (ms) | Percent % | Thread State | Exit Calls / Threads

- com.appdynamics.sample.service.DataServices.getUserProfile | 0 ms (self) | 0% | |
- com.appdynamics.sample.service.MySQLDataServices.getUserProfileImpl04 | 0 ms (self) | 0% | |
- com.appdynamics.sample.service.AccountProfileMySQL.getAccountProfile:74** | **18,293 ms (s...)** | **100%** | | JDBC (2)

**getAccountProfile:74**

Name: com.appdynamics.sample.service.AccountProfileMySQL.getAccountProfile  
 Type: POJO  
 Class: com.appdynamics.sample.service.AccountProfileMySQL  
 Method: getAccountProfile  
 Line Number: 74

Self Time: 18,293 ms | 100%  
 Total Time: 18,293 ms | 100%  
 Details: CPU: 598 ms, Block: 0 ms, Wait: 0 ms

**Exit Calls**

Type	Details	Count	Time (ms)	% Time	From	To
JDBC	select c.* from customers c, address_book a w...	1	593	3.2%	docker-data...	USERPROFL...
JDBC	select c.* from customers c, address_book a w...	96248	8476	45.3%	docker-data...	USERPROFL...

**JDBC Exit Call**

Time: 593ms  
 From: docker-data-services  
 To: USERPROFILE-MYSQL-DB-01VOPS-USER-MYSQL-5.7.10  
 Details: select c.\* from customers c, address\_book a where c.\* customers\_firstname like 'John' limit 96248

**Properties**

ResultSet: Count=96248 Iteration Time=8476 ms  
 Statement Type: Prepared Statement  
 Query Type: Query

Some packages have been excluded from this Call Graph

05/08/17 7:30:54 AM | 14,842 | devops-payments-web/accountlookup | devops-payments-web/accountlookup | docker-web-clr | docker-web-node-2

AppD captured initial SQL query and dev team let a poor written query using a wildcard in production. Query returns over 96k results. Code is iterating through retrieving and processing every results one at a time. We call this death by a 1k queries and it's causing major issue with account lookup function.

Transaction: 48469e0-5be1-4ed8-8328-00d011c2b053 18,755 ms - docker-web-6... 18,204 ms - docker-data-s...

Overview Call Graph Slow Calls & Errors DB & Remote Service Calls Server Data Collectors More

Execution Time 18.3 s Wait Time 0 ms Block Time 0 ms Node docker-data-node-2 Timestamp 05/06/17 8:07:22 AM

Details Filter Set Icon Search Icon Export

Name	Time (ms)	Percent %	Thread State	Exit Calls / Threads...
com.appdynamics.sample.service.DataServices.getUserProfile	0 ms (self)	0%		
com.appdynamics.sample.service.MySQLDataServices.getUserProfileImpl54	0 ms (self)	0%		
com.appdynamics.sample.service.AccountProfileMySQL.getAccountProfile:74	18,293 ms (b...)	100%		3/3C (2)

Some packages have been excluded from this Call Graph.

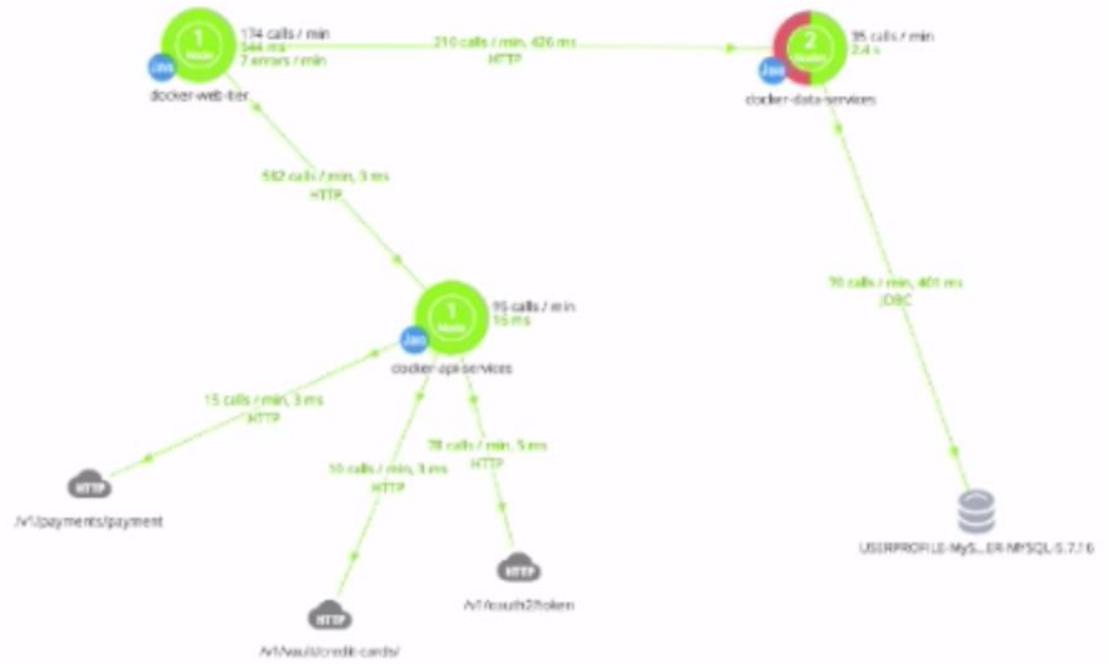
05/06/17 7:30:54 AM 14,842 /devops-payments-web/account/lookup /devops-payments-web/account/lookup docker-web-10r docker-web-node-2

