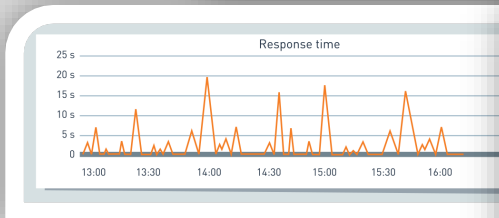




## State of the Art of Visualization in APM Tools



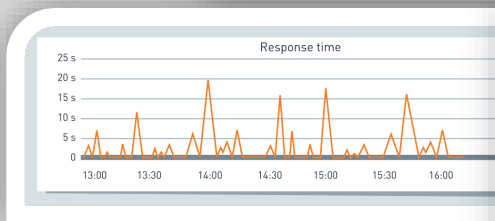
Trace	Response time
▼ doFilter(...)	[1793 ms]
▼ searchTitleAndDescription(...)	[1730 ms]
▼ searchTitleAndDescriptionWithOneWord(...)	[1632 ms]
▼ list(...)	[352 ms]
■ executeQuery() {SQL: Select PROD_ID ...}	[143 ms]
■ executeQuery() {SQL: Select INV_ID ...}	[12 ms]
▼ ...	...
...	...

André van Hoorn  
Dušan Okanović



# State of the Art of Visualization in APM Tools

## Part 1/2: Introduction to APM



Trace	Response time
▼ doFilter(...)	[1793 ms]
▼ searchTitleAndDescription(...)	[1730 ms]
▼ searchTitleAndDescriptionWithOneWord(...)	[1632 ms]
▼ list(...)	[352 ms]
■ executeQuery() {SQL: Select PROD_ID ...}	[143 ms]
■ executeQuery() {SQL: Select INV_ID ...}	[12 ms]
▼ ...	...
...	...

André van Hoorn  
Dušan Okanović

Alle Events >

Alle Orte >

Suche nach Künstlern und Events

Suchen

Warenkorb

- Konzerte >
- Kultur >
- Sport >
- Musical & Show >
- Humor >
- Eventreisen >
- Mehr... >
- Special Tickets >

**Ed Sheeran**  
NEUE KONTINGENTE 23.07. GELSENKIRCHEN

TICKETALARM

VORVERKAUF AB: MO. 09.07. | 10 UHR

**RABATTE**

DER EVENTIM NEWSLETTER

**ELVIS**  
DAS MUSICAL

EXKL. VORVERKAUF

DVD AUFEICHNUNG  
IN HAMBURG

DIE STADION-TOURNEE 2018

**HELENE FISCHER**

NEUE KONTINGENTE HAMBURG

© Universal Music / Kristian Schuller

HOLIDAY  
ON  
**ICE** 75  
JAHRE

SEI DABEI, WENN WIR  
75 JAHRE FEIERN!

NEUE SHOW-ERLEBNIS BEHÄLTET!

10 JAHRE  
**Andreas Gabalier**

DIE STADION TOUR ZUM JUBILÄUM 2019

© Stahl Records / Marie Christians

EVENTIM Charts

**Andreas Gabalier**  
★★★★★  
Tickets ab € 49,90

**Mark Forster**

02 01

Tipps (16)

Die Produzenten von  
**Riviera**  
WÄNDERN  
DIE NEUE SHOW  
**Heartbeat of home**

© Lions Limited

**JUDI JACKSON**

ZUSATZTERMINE  
**UDO LINDBENBERG**  
LIVE 2019

**SEH-**

# OTTO

## Übersicht

### Übersicht

Tickets

Fan-Reports

Biografie

Gefällt mir

Teilen



### Vorhang auf: OTTO bei eventim.de

OTTO – der Name ist Programm: Ostfriesisch Temperamentvoll Total Onverwüthlich. „Otto Live“ ist weiter auf Tour. Die Einschätzung einiger Kritiker, Otto wäre der beste Bühnenkomiker seiner Generation, ist natürlichbarer Blödsinn. Wahr ist allerdings, dass es weit und breit niemanden gibt, der ihm auf der Bühne das Wasser reichen kann – schon weil er wie immer ganz allein auftreten muss. Andere werfen Otto vor, dass er aus seinem Talent bisher zu wenig gemacht habe – auch die dürfen beruhigt sein: Er arbeitet ja weiter daran.

### Ticketalarm – kein Event mehr verpassen!

Registrieren Sie sich für den eventim.de-Ticketalarm, und Sie werden per E-Mail informiert, sobald es neue Termine Ihrer Lieblingsstars und -events gibt.

REGISTRIEREN SIE SICH JETZT FÜR OTTO

Jetzt anmelden ▶

Ich möchte den Ticketalarm des aufgeführten Künstlers abonnieren. Hierzu darf CTS EVENTIM AG & Co. KGaA meine E-Mail-Adresse verwenden. Die Abmeldung des Ticketalarms ist jederzeit möglich. EVENTIM legt großen Wert auf Datenschutz. Die Datenschutzerklärung können Sie [hier](#) nachlesen.

### Über diesen Künstler

#### Künstler-Biografie

Otto Gerhard Waalkes, häufig einfach nur Otto genannt, (\* 22. Juli 1948 in Emden) ist ein deutscher Komiker, Comiczeichner, Musiker, Schauspieler, Regisseur und Synchronsprecher. Der gebürtige Ostfrieser gilt als einer der erfolgreichsten Vertreter des deutschen Humors. Im Jahr 2007 kam Otto Waalkes bei der Wahl zum besten deutschsprachigen Komiker ...

▶ [weiterlesen](#)



# MOMENT, BITTE!

SIE BEFINDEN SICH IM WARTERAUM VON EVENTIM,  
BITTE VERSUCHEN SIE ES SPÄTER WIEDER.



“Application performance management (APM), as a core IT operations discipline, aims to achieve an adequate level of performance during operations. To achieve this,

APM comprises methods, techniques, and tools for

- continuously monitoring the state of an application system and its usage, as well as for
- detecting, diagnosing, and resolving performance-related problems using the monitored data.”

”

**Application Performance Management: State of the Art and Challenges for the Future**

Christoph Heger,<sup>1</sup> André van Hoorn,<sup>1</sup> Mario Mann,<sup>1</sup> Dusan Okanović,<sup>2</sup>  
<sup>1</sup>Novus Consulting GmbH, Compusera Area APM, Leibnizstr. 10, 30559 Hannover, Germany  
<sup>2</sup>University of Duisburg-Essen, Institute of Software Technology, Duisburg, Germany

**ABSTRACT**

The performance of application systems has a direct impact on business activities. For example, companies face customer and revenue loss in case of poor performance such as high response times. Application performance management (APM) aims to provide the required processes and tools to have a continuous and up-to-date picture of relevant performance measures during operations, as well as to support the detection and resolution of performance-related incidents.

In this featured paper, we provide an overview of the state of the art in APM in industrial practice and outline research, highlight current challenges, and outline future research directions.



Figure 1: Continuous APM activities

**1. INTRODUCTION**

Business systems are directly influenced by the performance of the software applications systems that support it. Any performance issue that may arise during the productive use of such applications may bring losses in revenue, and even cause customer to leave users. Examples of these losses and their impact are well-documented. Google loses 20% of its daily revenue when Google Maps is down [1], Amazon loses 1% of its revenue for every minute in latency [2]. Statistics clearly indicate that if the page is not loaded within one to five seconds, users will leave the web site [3].

Application performance management (APM), as a core IT operations discipline, aims to achieve an adequate level of performance during operations. To achieve this, APM comprises methods, techniques, and tools for (i) continuously monitoring the state of the application system and its usage, as well as for (ii) detecting, diagnosing, and resolving performance-related problems using the monitored data.

In this paper, we provide a state-of-the-art overview of the current APM activities (Section 2) and state (Section 3), and highlight some well-challenged and future research directions (4).

**2. APM ACTIVITIES**

Regardless of the varied technical realizations, APM to solve the following four – conventionally conducted and inter-related – activities (Figure 1):

1. Data collection. Performance measures are collected from the different system tiers, layers, and locations by a combination of complementary techniques and technologies.
2. Data storage and processing. The collected data is consolidated, aggregated, and data structures such as time series or correlation graphs.
3. Data presentation. Data is made available for visual access to an different levels of abstraction and detail.
4. Data interpretation and use. The data is used to manually or automatically reason about and act upon the current state.

The remainder of this section details these activities.

**2.1 Data Collection**

Various types of performance-related measures can and need to be collected from systems and locations. In our definition of APM regarding data collection is where, what, and how to collect, or related to the remainder of this section and depicted in Figure 2.

**2.1.1 Where to Collect Data?**

Multiple application system are monitored, highly distributed, multi-layered, and accessed via different types of clients and devices (e.g., third-party systems and browser using desktop or mobile devices).

Permission is granted to reproduce this paper for non-commercial purposes. This paper is part of the Proceedings of the ACM on Measurement and Analysis of Computing Systems, Vol. 1, Article 1. Copyright 2017 ACM 978-1-4503-4531-2/17/0001...\$15.00. DOI: <https://doi.org/10.1145/3000000>

C. Heger, A. van Hoorn, D. Okanović, M. Mann:  
*Application performance management: State of the art and challenges for the future.*  
 In: Proc. 8<sup>th</sup> ACM/SPEC ICPE, ACM (2017)





# Application Performance Management (APM)

Kontinuierliche Überwachung von Anwendungsperformance

supported by Inhäliche Entwicklung: Dr.-Ing. André van Hoorn (Prof.-Vert.) und Stefan Siegl

## Einfluss von Performance auf Erfolg

Anwendungsperformance hat direkte Auswirkungen auf den Unternehmenserfolg.

> Z. B. Wahrnehmung, Vertrauen, Umsatz



## Konsequenzen schlechter Performance



## Application Performance Management

Durch APM-Werkzeuge und -Prozesse kann Anwendungsperformance kontinuierlich überwacht und sichergestellt werden.

...wichtig im gesamten Software-Lebenszyklus



...relevant für alle Systemebenen



...erfordert Zusammenarbeit aller Abteilungen



## 1. Daten aus dem System abgreifen

Agenten sammeln Daten in sämtlichen Systemebenen. Auf der Anwendungsebene werden häufig technologiespezifische Agenten eingesetzt.

	Business	Benutzer	Anwendung	Middleware	Betriebssystem	Hardware
Umsatz	Umsatz, Conversion Rate, Bounce Rate					
Benutzerinteraktionen	Benutzerinteraktionen, Verweildauer, Ladezeiten, Fehler, Anzahl Ressourcen auf HTML-Seiten					
Komponenteninteraktion			Komponenteninteraktion, Methodenantwortzeiten, technische Fehlerzustände, Traces			
Statistiken zu Warteschlangen				Statistiken zu Warteschlangen, Pooling und Garbage Collection		
Statistiken zu File Handles, Threads, Virtualisierung					Statistiken zu File Handles, Threads, Virtualisierung	
Auslastung von CPU, Memory, I/O etc.						Auslastung von CPU, Memory, I/O etc.

**Aktiv**  
Stimuliert das System durch eigene Anfragen, Z. B. stichprobenartige Ausführung in der simulierten Benutzeranfragen

**Passiv**  
Liest und interpretiert Systemverhalten realer Benutzung, Z. B. Einfügen von Messlogik in den Anwendungscode, Auswertung von Netzwerkverkehr, Ressourcenauslastung oder Log-Dateien

Für die unteren Systemebenen existiert eine Vielzahl standardisierter Schnittstellen, > Z. B. JMX, Nagios

## 4. Interpretieren und Nutzen der Informationen

Aus den Informationen lassen sich manuell oder automatisiert Schlüsse ziehen und Aktionen ableiten.

**Performance-Probleme erkennen.** Typische Probleme sind erhöhte Antwortzeiten oder Ressourcenauslastungen. Erkant werden sie z. B. durch Vergleich mit Schwellwerten oder „Normalverhalten“ (Baselines). Herausforderung: Falsch-/Negativ/Positiv-Rate

**Ursachen typischer Performance-Probleme erkennen.** Z. B. N+1-Problem, zu viele Remote-Calls, schlechte Datenbankabfragen, diese (z. T. wiederkehrenden) Architektur- und Implementierungsfehler (inkl. Anti-Patterns) lassen sich insbesondere auf Basis aufzeichneter Trace-Informationen erkennen.

**Automatisierte Aktionen.** Insbesondere Cloud-Infrastrukturen bieten Dienste zur automatisierten Skalierung auf Basis der Monitoringinformationen an.



