



# Elastic Stack

## 오픈 프로젝트 기반의 DevSecOps 구현

---

김종민

Principal Community Advocate  
@Elastic





40여 개국에 분산된 직원  
글로벌: 3000명  
국내 20여명  
(한국어 기술지원, 컨설팅)

---

2018. 10. NYSE 상장

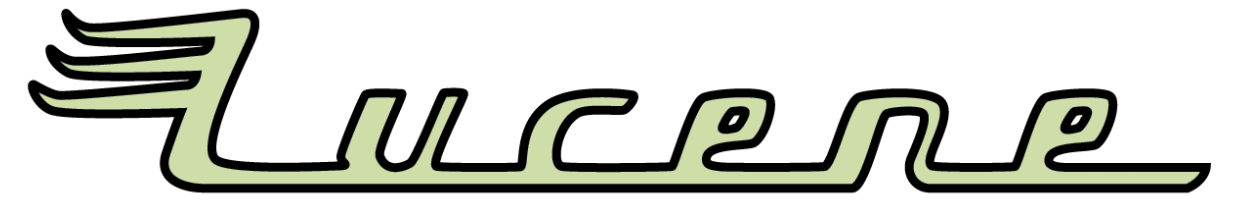


# Elastic Stack in history (& Open Source)

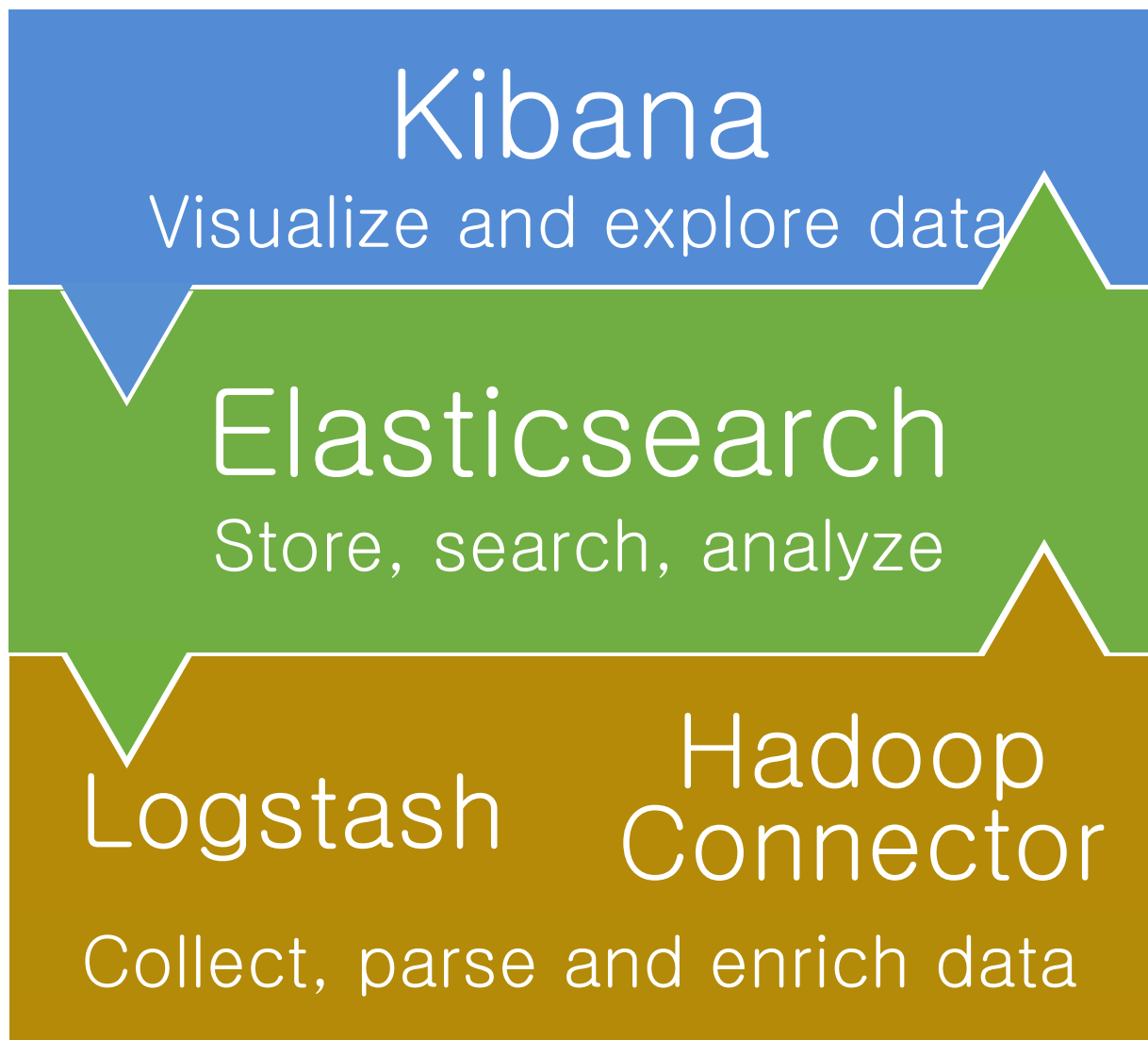
# 아파치 루씬 - Apache Lucene

1999년 더그 커팅 (Doug Cutting) 에 의해 개발

- 풀텍스트 검색 엔진을 만들 수 있는 자바 라이브러리
- 2005년 Apache Top Level 프로젝트로 선정
- 루씬을 사용해서 구현된 검색엔진
  - Elasticsearch
  - Solr
- Elastic 사 에는 20명 이상의 루씬 개발자 (PMC, Committer, Contributor) 들이 있으며 전체 루씬의 60% 이상의 기능이 Elastic 엔지니어들에 의해 개발되고 있습니다.