Copied to clipboard  
https://devops.demo.appdynamics.com:8090/controlpanel/location=APP\_SNAPSHOT\_VIEW&requestURL=48469e0-5be1-4ed8-8328-00d011c2b056&application=75/rsd?time=Custom\_Time\_Range.RETRYER\_TIMES\_1468963542741.1468963542741.606.tab=35522457&dashboardMode=force

We can easily grab a link and share findings with teams

# Payments Modernization

## Application Flow Map



After fix is pushed in new release

## Events

Health Rule Violations Started	45
Business Transaction Health	15
Node Health	30
Application Changes	1,160

## Business Transaction Health

1 critical, 0 warning, 7 normal

## Node Health

1 critical, 0 warning, 5 normal

## Servers

0 critical, 0 warning, 1 normal

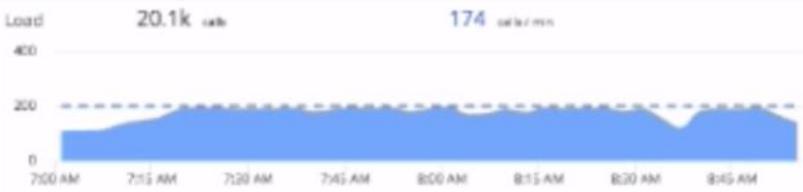
## Transaction Scorecard

Normal	64.1%	18/28
Slow	0.8%	151
Very Slow	1.1%	276
Stall	0.0%	<1
Errors	4.1%	815

## Exceptions

Exceptions	859 total	7 / min
HTTP Error Codes	30 total	<1 / min
Error Page Redirects	0 total	<1 / min