## APM-Werkzeuge

Übliche Funktionen von APM-Werkzeugen (Gartner-Dimensionen):

- > End user experience
- > Component deep dive
- > Architecture discovery
- > Resource Monitoring
- > Transaction profiling
- > Analytics

Kommerzielle Werkzeuge bieten umfassende Lösungen.

- > Z. B. AppDynamics, BMC Software, CA APM, Dynatrace, HP Enterprise, IBM, New Relic, Riverbed Technology

Open-Source-Werkzeuge sind i. d. R. weniger mächtig, können jedoch Alternativen darstellen.

- > Z. B. ICINGA, InspecT, Kieker, PinPoint, Zipkin, ZMon

## APM-Wunschliste

Das Themenfeld APM hat sich in den letzten Jahren stark entwickelt. Es gibt aber noch genug Potenzial. Z. B.

**Interoperabilität.** Einige APM-Werkzeuge bieten einen Daten-Export. Eine erhöhte Interoperabilität (einheitliche Schnittstellen, Formate) ist wünschenswert.  
> Z. B. OpenTracing, OPEN.xtrace

**Automatisierung.** Manuelle APM-Aktivitäten (z. B. Konfiguration, Problemdiagnose) sind aufwändig und fehleranfällig. Automatisierung, z. B. basierend auf Best Practices ist wünschenswert.

**APM ist kein reines Technikthema.** APM wird immer noch stark mit Technik assoziiert, ist aber längst auch ein Business-thema. Bis 2020 wird der Anteil des IT-Betriebs an den Käufem von APM-Lösungen lt. Gartner nur noch 40 % betragen (aktuell 75 %).

**Technologietransfer.** Forschung und Industrie können stark voneinander profitieren.  
> Z. B. Austausch von Traces, Probleme und Algorithmen, Kombination mit modellbasierten Ansätzen über Plattformen wie die .SPEC Research Group.

## 2. Von Daten zu Informationen

Es hat sich bewährt, Zeitreihendaten für das Monitoring zu erheben und diese mit detaillierten Traces für die Problemanalyse zu unterbauen.

Beispiele in APM-Werkzeugen

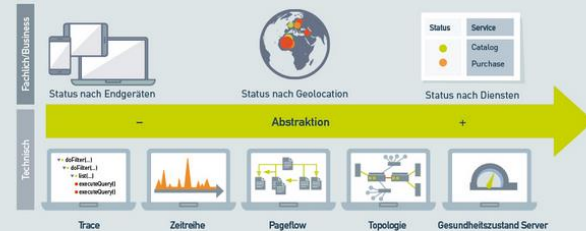
- > Solche Visualisierungs-Werkzeuge bieten in der Regel eine recht intuitive und schnelle Möglichkeit, Graphen, Metriken, Tabellen, etc. zu konfigurieren
- > Hierzu werden Abfragen an die darunterliegende Zeitreihen-Datenbank angegeben, die spezifizieren, welche Zeitreihen visualisiert werden sollen
- > Weitere Darstellungseigenschaften können dann konfiguriert werden (Legende, Axen, etc.)
- > Auf diese Weise können einzelne Panels angegeben werden



Trace	Anzahl
af3f8d1...	1793
af3f8d1...	1793
af3f8d1...	1432
af3f8d1...	262
af3f8d1...	143
af3f8d1...	17

## 3. Visualisierung in navigierbaren Sichten

Aufgrund der hohen Menge müssen Informationen aufbereitet werden. Es hat sich bewährt, verschiedene Sichten auf die Daten anzubieten. Die Sichten sind navigierbar, um zu abstrakteren oder detaillierteren Sichten zur Beantwortung der jeweiligen Fragestellung zu gelangen.



André van Hoorn und Stefan Siegl.

Application Performance Management (APM). Continuous Monitoring of Application Performance. (in German)

Order for free: <http://www.sigs-datacom.de/wissen/fachposter.html>



# Application Performance

Kontinuierliche Überwachung von An...

## Einfluss von Performance auf Erfolg

Anwendungsperformance hat direkte Auswirkungen auf den Unternehmenserfolg.

> Z. B. Wahrnehmung, Vertrauen, Umsatz

2 Sek. 50% der Online-Kunden verlassen eine Seite nach Ladezeiten > 2 Sekunden.

60% minds Nur 40% der Online-Kunden kommen nach einem Performance-Problem zurück.

## Konsequenzen schlechter Performance



## Application Performance Management

Durch APM-Werkzeuge und -Prozesse kann Anwendungsperformance kontinuierlich überwacht und sichergestellt werden

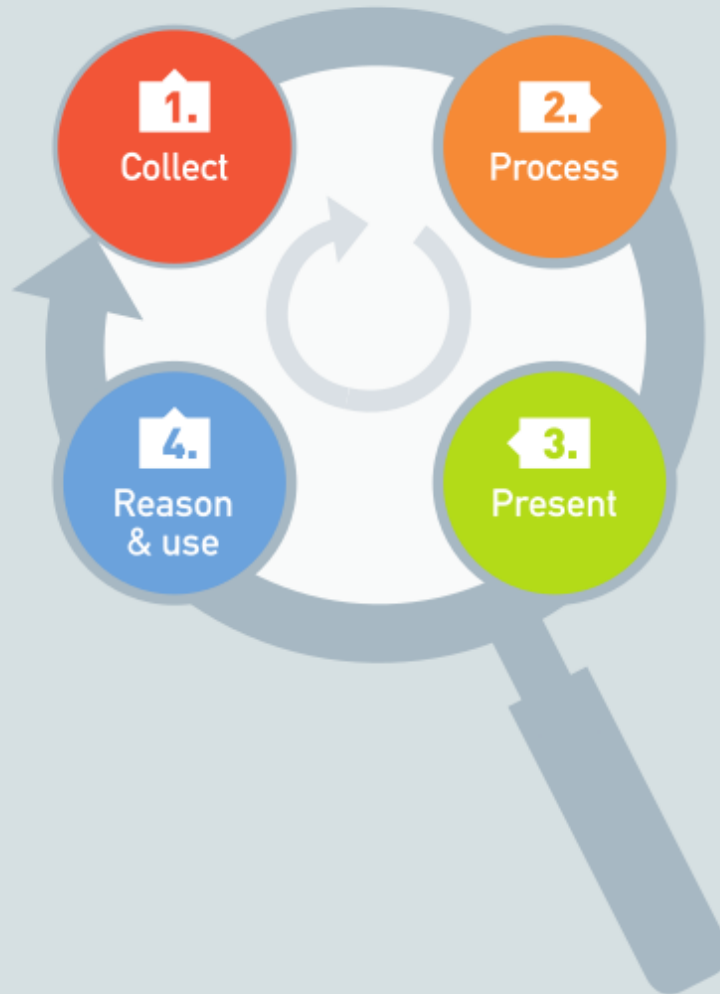
...wichtig im gesamten Software-Lebenszyklus



... relevant für alle Systemebenen



... erfordert Zusammenarbeit aller Abteilungen



by Inhäliche Entwicklung: Dr.-Ing. André van Hoorn (Prof.-Vert.) und Stefan Siegl

it detaillierten Traces für die

und schnelle Möglichkeit, Graphen, Metriken, Tabellen, etc. zu konfigurieren gegeben, die spezifizieren, welche Zeitreihen visualisiert werden sollen (siehe Axon etc.)

Trace	Anzahlwert
af3f3d1e...	[179] ms
searchTitleAndDescription...	[178] ms
searchTitleAndDescriptionWithOut...	[132] ms
...	[82] ms
searchQuery@ SQL Server MW_0...	[143] ms
searchQuery@ SQL Server MW_0...	[17] ms

ten

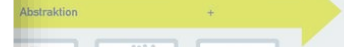
et werden. Es hat sich bewährt, verschiedene ar, um zu abstrakteren oder detaillierteren elangen.



Status	Service
●	Catalog
●	Purchase

s nach Geolocation

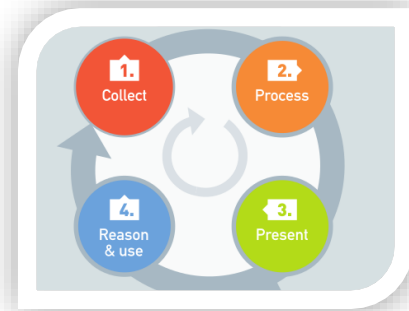
Status nach Diensten



ma. mit Technik ein Business-... des IT-B...-Lösungen ragen (aktuell

Technologietransfer. Forschung und Industrie können stark voneinander profitieren. > Z. B. Austausch von Traces, Probleme und Algorithmen, Kombination mit modellbasierten Ansätzen über Plattformen wie die „SPEC Research Group“.

# 1. Collecting Data from All System Levels



- Agents collect data from all system levels
- On application level the agents are often technology-dependent

Where?

What?

How?

		Active	Passive
Business	Sales data, conversion and bounce rate		
User	User interactions: length of stay, load times, errors; number of resources on HTML pages	Stimulation of the system by periodic requests.	Collection of runtime data from real system usage.
Application	Component interactions, method response times, trace data	E.g., synthetic user transactions	E.g., injection of code, analysis of network traffic, resource utilization, or log files
Middleware	Queuing statistics, pooling, garbage collection		
Operating System	File handling statistics, virtualization, thread statistics		
Hardware	CPU load, memory consumption, I/O statistics	Some technologies on lower levels provide standard interfaces for data collection, e.g., Nagios, JMX	

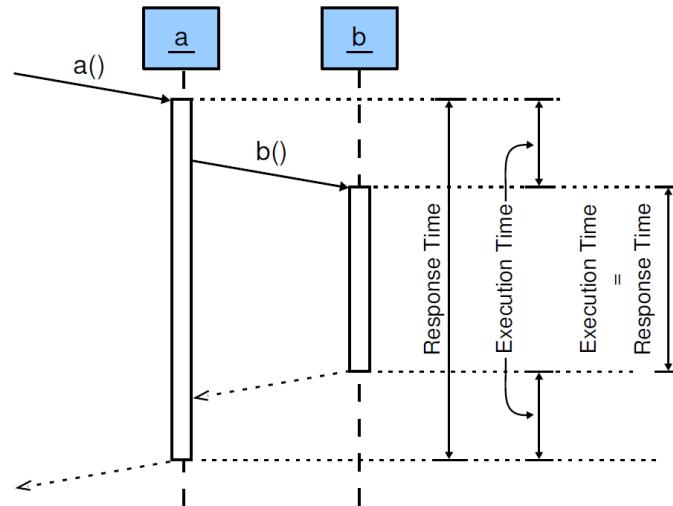
# Trace-based Metrics (Selection)

What?