### Open Source Products



### Commercial Products



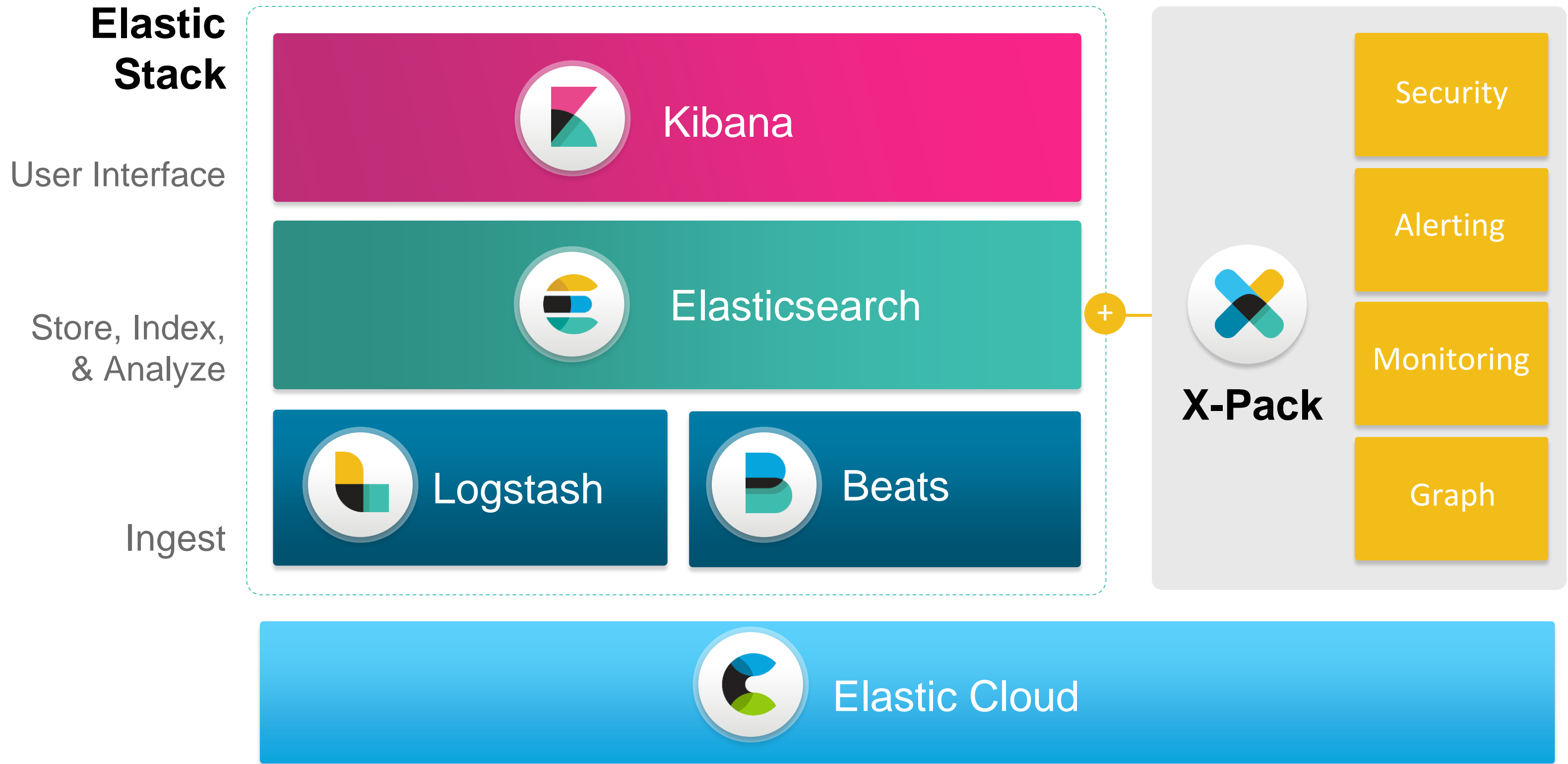
Training

Professional Services

Support Subscriptions

FOR BUILDING SCALABLE, DISTRIBUTED SYSTEMS

# Our Product Portfolio



6.4 ~ 7.x

Solutions



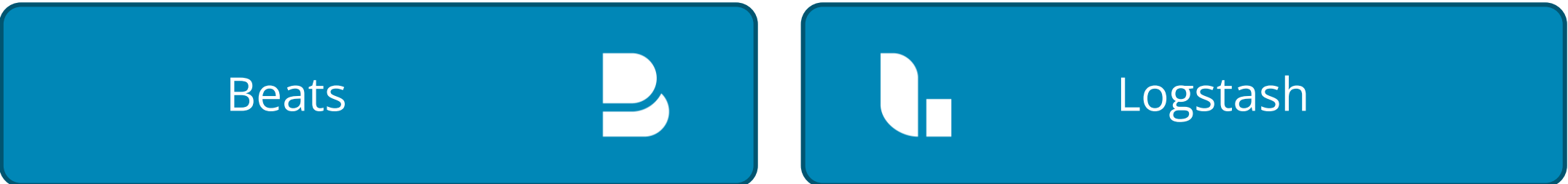
Visualize & Manage



Store, Search, & Analyze



Ingest



Deployment





# Solutions

Elastic Enterprise Search

Elastic Observability

Elastic Security



Kibana



Elasticsearch



Agent / Beats



Logstash



Endpoint Security

SaaS

Self Managed



Elastic Cloud



Elastic Cloud Enterprise



Elastic Cloud on Kubernetes



Standalone

7.x ~

# Features

STACK SECURITY

ENDPOINT PROTECTION

SIEM

REPORTING

LOGS

METRICS

APM

UPTIME

MAPS

ALERTING

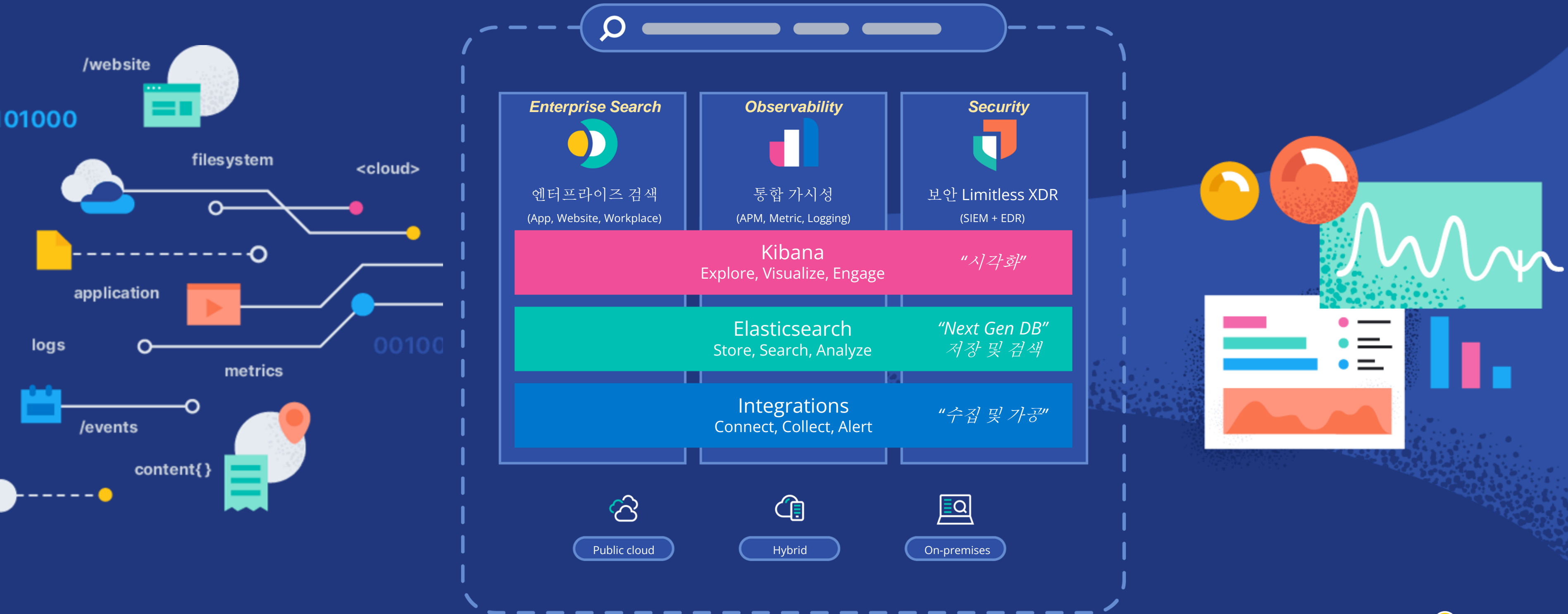
MACHINE LEARNING, GRAPH



# Elastic Stack

# Deployment methods

# Elastic은 "검색" 플랫폼입니다



# 오픈소스 기업들의 라이선스 변화

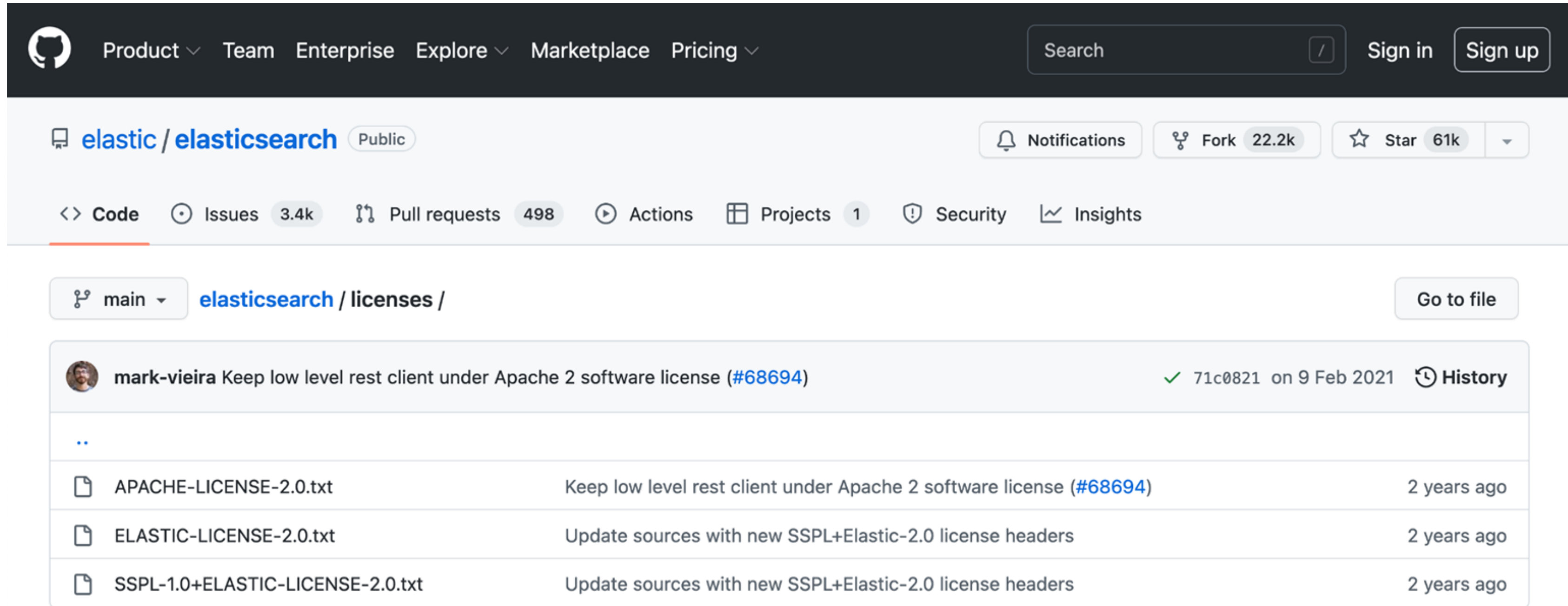
<https://www.scalevp.com/blog/making-sense-of-a-crazy-year-in-open-source>