## Legend



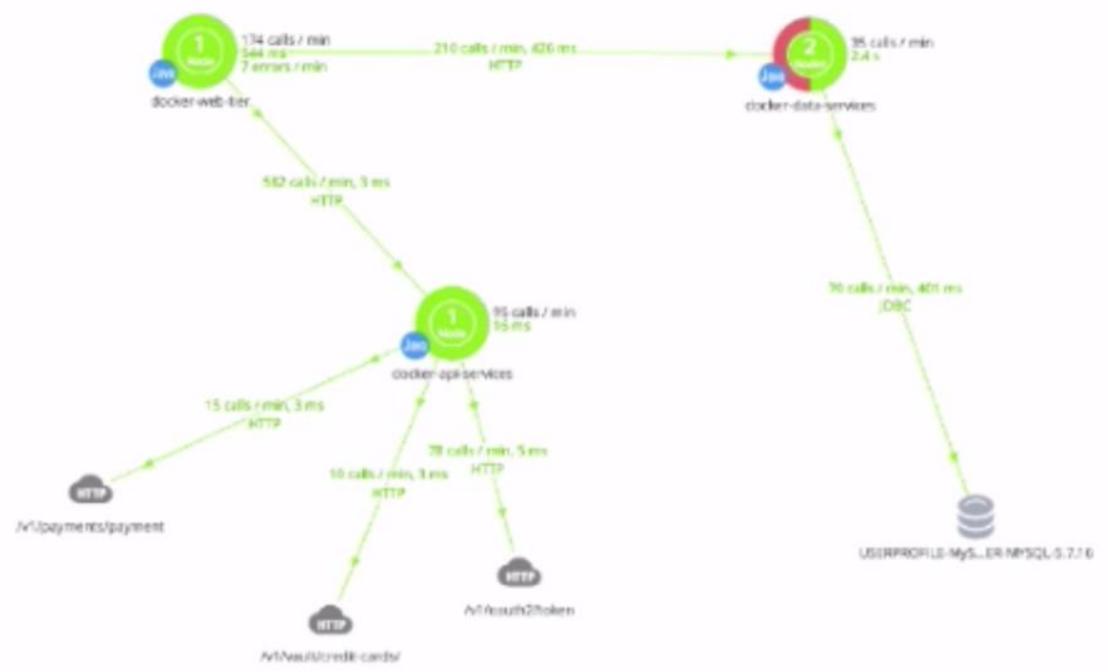
# Payments Modernization

steady-state-optimized baseline

02/08/17 7:01 AM - 02/08/17 8:50 AM

Dashboard Events Top Business Transactions Transaction Snapshots Transaction Score

Application Flow Map



- 5 Minutes
- 15 Minutes
- 30 Minutes
- 1 Hour
- 2 Hours
- 3 Hours
- 4 Hours
- 6 Hours
- 12 Hours
- 1 Day
- 3 Days
- 1 Week
- 2 Weeks
- 1 Month
- 3 Month
- 6 Month
- 1 Year
- Custom
- Saved Time Ranges
  - application-deploy-v1.0.530
  - application-deploy-v2.0.528
  - application-deploy-v1.0.531
  - application-deploy-v2.0.529
  - application-deploy-v1.0.532
  - application-deploy-v2.0.530
  - Manage Custom Time Ranges

We can verify fix

Legend

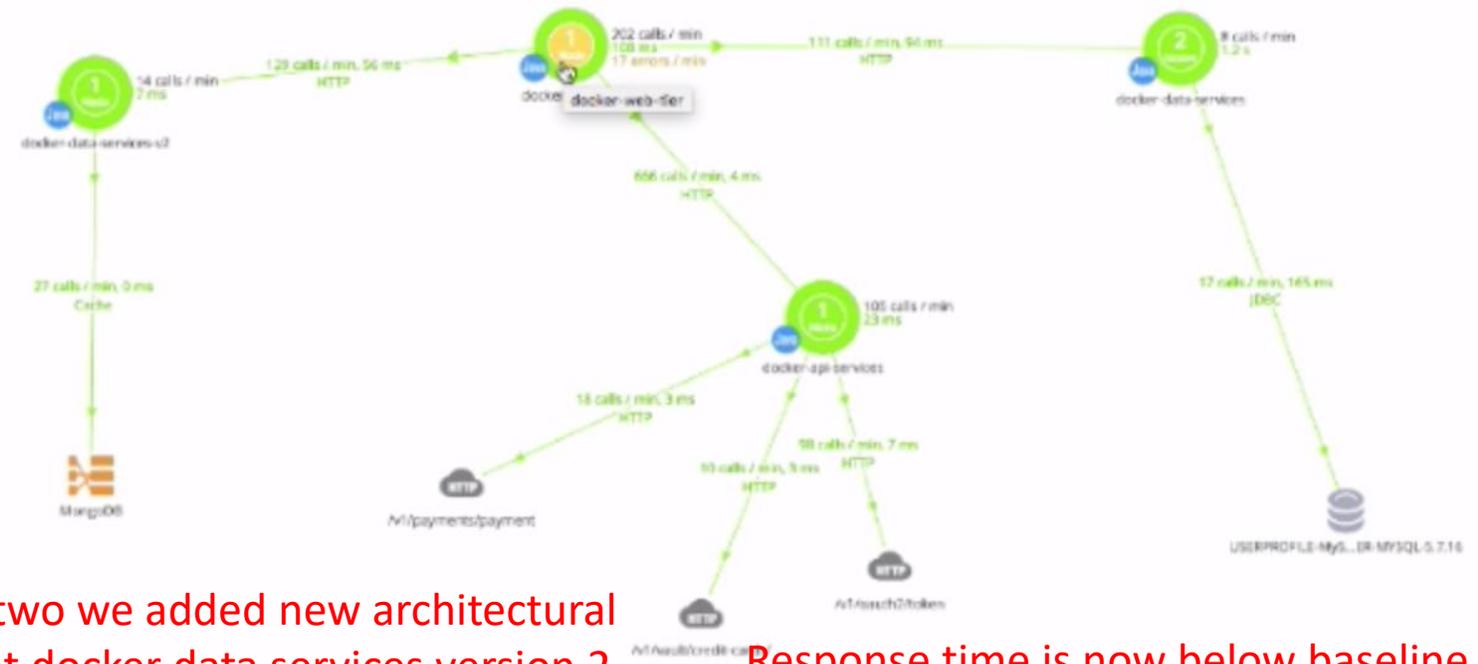


# Payments Modernization

- Dashboard
- Events
- Top Business Transactions
- Transaction Snapshots
- Transaction Score