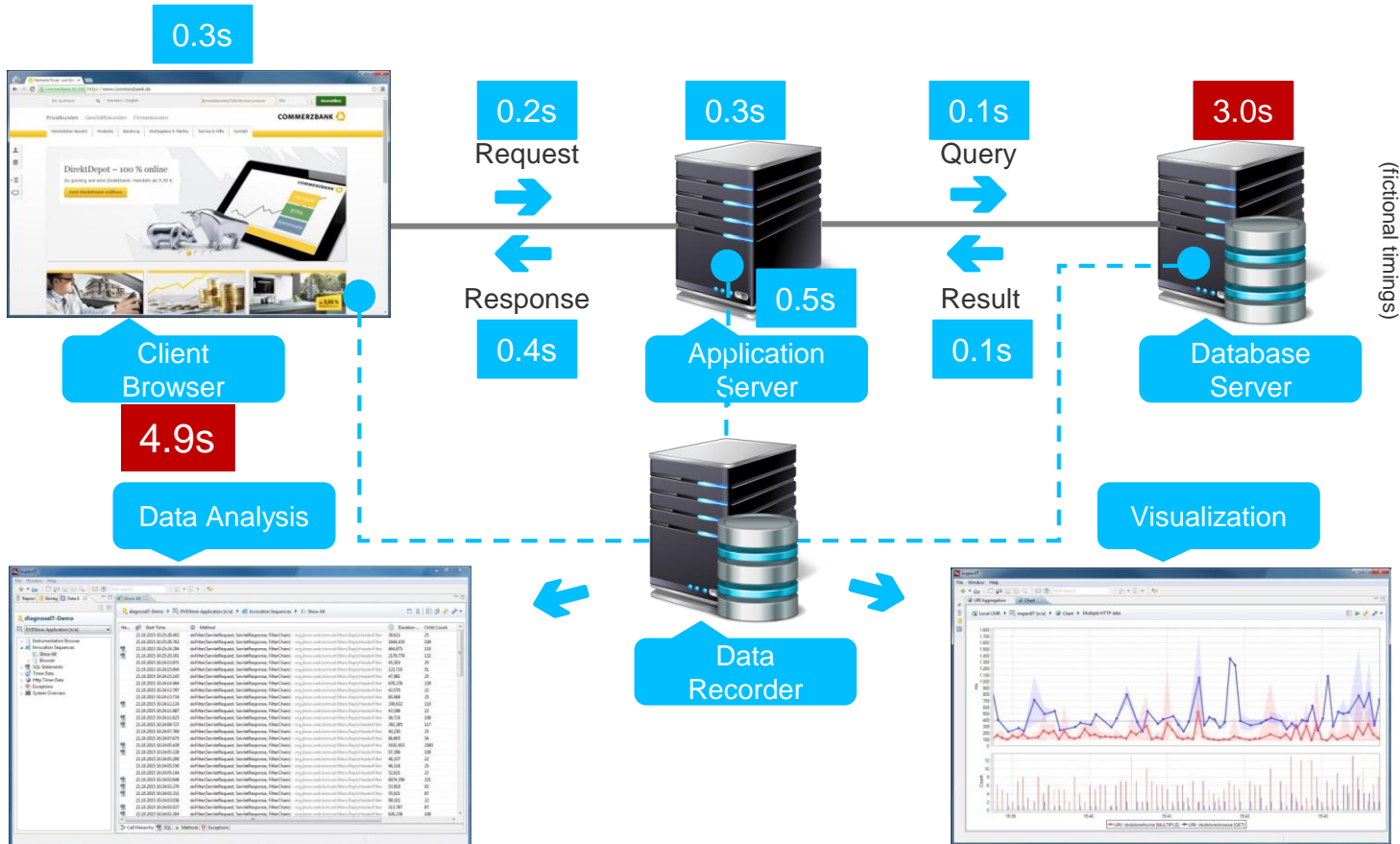
Application

Component interactions, method response times, trace data

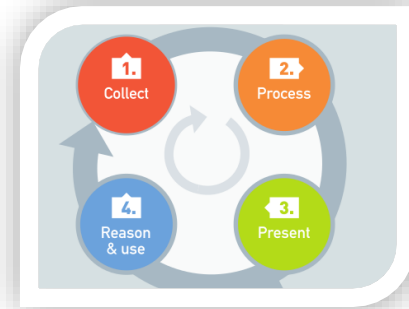
Metric
Response Time
CPU Time
Method Name
Return Type
Logging Level
SQL Statement
Error Message
...



# Monitoring (Measurement-based Performance Evaluation)

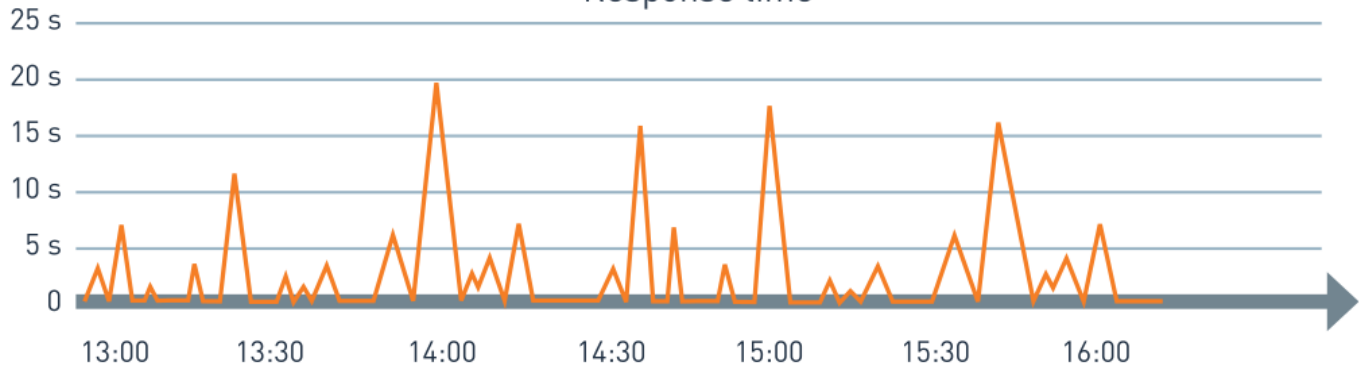


## 2. Reconstructing Information from Data

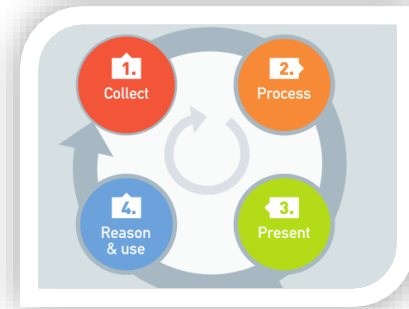


- Data is collected from the system...
- represented as **time series**...

Response time



## 2. Reconstructing Information from Data

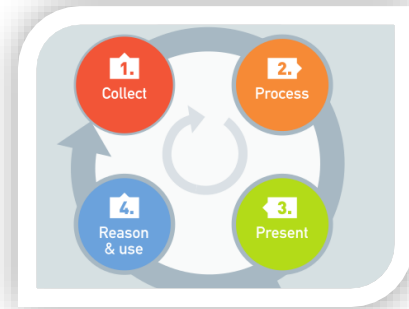


- Data is collected from the system...
- represented as **time series**...
- ... and as **detailed execution traces**, and used to support problem analysis

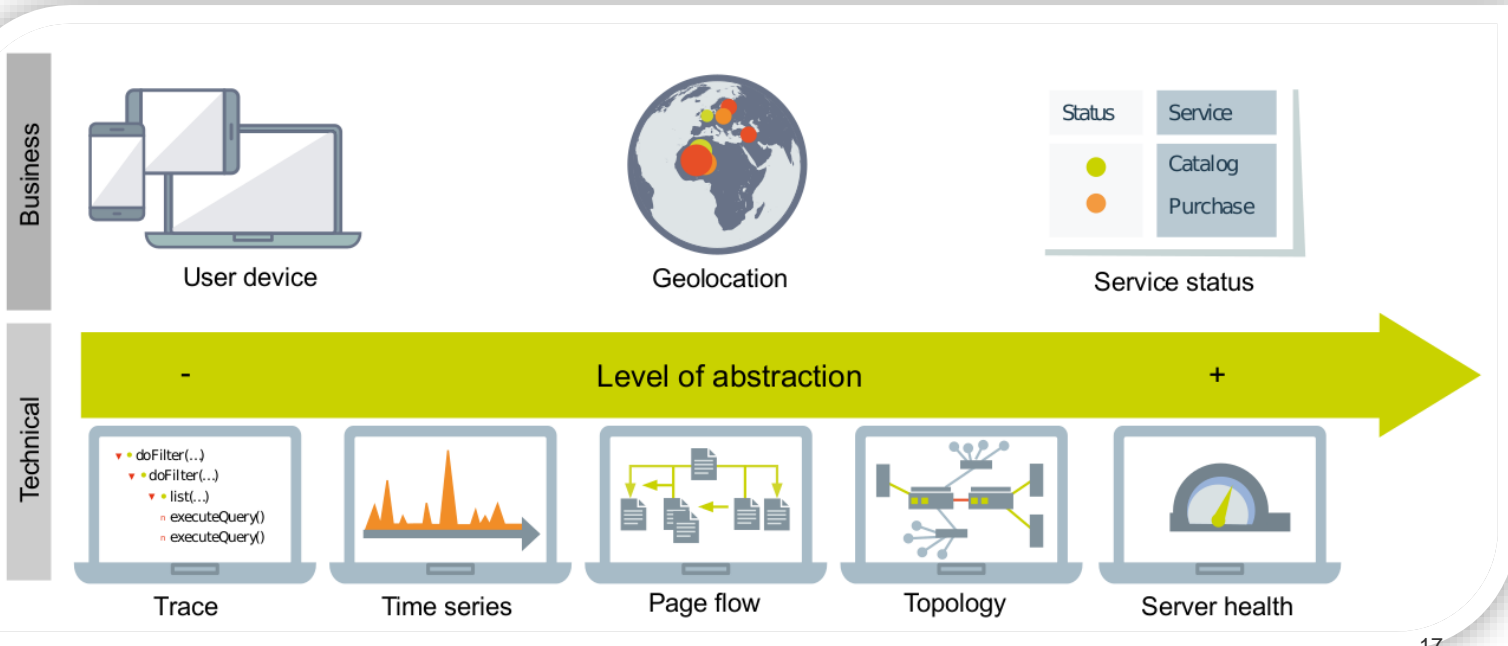
Trace	Response time
▼ ● doFilter(...)	[1793 ms]
▼ ● searchTitleAndDescription(...)	[1730 ms]
▼ ● searchTitleAndDescriptionWithOneWord(...)	[1632 ms]
▼ ● list(...)	[352 ms]
■ executeQuery() {SQL: Select PROD_ID ...}	[143 ms]
■ executeQuery() {SQL: Select INV_ID ...}	[12 ms]
▼ ● ...	...
...	...
...	...



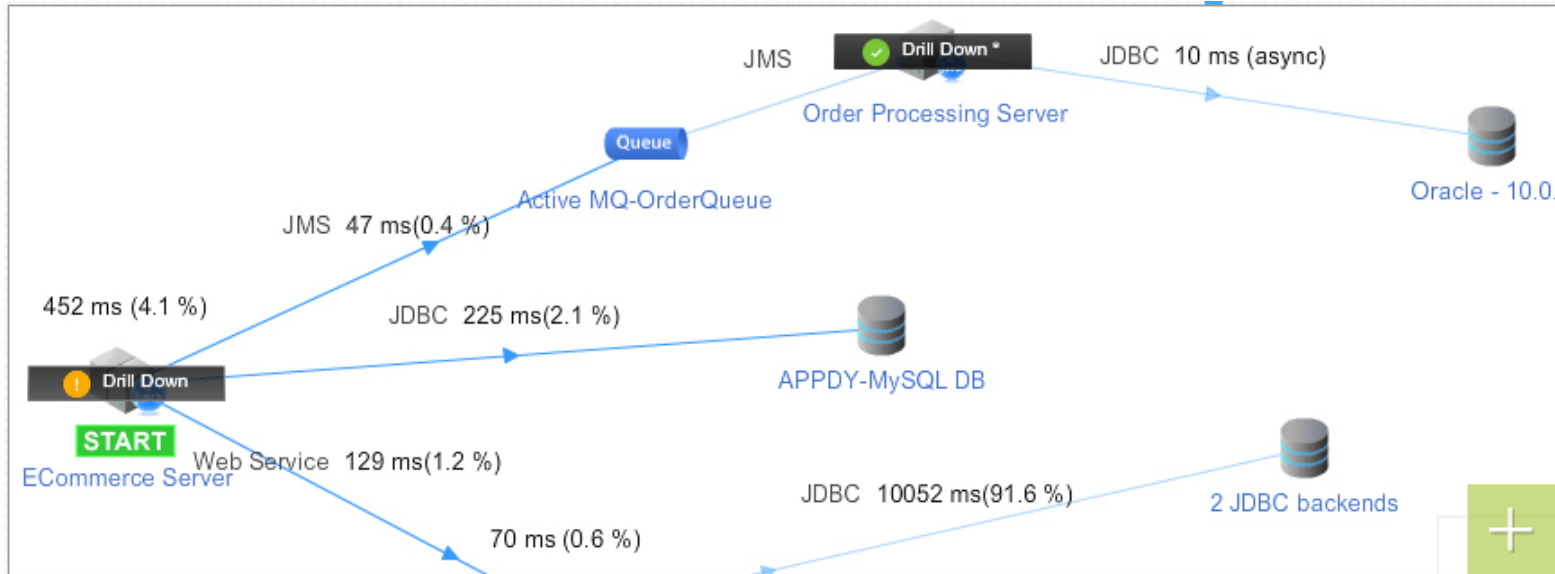
### 3. Visualization Through Navigable Views



- High quantity of information has to be pre-processed
- It has proven useful to use **different views** to show the data
- **Views are navigable** and can be categorized by both scope and detail level

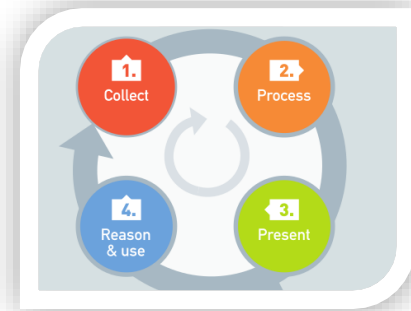


# Example: Application Topology Discovery and Visualization



© AppDynamics

## 4. Interpreting and Using the Information



Manual or automated conclusions and actions can be derived from the information, e.g.,

- **Problem detection and alerting**

- E.g., increased response times and resource utilization
- Detection, for instance, based on thresholds and baselines