	Before	After	What changed	Why the fuss
<b>Redis</b>	BSD 3-Clause AGPL Proprietary	BSD 3-Clause Redis Source Available Proprietary	Clarified semi-open as excluding competitors	Attempted to use Common Clause which is confusing/contested. First to provide company-specific license
<b>MongoDB</b>	Apache 2.0 AGPL Proprietary	Apache 2.0 Server Side Public Proprietary	Clarified semi-open as excluding competitors	Withdrew SSPL "open source" application from OSI consideration
<b>Elastic</b>	Apache 2.0 Proprietary	Apache 2.0 Elastic	Made proprietary code visible alongside open source code in the same codebase	Made things more open. AWS/others thought co-mingling open/proprietary is confusing and forked a distro of the open parts
<b>Confluent</b>	Apache 2.0 Apache 2.0 Proprietary	Apache 2.0 Confluent Community Proprietary	Made some open source features less so by excluding competitors	Made things less open They learned from the others and clearly communicated it as such
<b>Chef</b>	Apache 2.0 Proprietary	Apache 2.0 Apache 2.0	Made all code open source	When others were moving to semi-open/closed, they moved to all open

# Apache 2.0 → SSPL → Elastic License 2.0

OSS 라이선스는 아니지만 모든 코드는 여전히 공개되어 있으며 자유롭게 기여가 가능합니다.

<https://github.com/elastic/elasticsearch>

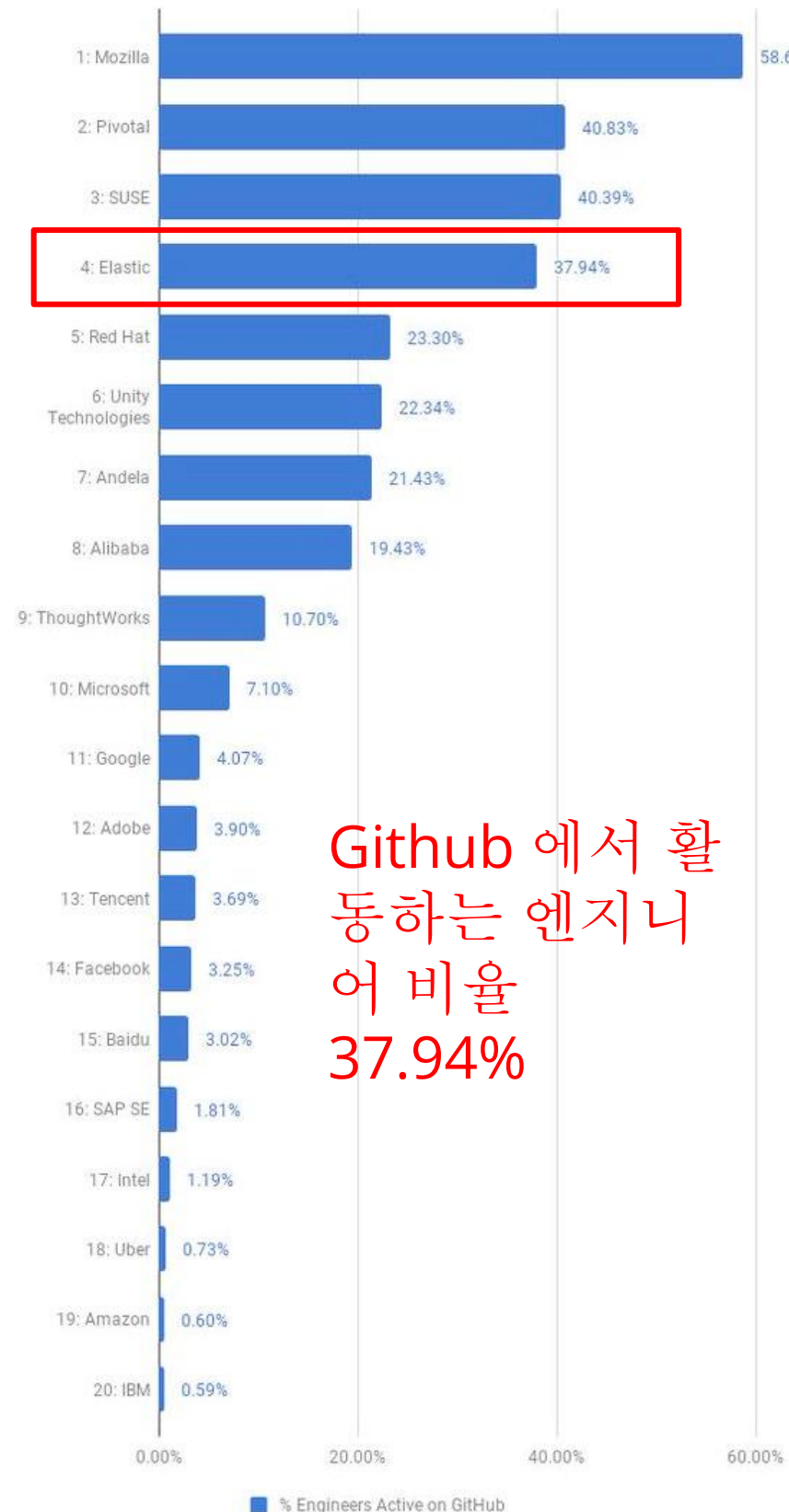


The screenshot shows the GitHub repository page for `elastic/elasticsearch`. The repository is public and has 22.2k forks and 61k stars. The current branch is `main`. The page displays a commit history for the `licenses` directory, showing a pull request by `mark-vieira` to update the license headers to SSPL+Elastic-2.0. The commit message is "Keep low level rest client under Apache 2 software license (#68694)". The commit is dated 9 Feb 2021 and is marked as merged. The commit history shows three files: `APACHE-LICENSE-2.0.txt`, `ELASTIC-LICENSE-2.0.txt`, and `SSPL-1.0+ELASTIC-LICENSE-2.0.txt`, all updated 2 years ago.