Actions

## Application Flow Map



**Events**  
Application Changes: 1,112

**Business Transaction Health**  
0 critical, 0 warning, 6 normal

**Node Health**  
0 critical, 0 warning, 10 normal

**Servers**  
0 critical, 0 warning, 1 normal

**Transaction Scorecard**

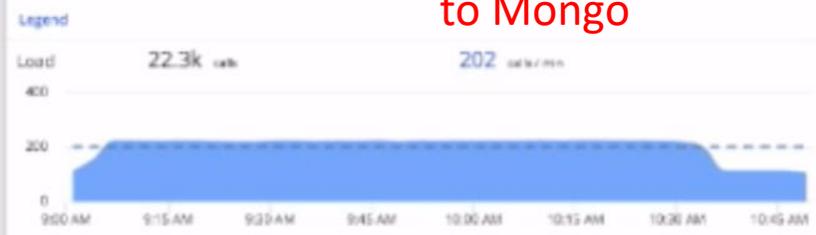
Normal	61.3%	20.3k
Slow	0.1%	23
Very Slow	0.1%	28
Stall	0.6%	< 1
Errors	8.5%	1.8k

**Exceptions**

Exceptions	1,883 total	1.7 / min
HTTP Error Codes	23 total	< 1 / min
Error Page Redirects	0 total	< 1 / min

In version two we added new architectural component docker data services version 2 which calls into a back end mongo instance. We moved account functionality from mySQL to Mongo

Response time is now below baseline which improve overall user experience.



Navigation bar: Home, Applications, User Experience, Databases, Servers, Analytics, **Dashboards & Reports**, Alert & Respond

### Dashboards & Reports

**Dashboards 4**

- DevOps Leadership Dashboard
- DevOps Release Analytics v1.0**
- Docker-DockerEngine
- Offer Conversion Analysis

+ Create Dashboard

**Reports 0**

No Scheduled Reports

+ Create Report

Jump into release analytics dashboard

DevOps Release Analytics v2.0

APPDYNAMICS

ACTIVE USER SESSIONS      % USER CONVERSION RATE      REVENUE BY VERSION

Conversion increased from 4% to 14%)

Version	Active User Sessions	% User Conversion Rate	Revenue
Version 1.0	3,625	4.83	14,148
Version 2.0	3,206	14.41	61,508.5

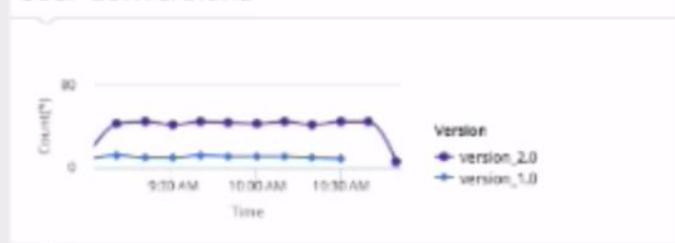
Fewer users pushed to version 2 (3206)

Revenue increased 4X

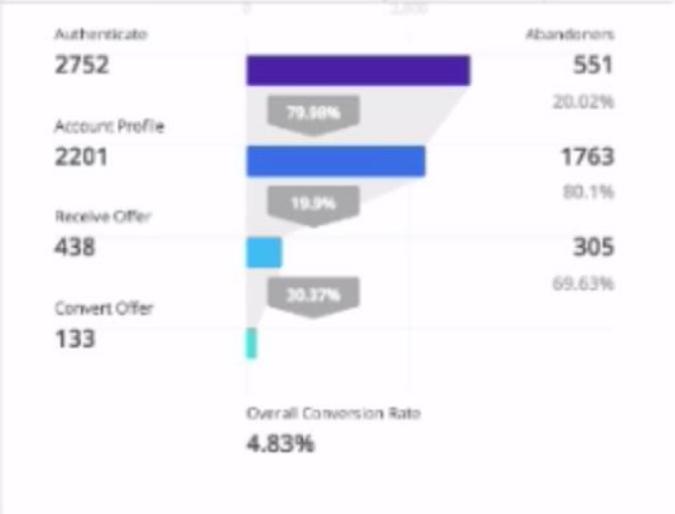
User Experience



User Conversions



Conversion Funnel (Ver 1.0)



Conversion Funnel (Ver 2.0)