- **Problem diagnosis and root cause isolation**

- E.g., N+1 problem, too many remote calls, poor DB queries
- Detection based on monitoring information

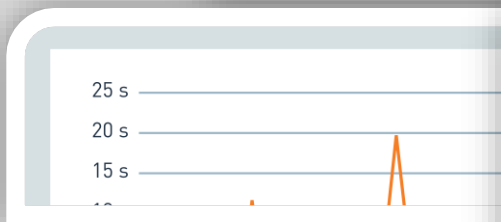
- **System refactoring and adaptation**

- E.g., auto-scaling in cloud-based architectures



# State of the Art of Visualization in APM Tools

## Part 2/2: Examples Visualizations in APM Tools



```
Trace
▼ doFilter(...)
▼ searchTitleAndDescription(...)
▼ searchTitleAndDescriptionWithOneWord(...)
▼ list(...)
  ■ executeQuery() {SQL: Select PROD_ID ...}
  ■ executeQuery() {SQL: Select INV_ID ...}
```

**André van Hoorn**  
**Dušan Okanović**

# Commercial APM Tools

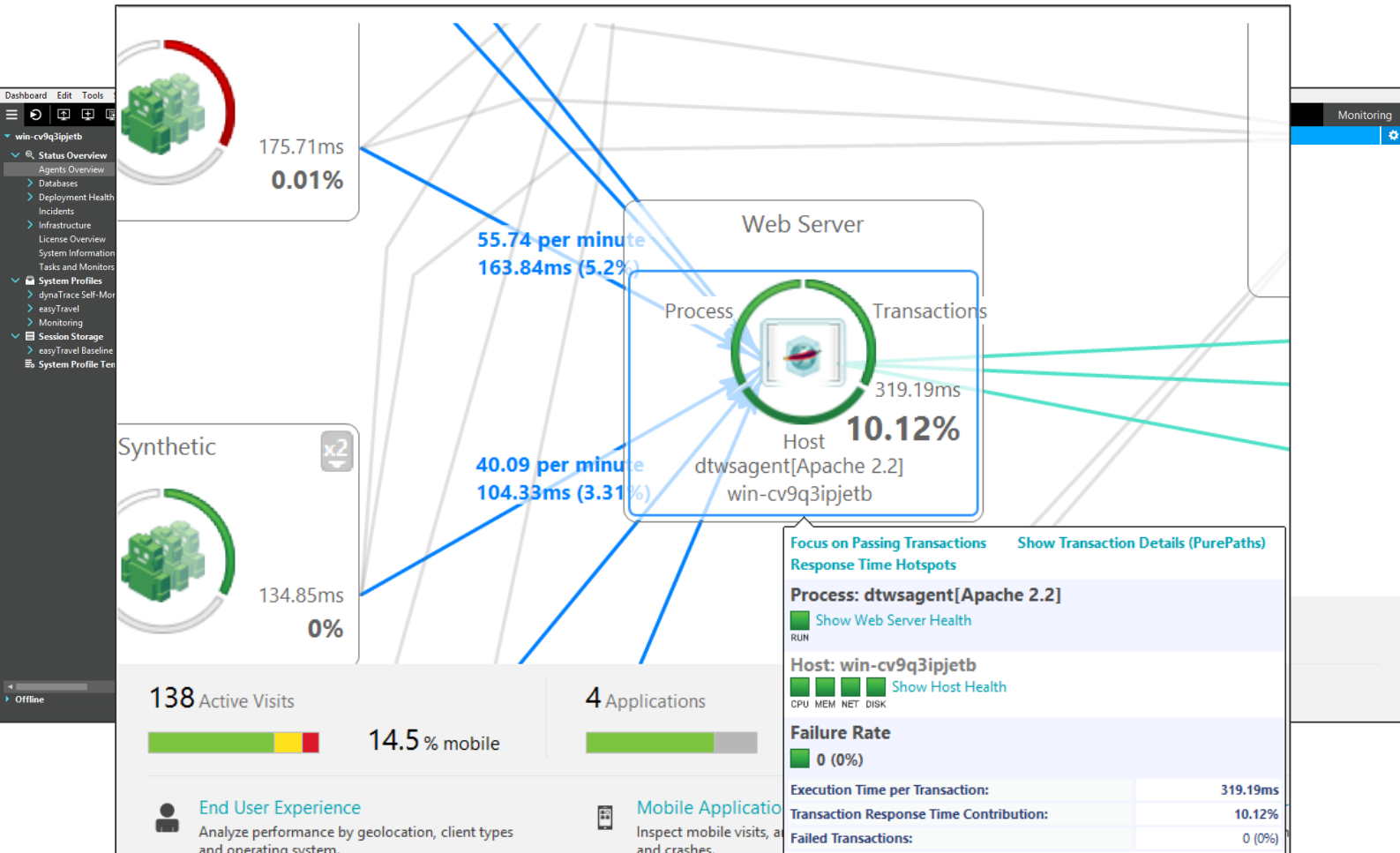
# Magic Quadrant



Source: Gartner (March 2018)



# Application Overview – Dynatrace



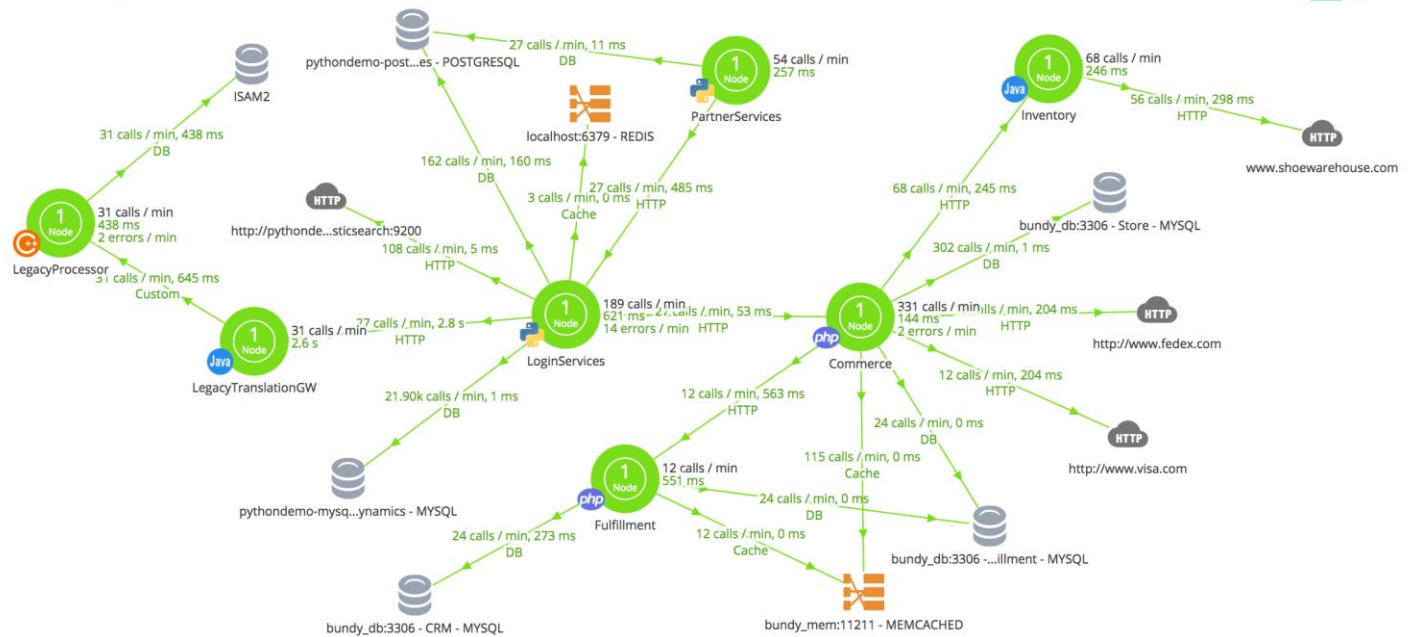
# Application Overview – AppDynamics

Online\_Retail

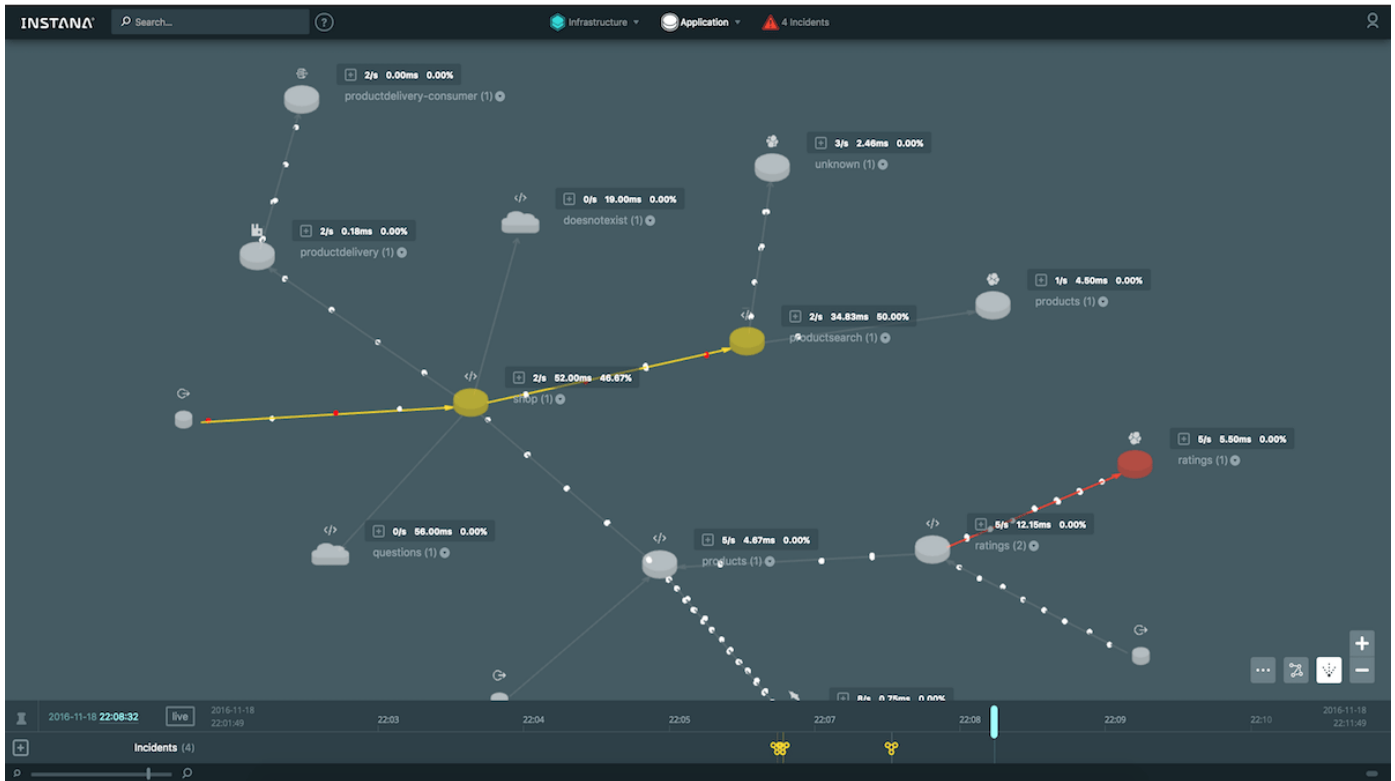
🔄 Daily Trend - Last 30 days 📅 last 15 minutes ?