File	Commit Message	Time
..		
APACHE-LICENSE-2.0.txt	Keep low level rest client under Apache 2 software license (#68694)	2 years ago
ELASTIC-LICENSE-2.0.txt	Update sources with new SSPL+Elastic-2.0 license headers	2 years ago
SSPL-1.0+ELASTIC-LICENSE-2.0.txt	Update sources with new SSPL+Elastic-2.0 license headers	2 years ago

# Elastic Engineers @ Open Source World

Active GitHub Contributors, 2019 Q1, Ranked by % Engineers Active  
Users w/ at least 1 commit to a public repo on GitHub and received a star that year



Github 에서 활  
동하는 엔지니  
어 비율  
37.94%

## Open Source Contributor Index

Welcome to OSCI, where commercial organizations are ranked by the volume of their employees' year-to-date open source contributions on GitHub

<https://opensourceindex.io>

RANK	ORGANIZATION	ACTIVE CONTRIBUTORS
1	Google	
2 ↗ 2	Microsoft	
3	Red Hat	
4 ↗ 3	Intel	
5 ↗ 20	IBM	
6 ↗ 3	Amazon	
7 ↘ 1	Facebook	
8 ↘ 3	VMware	
9 ↘ 7	GitHub	
10 ↗ 151	SAP	

Q Organizations... Industry January 2022

RANK	ORGANIZATION
11 ↗ 17	SUSE
12 ↘ 4	Oracle
13 ↘ 2	NVIDIA
14 ↗ 81	Elastic
15 ↗ 2	Liferay
16 ↗ 2	EPAM
17 ↘ 4	Tencent
18 ↗ 3	Mozilla
19 ↗ 20	Adobe
20 ↗ 50	Alibaba

ORGANIZATION: Elastic

INDUSTRY: Technology

ACTIVE CONTRIBUTORS: 223

TOTAL COMMUNITY: 353 +340

TOP LANGUAGES: TypeScript, Java, Go

LICENSE USAGE: Other, Apache-2.0, MIT

CONTRIBUTOR GROWTH: 47.7% vs Jan-21 ↗, 182.3% vs Jan-20 ↗

ORGANIZATION: Elastic

INDUSTRY: Technology

ACTIVE CONTRIBUTORS: 223

TOTAL COMMUNITY: 353 +340

TOP LANGUAGES: TypeScript, Java, Go

LICENSE USAGE: Other, Apache-2.0, MIT

CONTRIBUTOR GROWTH: 47.7% vs Jan-21 ↗, 182.3% vs Jan-20 ↗

# 대부분의 주요 기능 무료로 사용 가능 (Free & Open)

	무료 개방형 - 베이직 <sup>1,2</sup>	플래티넘	엔터프라이즈
ELASTIC STACK 운영 및 관리			
<b>저장소 유형</b>			
(검색을 위한) 역 인덱스	✓	✓	✓
인덱스 시에 계산된 필드 평가	✓	✓	✓
런타임 필드	✓	✓	✓
런타임 필드 조회	✓	✓	✓
(구조화되지 않은 데이터를 위한) 문서 저장소	✓	✓	✓
(분석을 위한) 열 형식 저장소	✓	✓	✓
(숫자, 날짜 및 위치 정보를 위한) BKD 트리	✓	✓	✓

# 대부분의 주요 기능 무료로 사용 가능 (Free & Open)

보안 심화 기능 및 머신러닝 관련 기능에 한해 일부 유료로 제공함  
(기본적인 보안 기능은 무료)

	무료 개방형 - 베이직 <sup>1,2</sup>	플래티넘		무료 개방형 - 베이직 <sup>1,2</sup>	플래티넘
<b>ELASTIC STACK 운영 및 관리</b>					
LDAP, PKI <sup>3</sup> , 액티브 디렉터리 인증	—	✓			
Elasticsearch 토큰 서비스	—	✓	<b>MACHINE LEARNING</b>		
싱글 사인온(SAML, OpenID Connect, Kerberos)	—	✓		—	✓
				—	✓
			군집/개체군 분석	—	✓
			로그 메시지 분류	—	✓
			희귀 분석	—	✓