Dashboard Events Top Business Transactions Transaction Snapshots Transaction Score Actions

Application Flow Map






# Application Overview – Instana



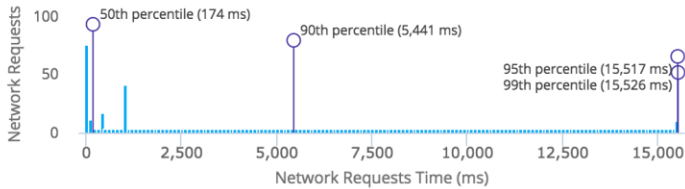
# End User Monitoring – AppDynamics

Ecommerce-iOS

  last 15 minutes  

**Overview**    Geo Dashboard    Usage Stats

Network Requests Time Distribution



Total Crashes

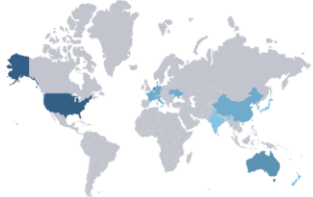
2  
Total Crashes

Crash Rates by App Version

2.0	5.9%
2.1	5.9%
1.0	-

App Opens by Country >

×



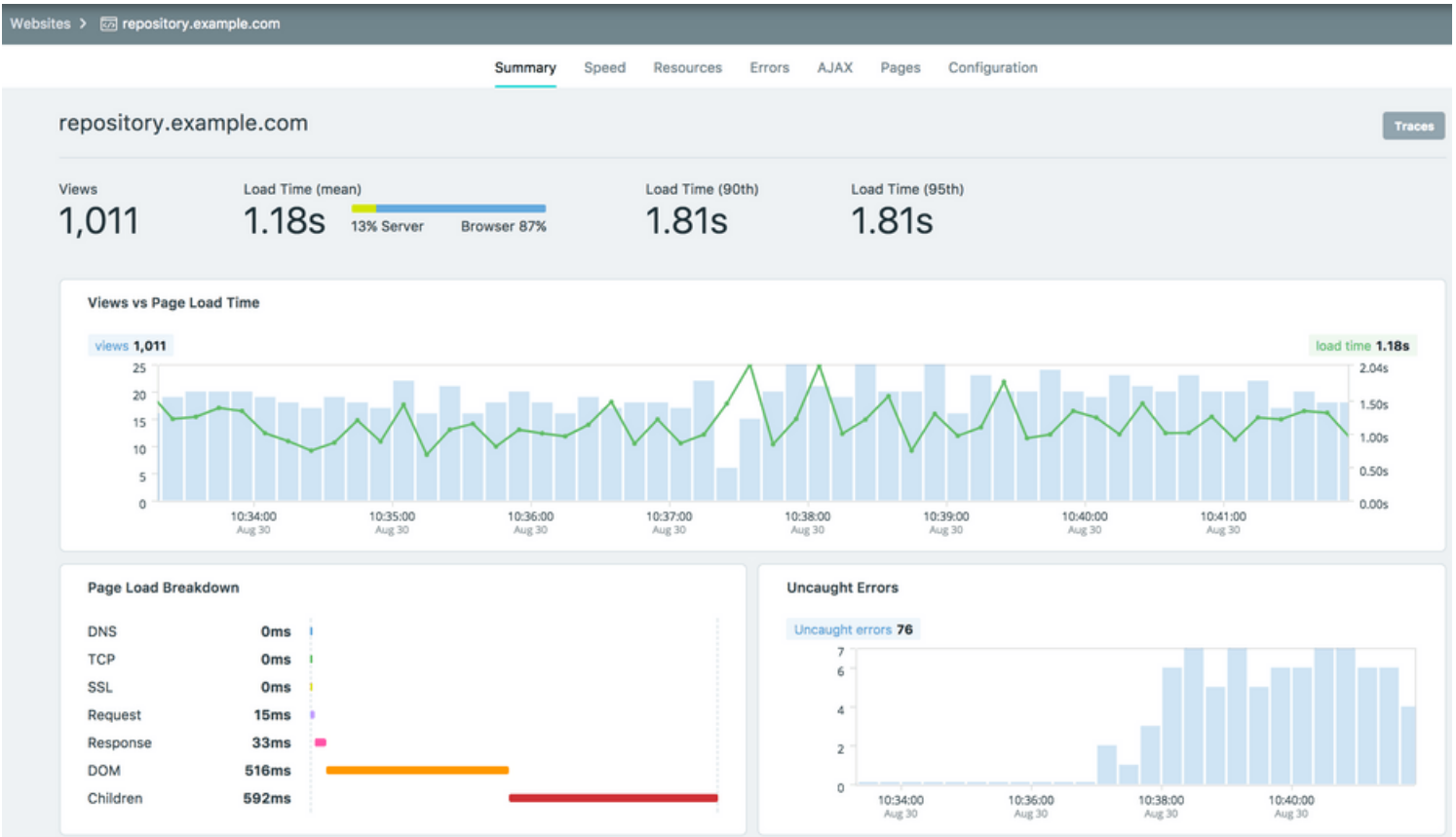
App Opens Per Minute



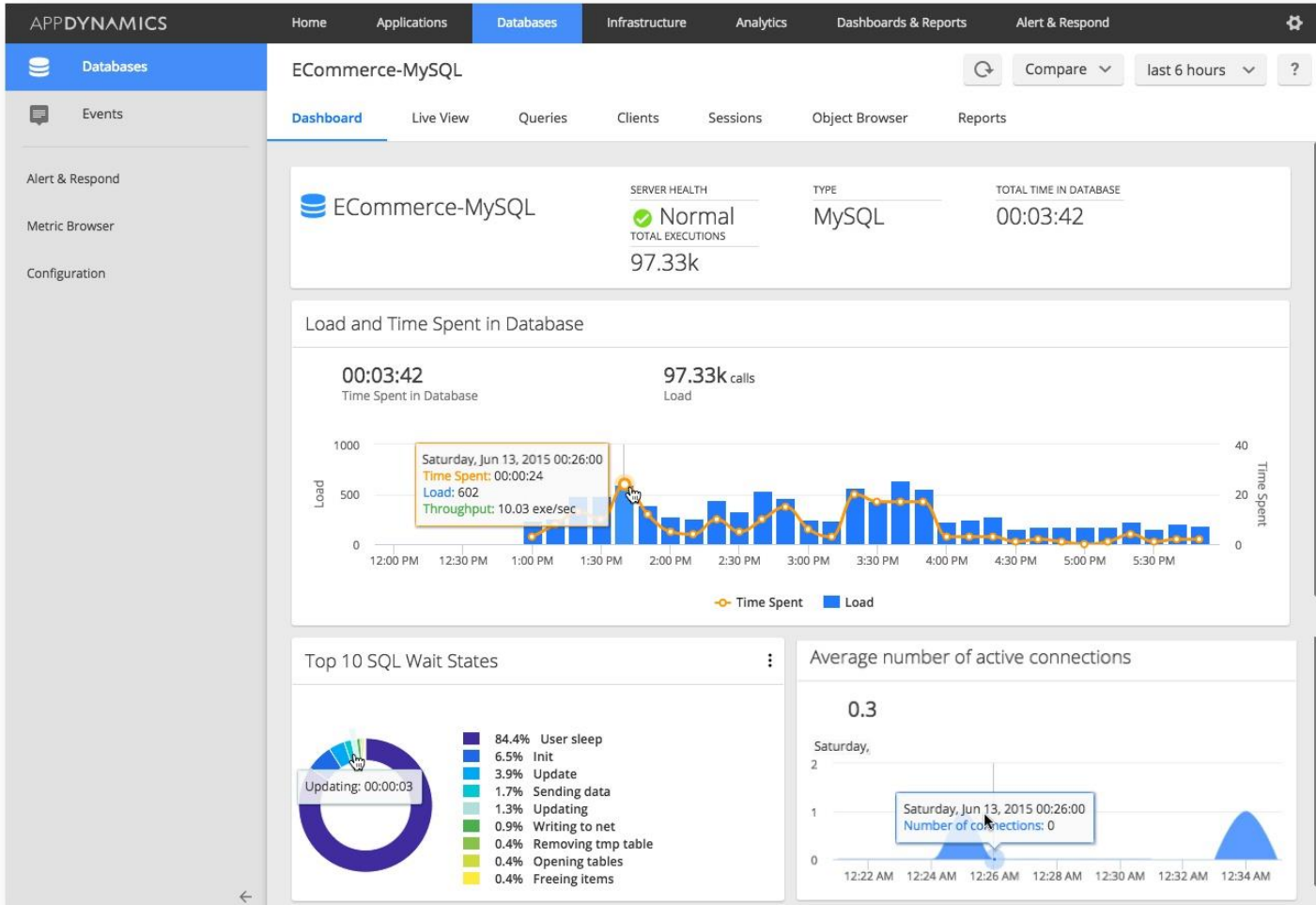
Top 5 Unique Crashes

SIGSEGV in promovie	1 Crashes	1 Impacted Users
SIGSEGV in add	1 Crashes	1 Impacted Users

# End User Monitoring– Instana

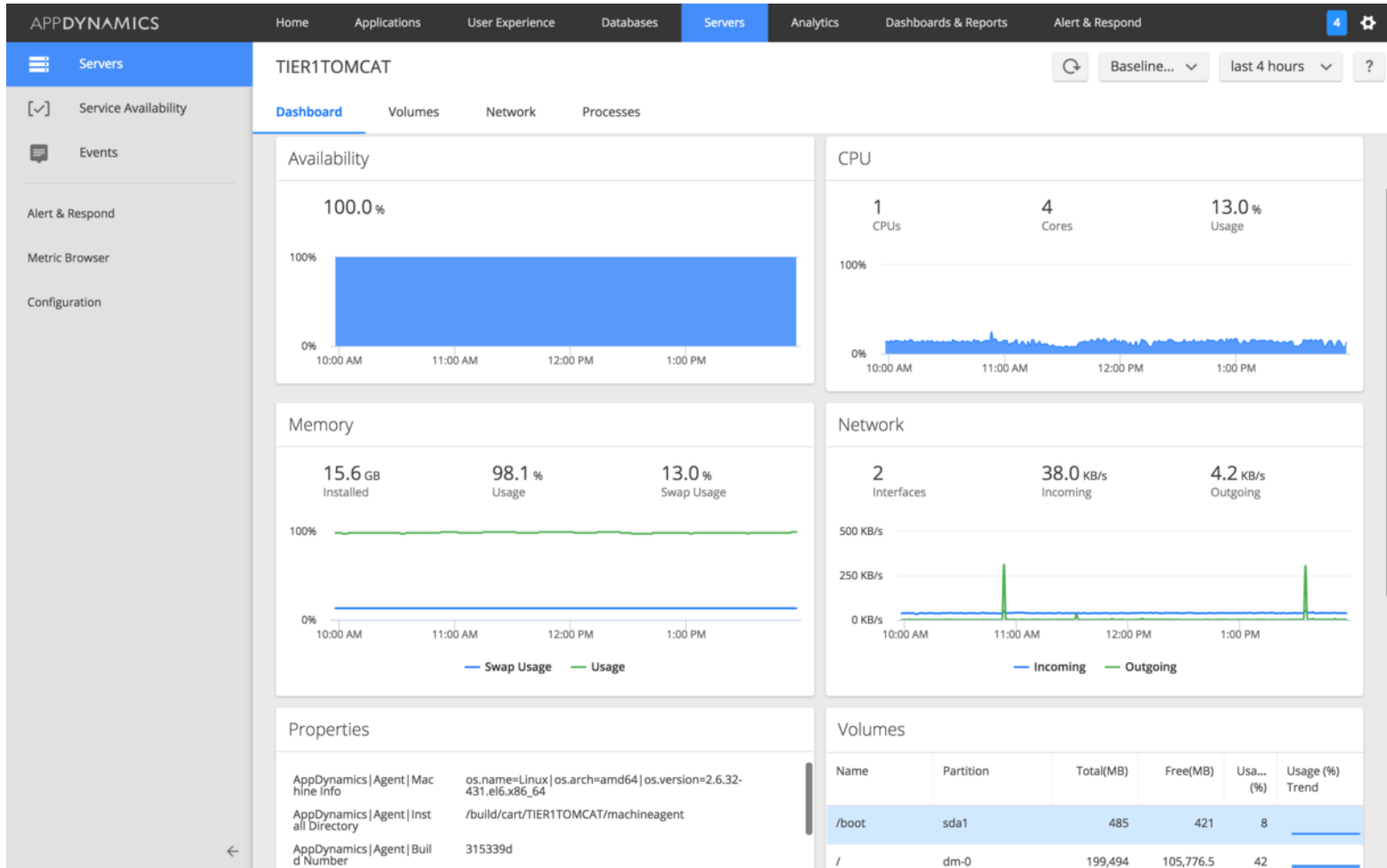


# Database Monitoring – AppDynamics





# Server Monitoring – AppDynamics

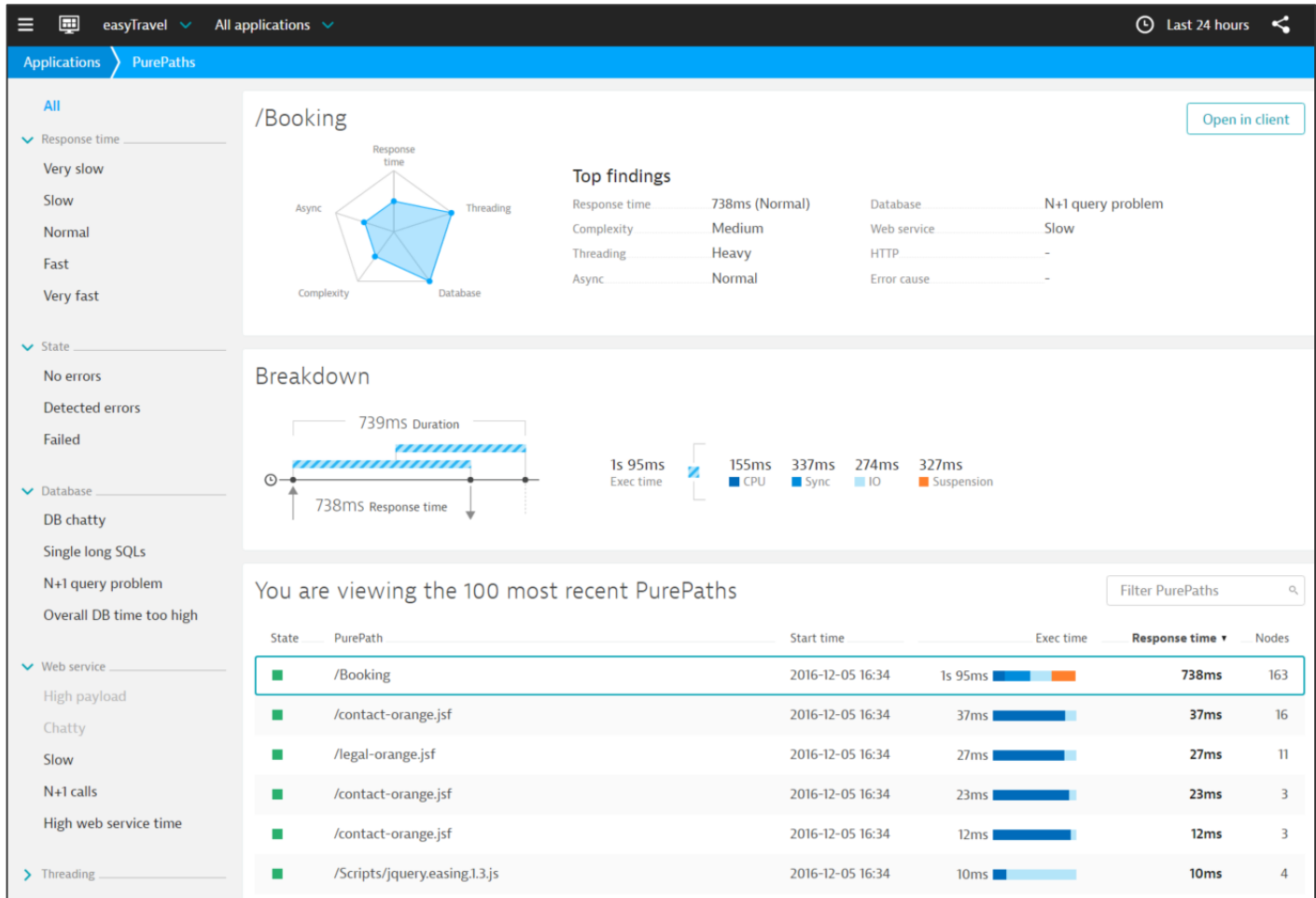


Tutorial: State of the Art of Visualization in APM Tools

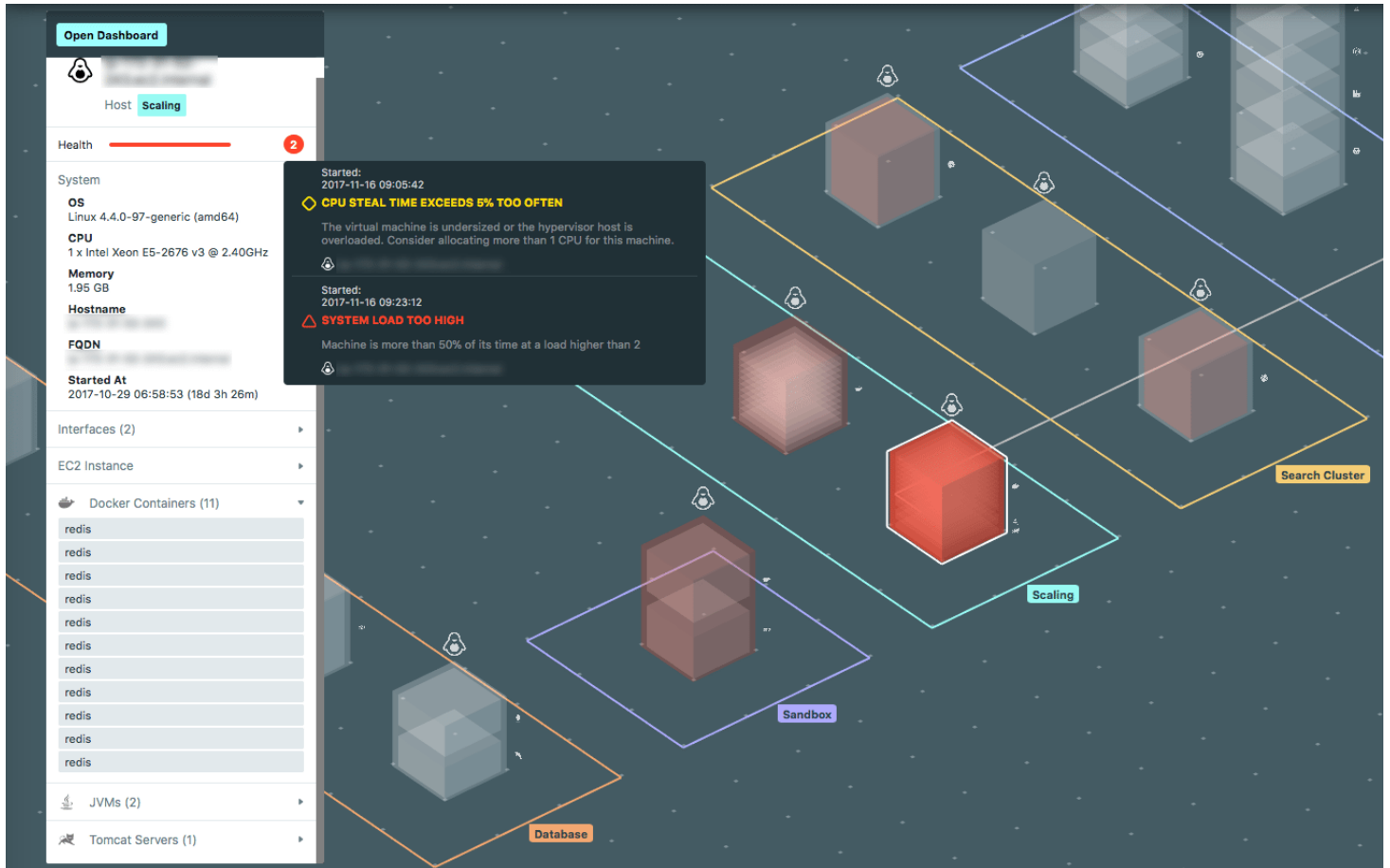
# Server Monitoring – Instana

The screenshot displays the Instana monitoring dashboard. At the top, the navigation bar includes 'INSTANA', 'Infrastructure', 'Application', and 'Incidents'. The main content area features a 3D visualization of a server stack with a tooltip that reads 'Spring Boot App : Instana Demo - Shop Service 0.0.1'. On the left, a sidebar provides application details for 'Instana Demo - Shop Service 0.0.1', including its health status (UP), name, version (0.0.1), Springboot version (1.4.1.RELEASE), status (UP), port (8080), and HTTP sessions max (-1). Below the sidebar, a 'Live' status indicator and a timeline are visible, showing 'Incidents (2)', 'Issues (31)', and 'Changes (273)'. The bottom of the interface shows a 'Set Time' button, a 'Live' indicator, and a timeline with 'Incidents (0)'.