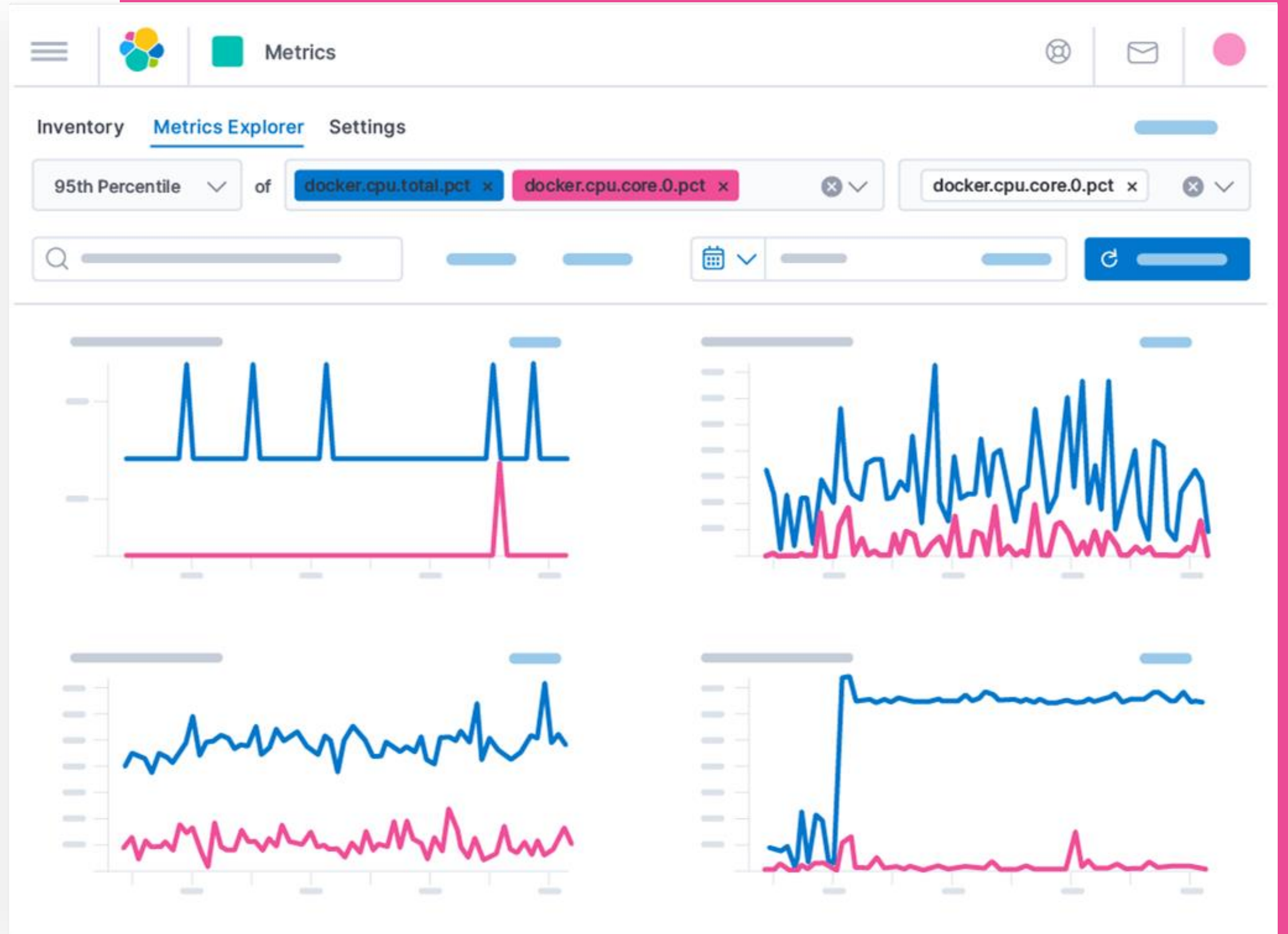
# Observability

APM

Logs

Metrics

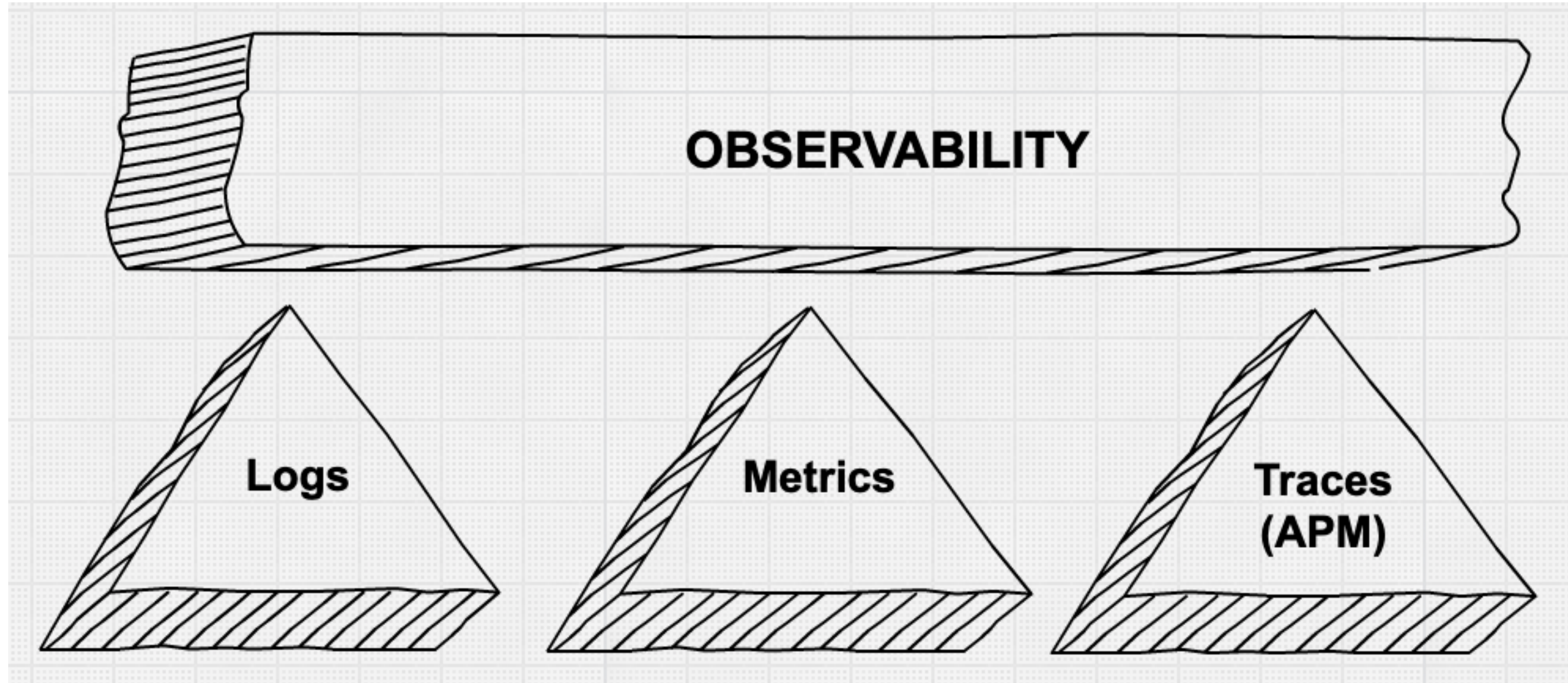
Uptime





# Observability

통합모니터링의 세 가지 데이터 타입



# 로그

## 특정 이벤트가 발생할 때마다 생성되는 데이터

```
64.242.88.10 - - [07/Mar/2017:16:10:02 -0800] "GET /mailman/listinfo/hsdivision HTTP/1.1" 200 6291
64.242.88.10 - - [07/Mar/2017:16:11:58 -0800] "POST /twiki/bin/view/TWiki/WikiSyntax HTTP/1.1" 404 7352
64.242.88.10 - - [07/Mar/2017:16:20:55 -0800] "GET /twiki/bin/view/Main/DCCAndPostFix HTTP/1.1" 200 5253
```

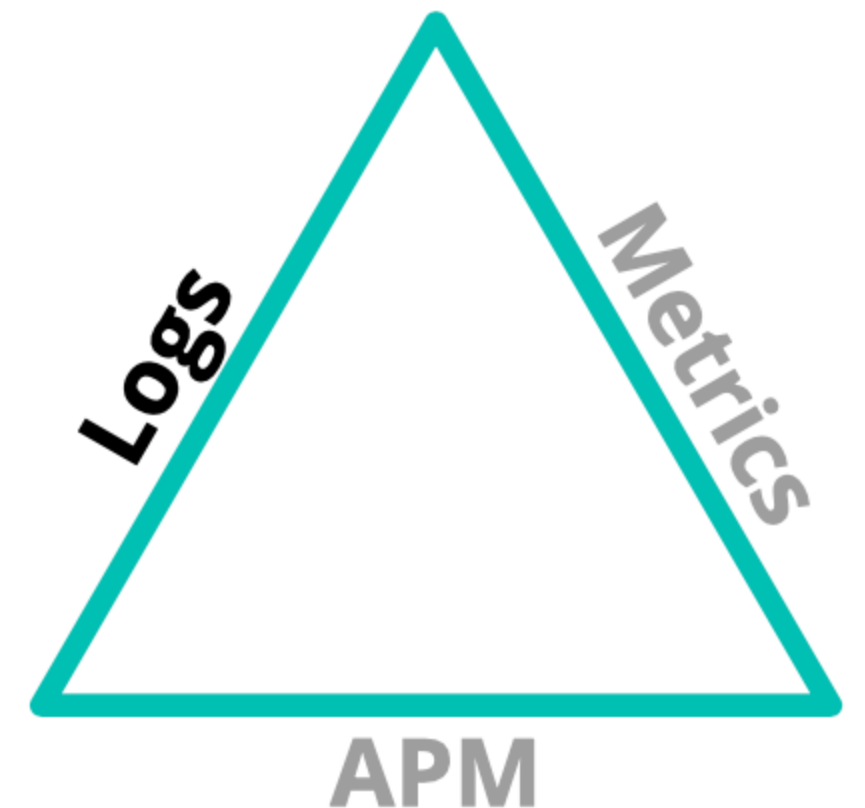
## 각 로그는 ‘어떤’ 일이 일어났는지 기록

### 장점:

- 주로 애플리케이션 또는 서비스가 돌아가는 호스트에 남음
- 사람이 읽을 수 있고 커스터마이징이 가능

### 단점:

- 애플리케이션 수준이 아닌 컴포넌트 수준에서 기록됨
- 출력하지 않으면 기록이 남지 않음
- 포맷이 중요함



# 메트릭

주기적으로 특정 값을 측정하는 데이터

07/Mar/2017	16:10:00	all	2.58	0.00	0.70	1.12	0.05	95.55	server1	containerX	regionA
07/Mar/2017	16:20:00	all	2.56	0.00	0.69	1.05	0.04	95.66	server2	containerY	regionB
07/Mar/2017	16:30:00	all	2.64	0.00	0.65	1.15	0.05	95.50	server2	containerZ	regionC

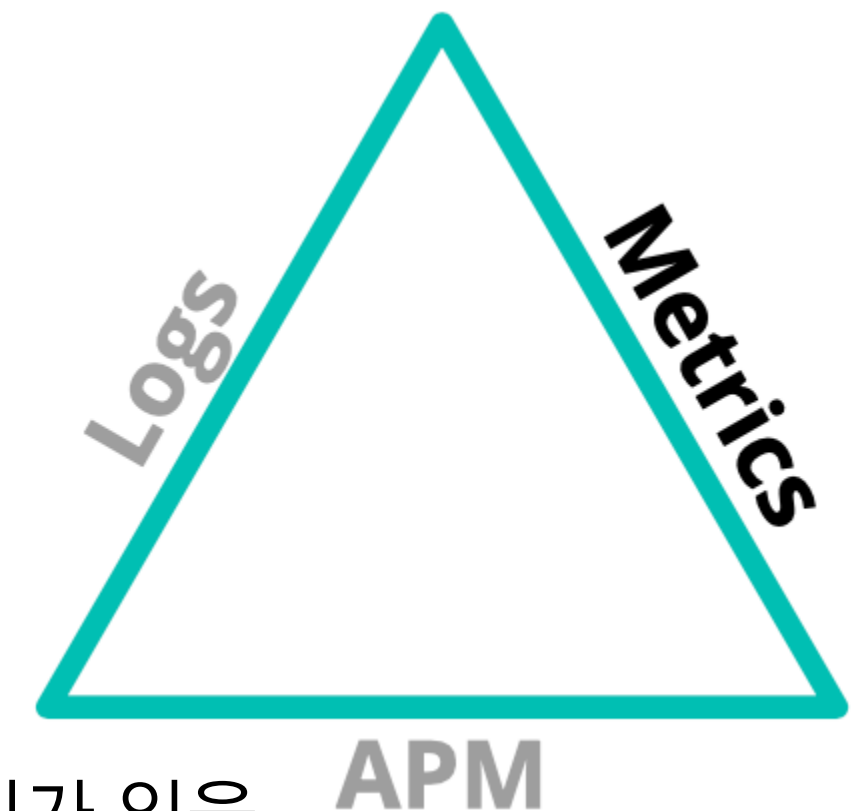
예) 매 10초마다 CPU 사용률을 측정해서 기록 함

장점:

- 트렌드와 이력을 보여줌
- 특정 사건이나 이상치를 잡기 위해 간단한 경고를 만들기 용이함

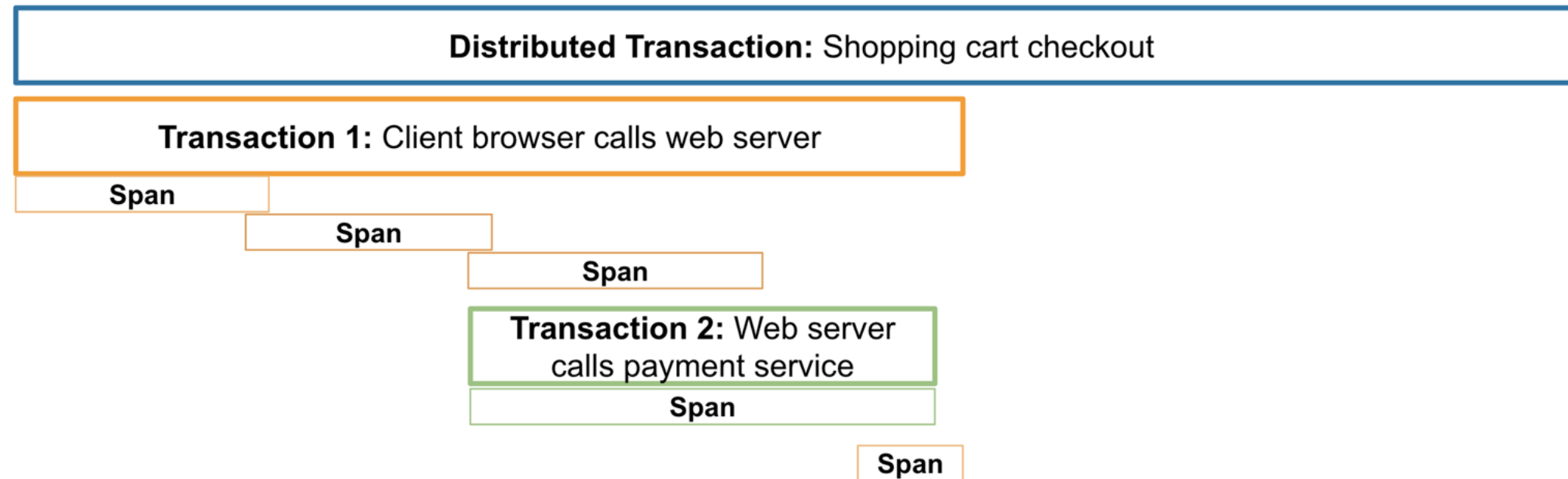
단점:

- 측정 주기 안에 있는 이상치는 평균에 묻혀버릴 수 있음
- 호스트 또는 네트워크에 기반하지만 컨테이너 환경에서는 총계값이 왜곡될 소지가 있음



# 트레이스

애플리케이션 코드 안에서 일어나는 활동을 기록한 데이터



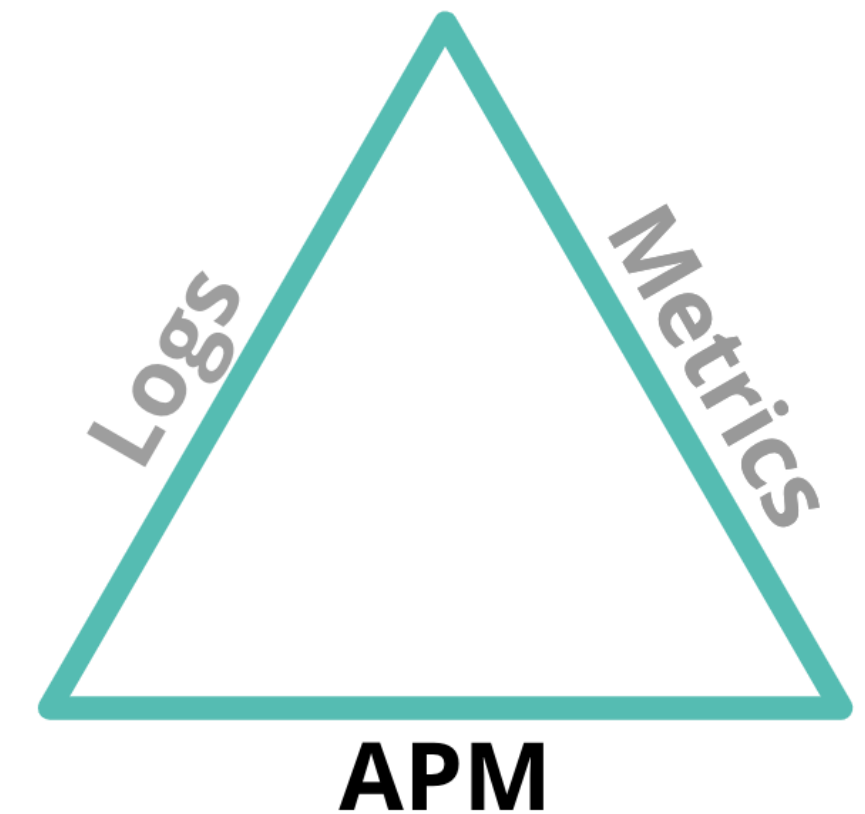
각 트랜잭션마다 관련된 모든 컴포넌트와 그 세부 실행 시간을 추적

장점:

- 근본 원인 분석을 가능케 하는 매우 상세한 정보 제공

단점:

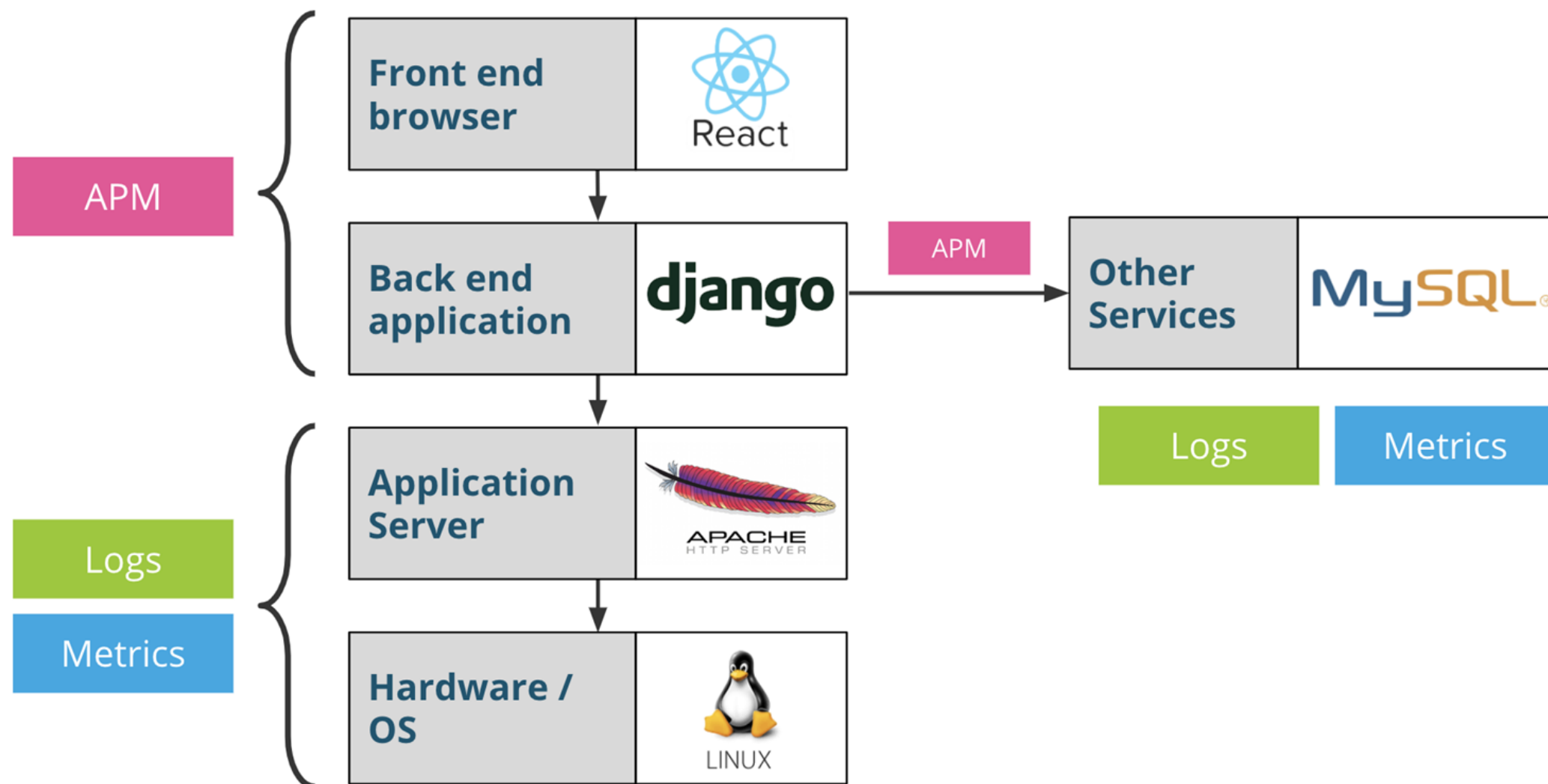
- 데이터 양이 매우 많고 내용이 장황함
- 애플리케이션 코드를 인스트루먼트(instrument) 할 필요가 있음