# Problem Identification – Dynatrace



# Problem Identification - Instana



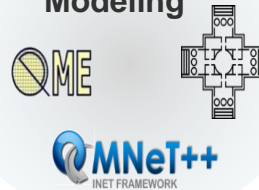
# Open Source APM tools

# Open Source APM tools

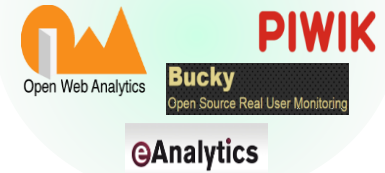
## Monitoring & Application Deep Dive



## Performance Modeling



## Real User Monitoring



## Low-Level Performance Profiling



## System & Resources Monitoring

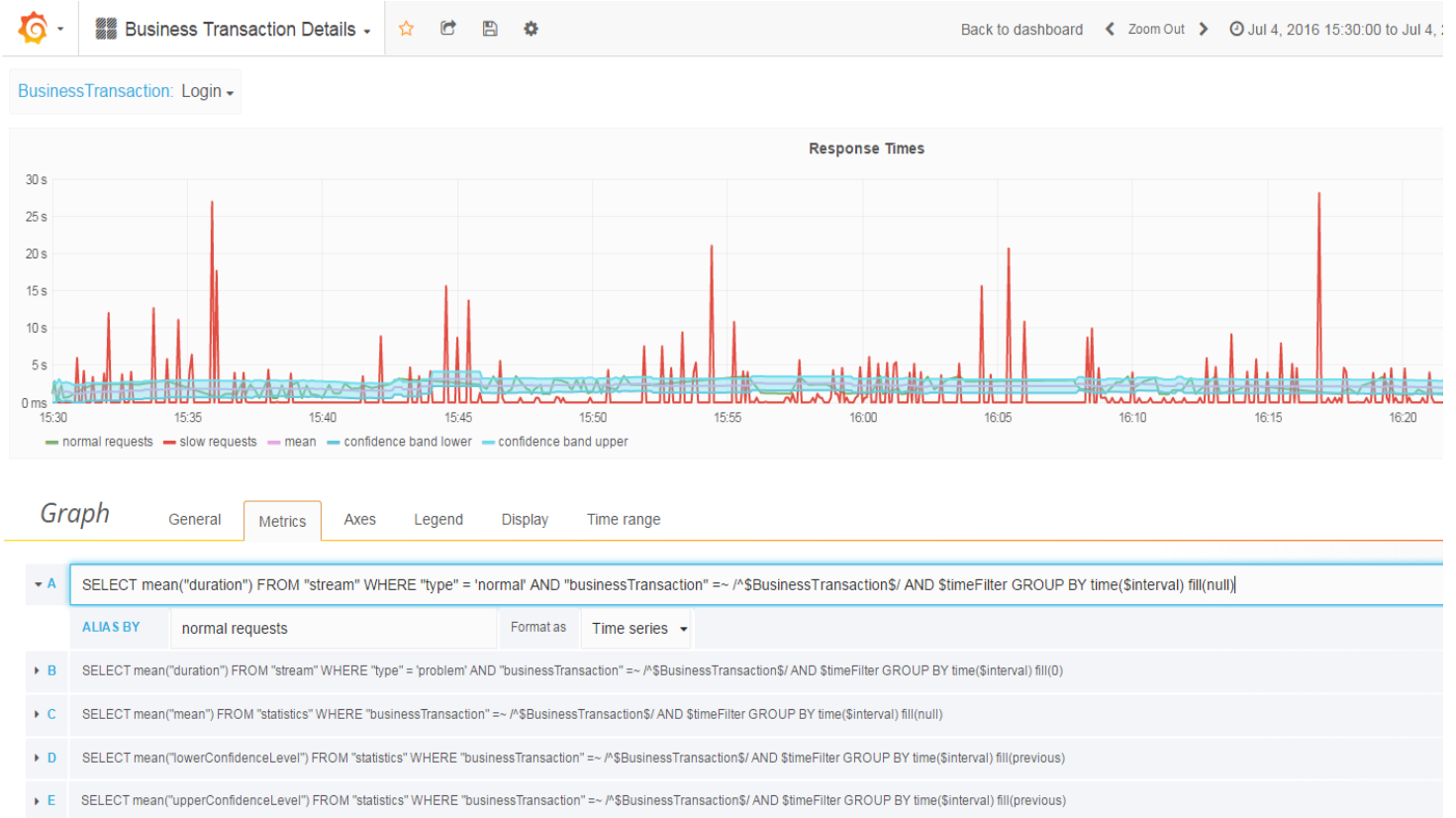


## Web Performance Analysis

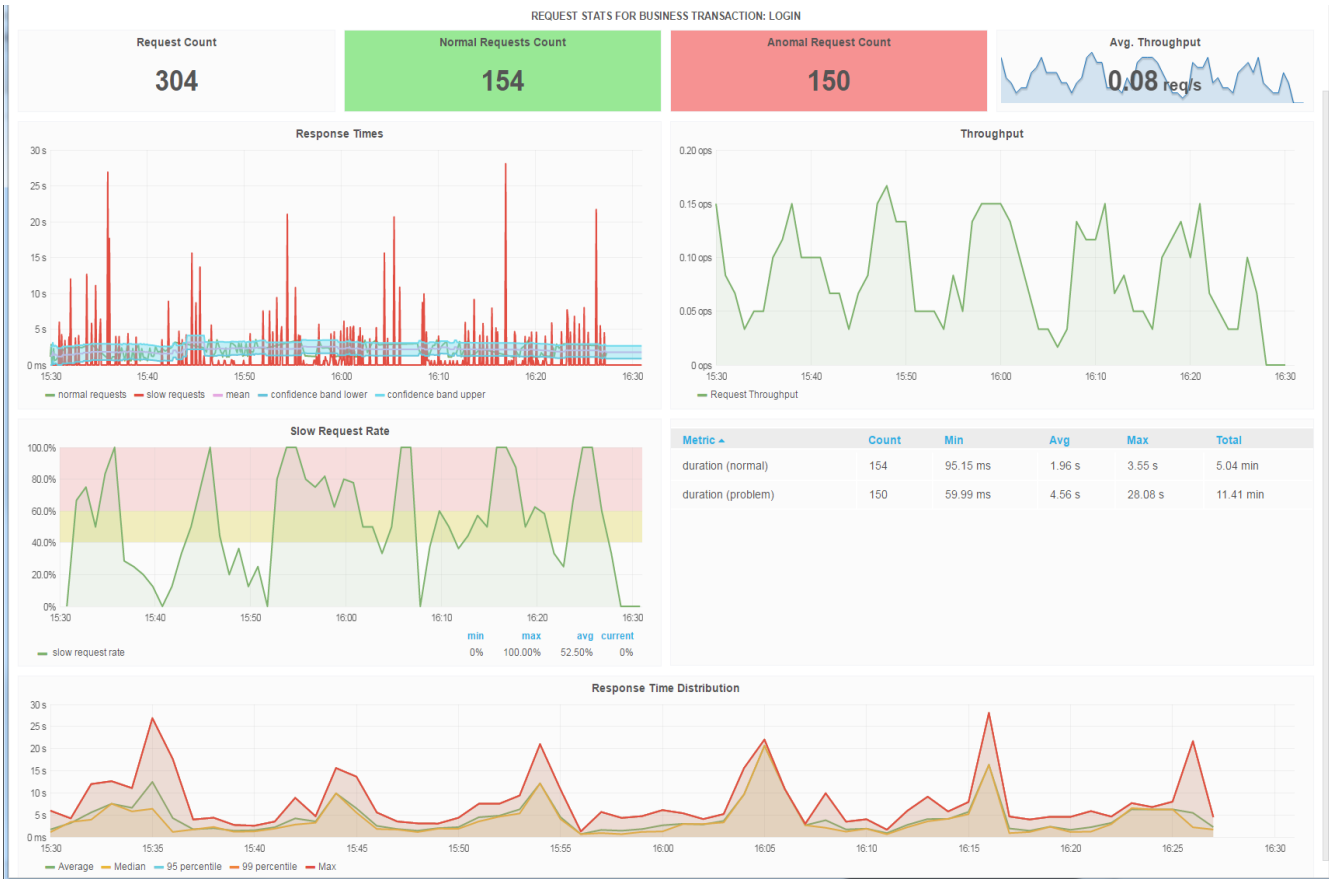




# Dashboards – Grafana

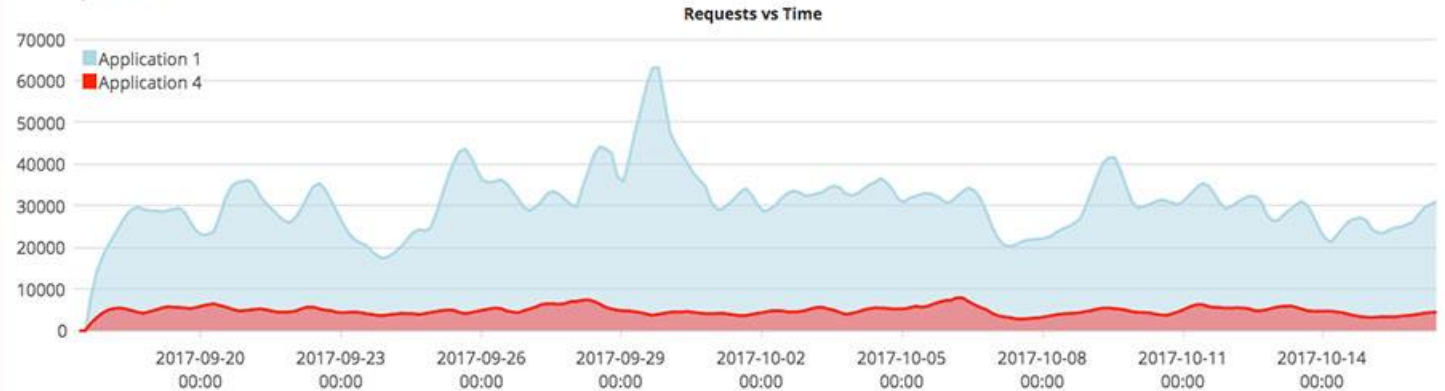


# Dashboards – Grafana

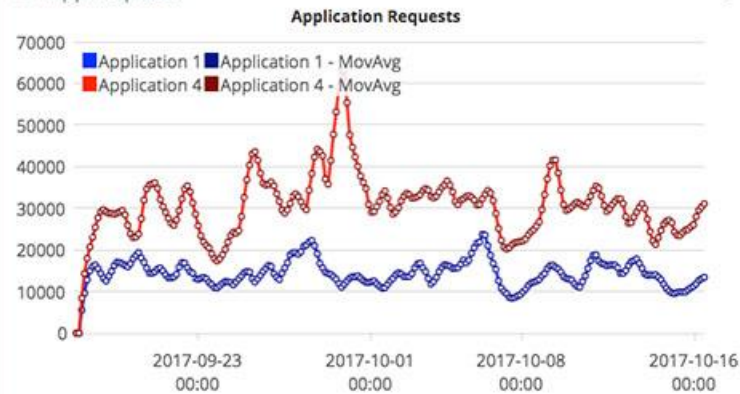


# Dashboards – Kibana

TS - Req v Time



TS - App Requests



TS - App4

TS - App1



# Dashboards – Kibana

Apache - Total Visitors

4,931,584

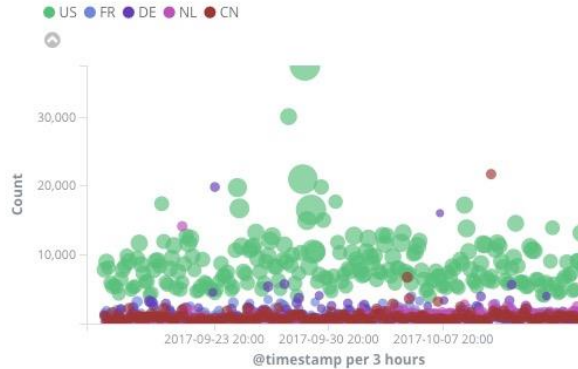
Apache - Unique Visitors ...

City	Unique Visitors
Beijing	562
Redmond	445
Ashburn	400
Chicago	373
Los Angeles	245
Seattle	233
San Jose	232
Singapore	208

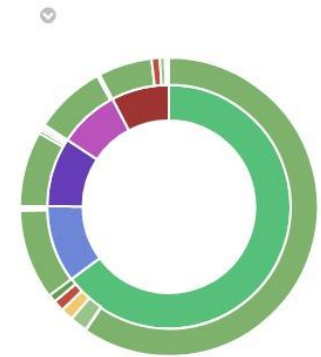
Apache - Unique Visitors

29,740

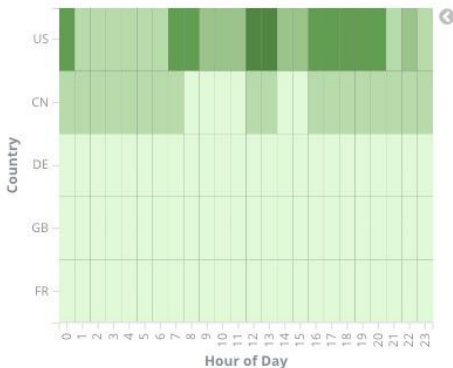
Apache - Bytes and Count



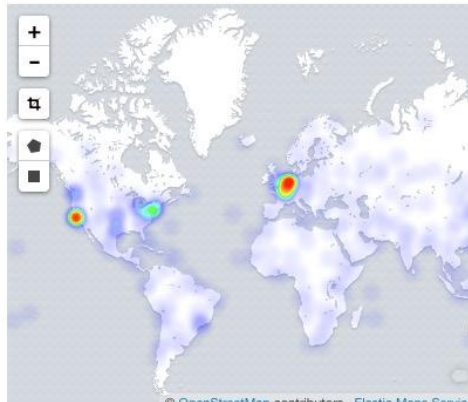
Apache - Country and Status



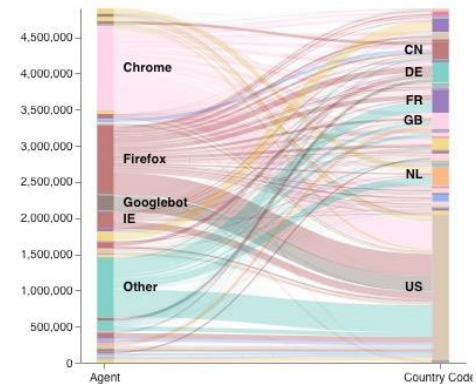
Apache - Country traffic by hour



Apache - Visitor Map (geocentroid)



Apache - Browser to Country (vega)



# Execution Traces – inspectIT

The screenshot shows the inspectIT application window. The title bar reads 'inspectIT'. The menu bar includes 'File', 'Window', and 'Help'. Below the menu bar is a toolbar with various icons and a 'Wiki search' field. The main window area displays a breadcrumb path: 'Local CMR > inspectIT [n/a] > Invocation Sequences > Show All'. A large blue rounded rectangle is overlaid on the center of the screen, containing the text 'OPEN.xtrace' and the URL 'https://github.com/spec-rgdevops/OPEN.xtrace'. In the background, a table of execution traces is visible, with columns for 'Start Delt...'. The table shows several rows of data, including method names like 'executeQuery()' and their corresponding start and end times.

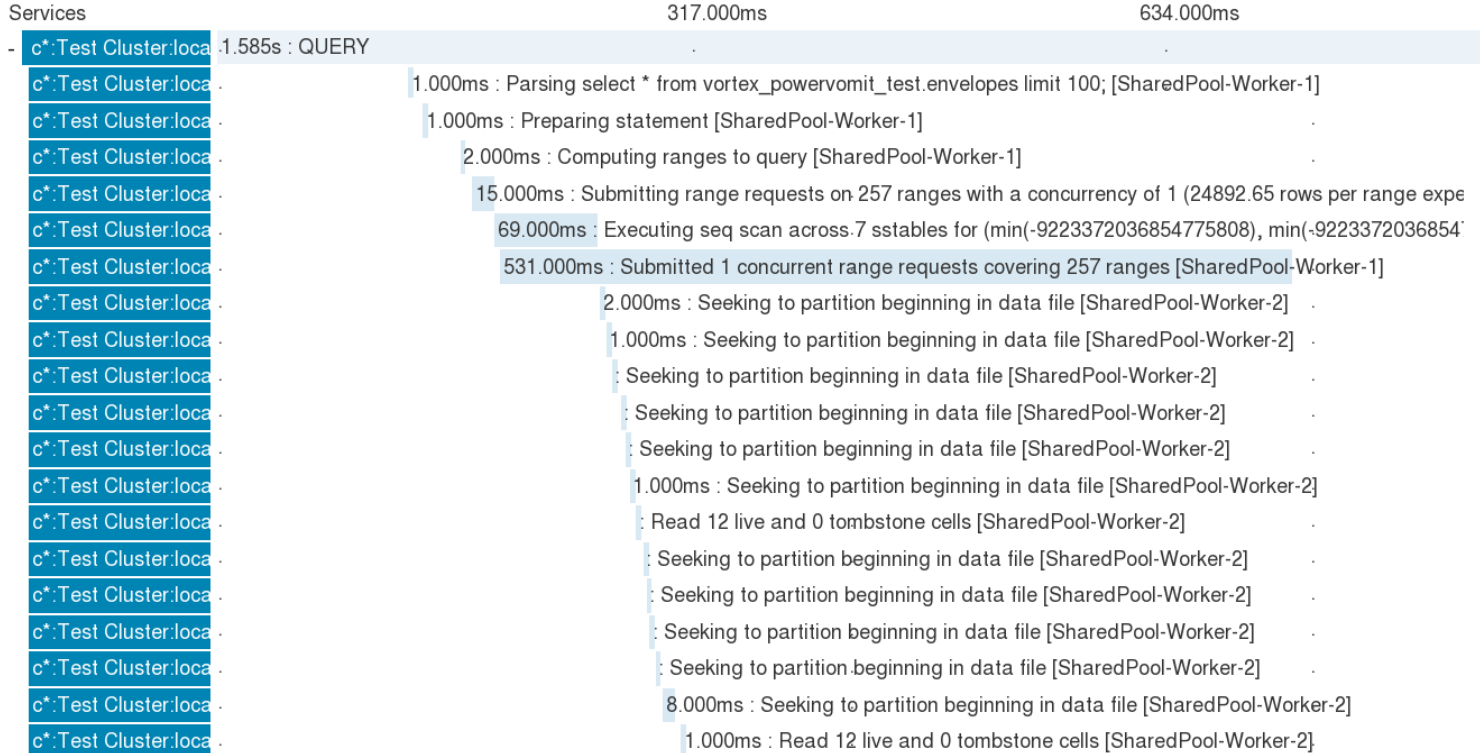
Method	Start Delt...	End Delt...	Category
executeQuery() - org.h2.jdbc.JdbcPreparedStatement	0,244	0,244	SE
executeQuery() - org.h2.jdbc.JdbcPreparedStatement	0,125	0,125	SE
executeQuery() - org.h2.jdbc.JdbcPreparedStatement	0,100	0,100	SE

# Tracing – Zipkin

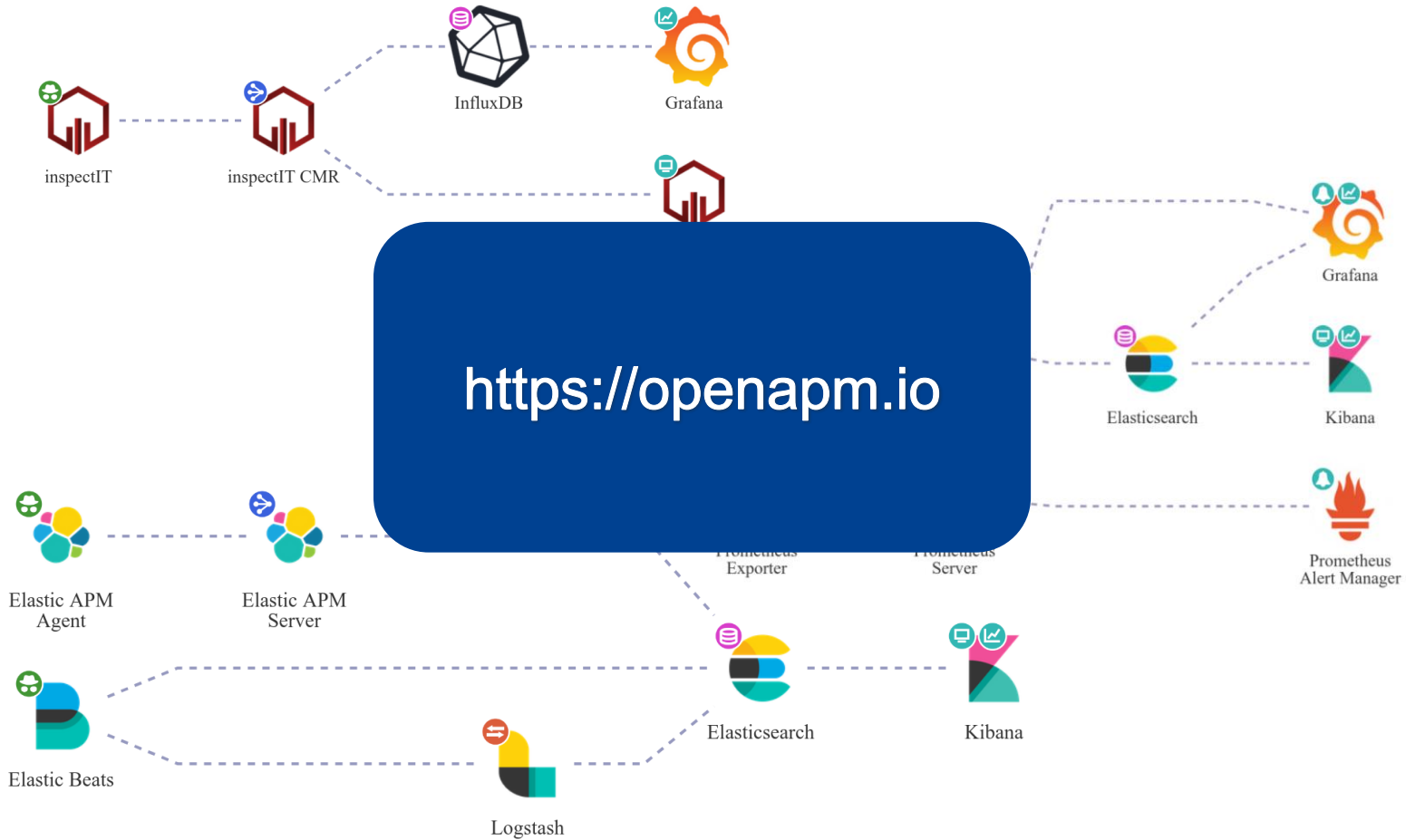
Zipkin Investigate system behavior

Find a trace

Aggregates

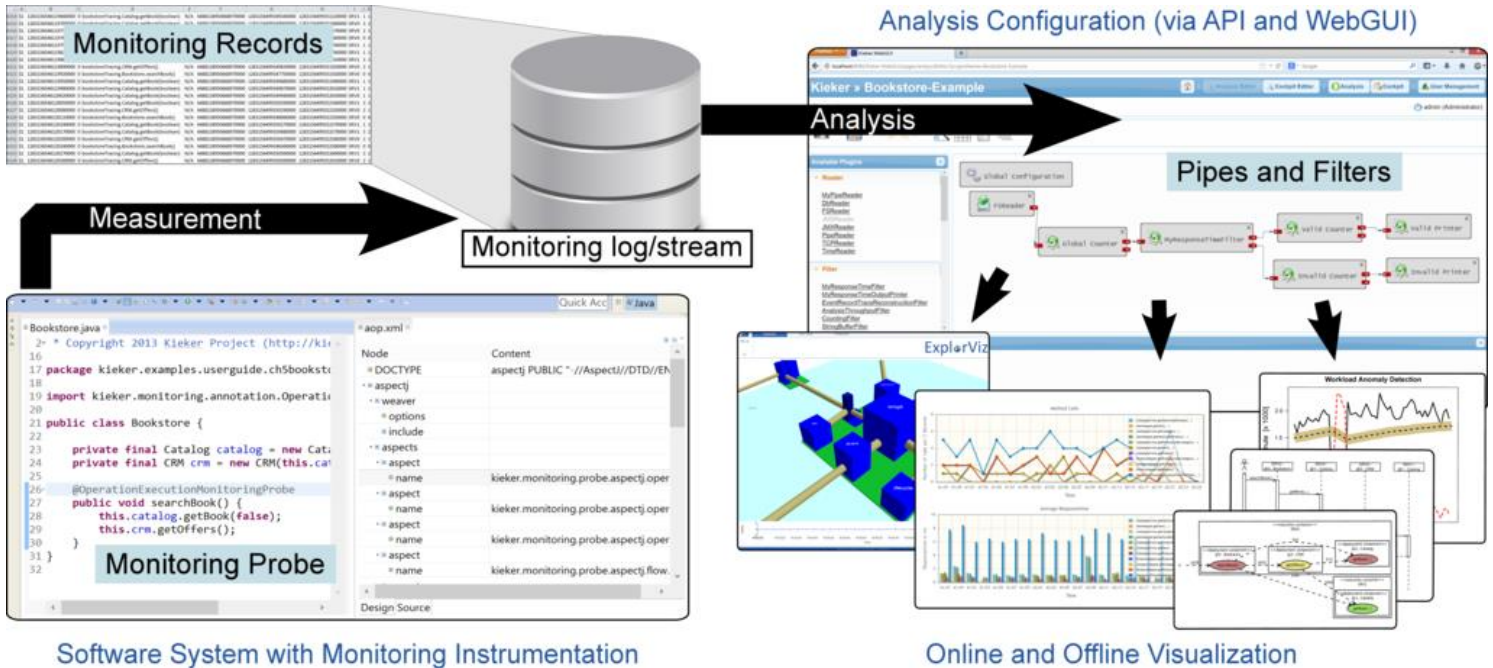


# Build Your Own Landscape





# Dynamic Software Analysis and Application Performance Management



• <http://kieker-monitoring.net>

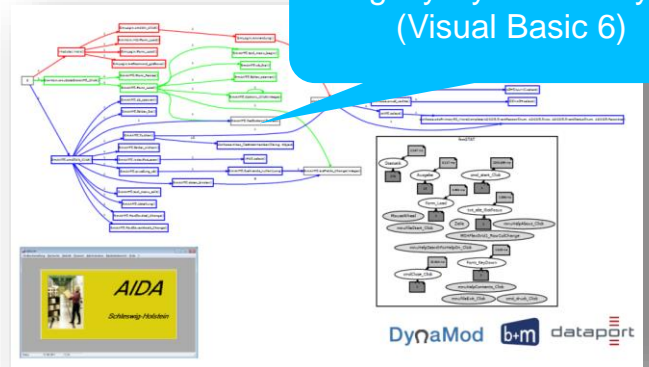
Kieker is distributed as part of SPEC® RG's repository of peer-reviewed tools for quantitative system evaluation and analysis

<http://research.spec.org/projects/tools.html>

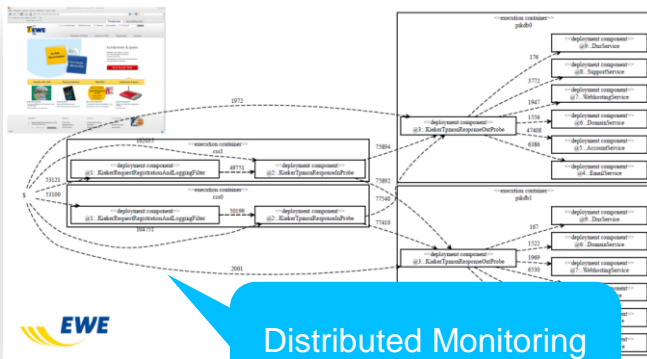




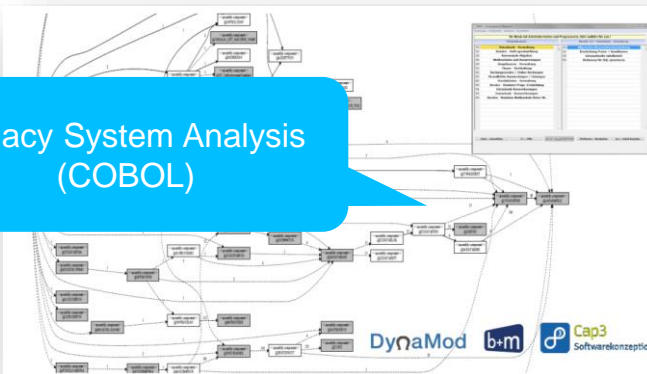
Legacy System Analysis  
(Visual Basic 6)



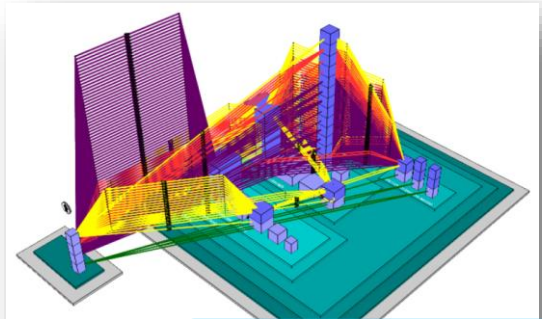
Distributed Monitoring  
(Java EE/SOAP)



Legacy System Analysis  
(COBOL)



3D Visualization  
of Concurrency



# Conclusion

- Commercial tools have most fancy visualizations  
... how useful are they really?
- Open-source tools are flexible and can be adapted as needed  
... but require a lot of work to setup
- Thanks to a more widespread adoption of (open-source) APM, it is easier to get access to rich APM data (e.g., distributed traces) and to integrate visualization approaches (e.g., via APIs)  
... if you know what and how to visualize