# 로그 + 메트릭 + 트레이스

통합 모니터링을 위해서는 세 가지 모두가 필요함

- 웹서버에 “로그” 형태로 기록되는 사용자 요청과 에러 내용
- 운영체제 API 호출로 발생하는 CPU와 RAM 사용 “메트릭”
- 애플리케이션의 로딩 시간과 그 상세 내역이 기록된 “트레이스”



# 트레이스 데이터

예: 응답 시간 또는 실행 시간이 느린 경우

03:43:45 Request "GET cyclops.ESProductDetailView"

03:43:57 Response "cyclops.ESProductDetailView 200 OK"

12 초

# 트레이스 데이터

예: 오류 및 예외 발생

03:43:59 Request "POST /api/checkout"

03:43:59 Response "/api/checkout 500 ERROR"

# 분산 트레이싱

## 단일 트랜잭션





# 분산 트레이싱

다수 서비스

트레이스 A

트랜잭션 1

스팬

스팬

스팬

트랜잭션 2

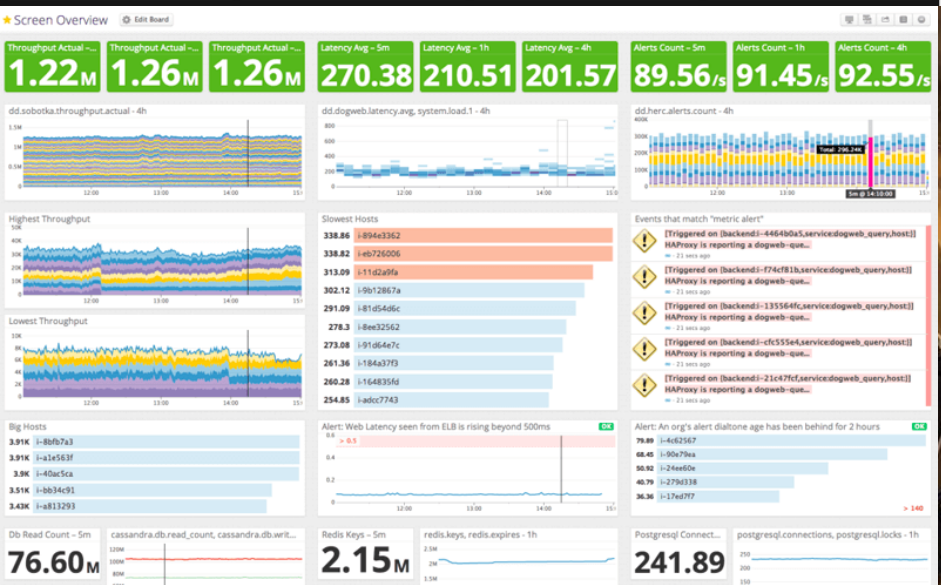
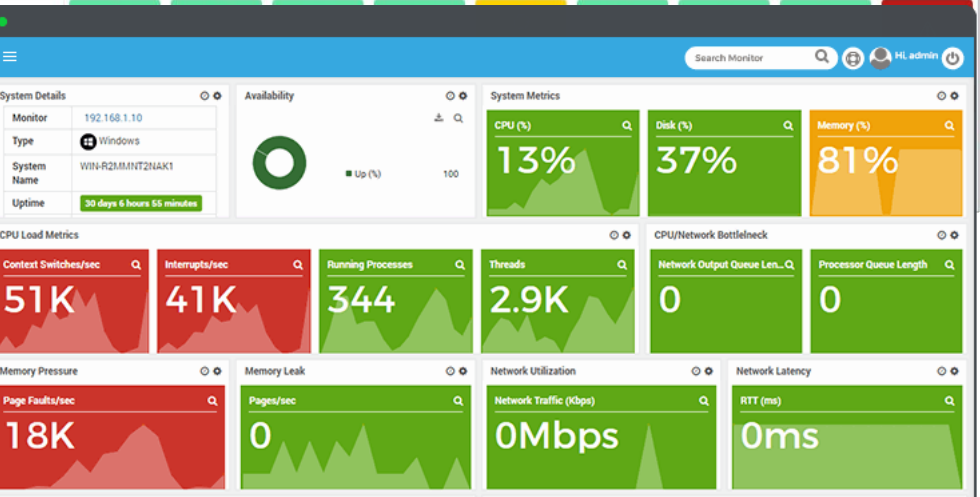
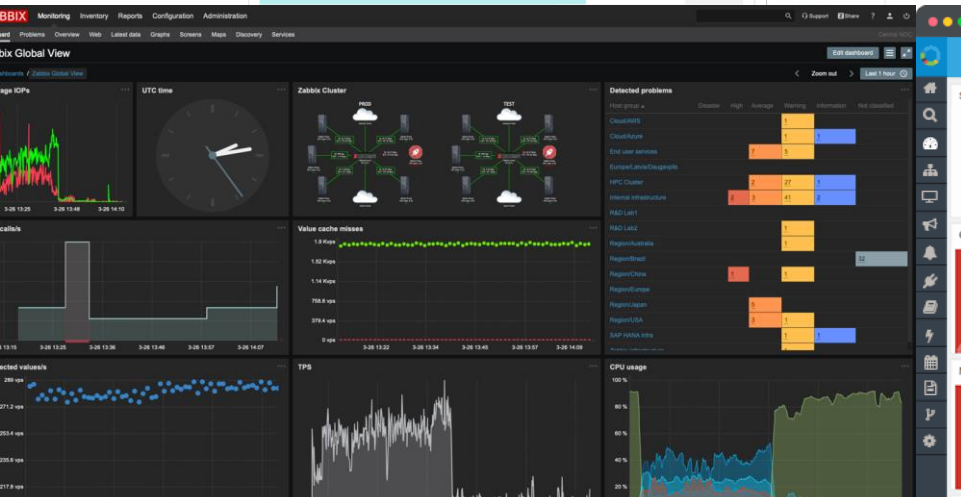
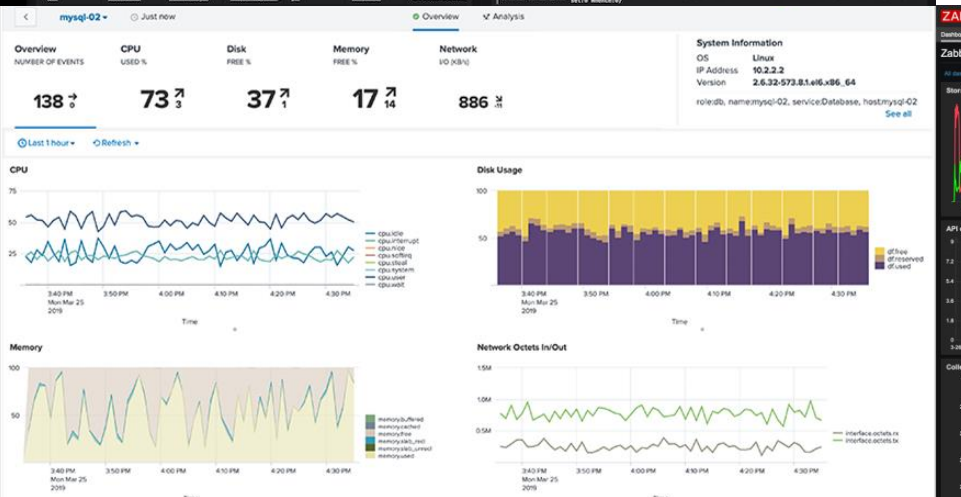
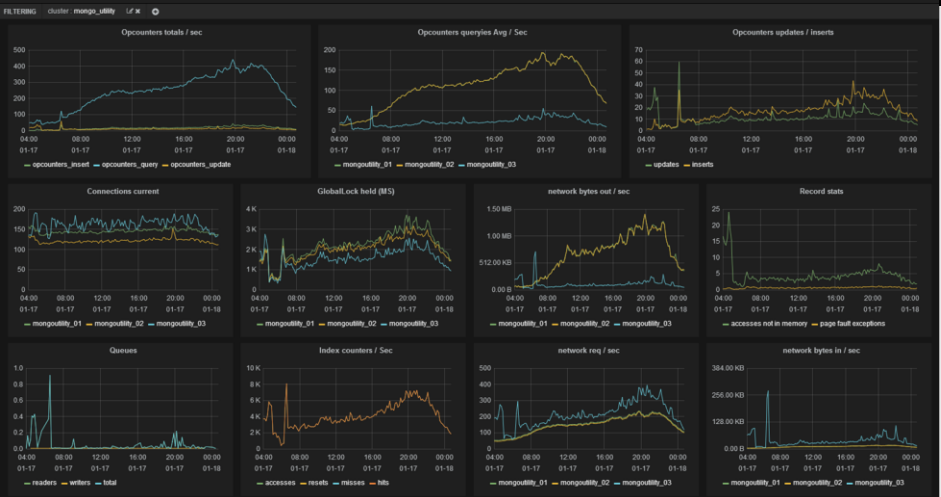
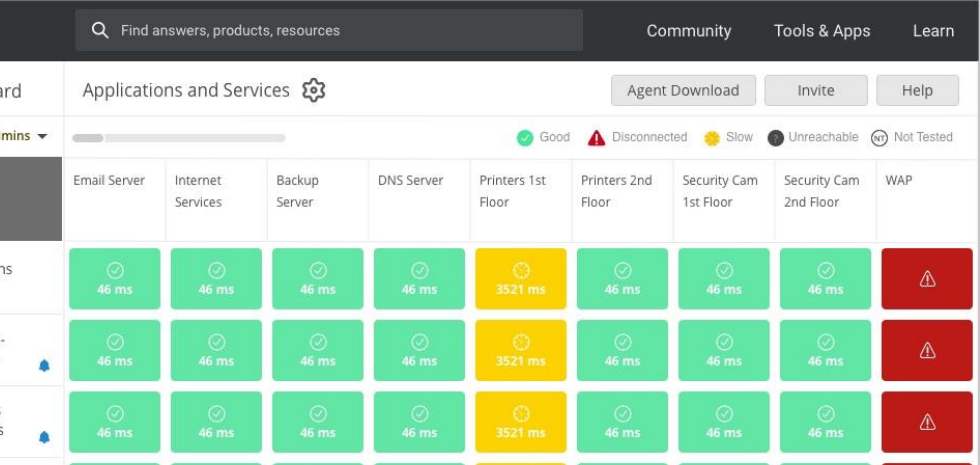
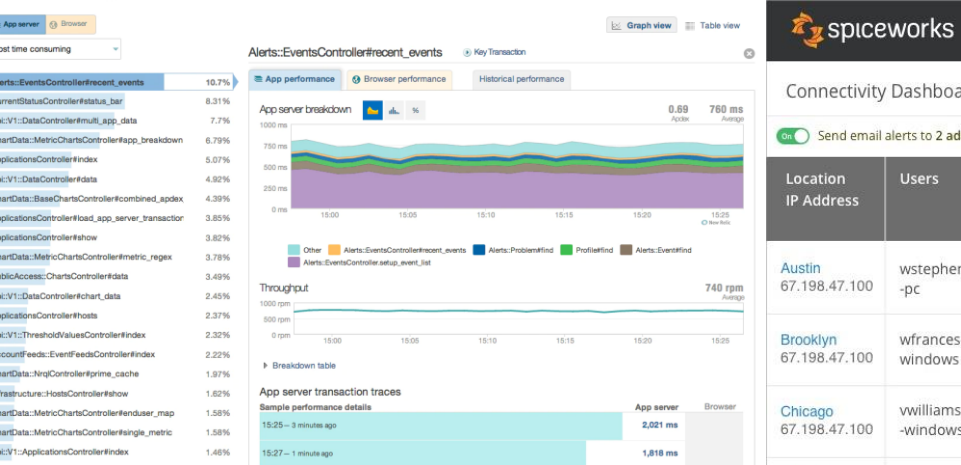
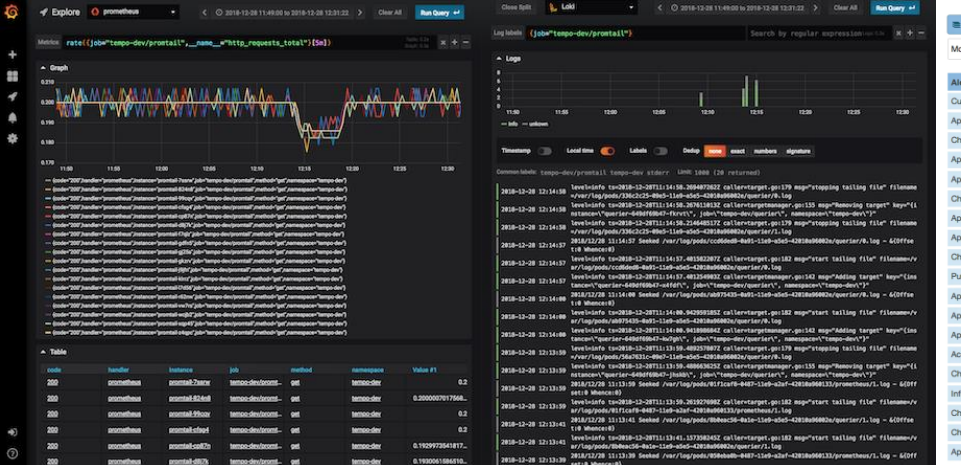
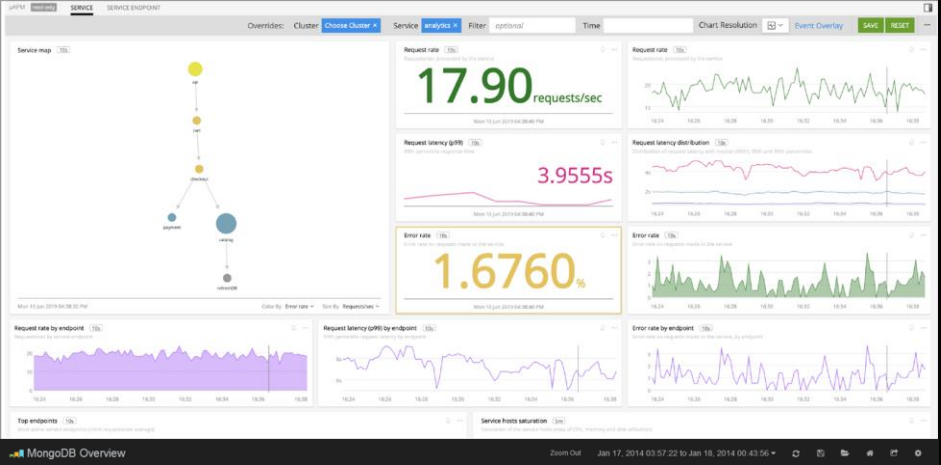
스팬

스팬

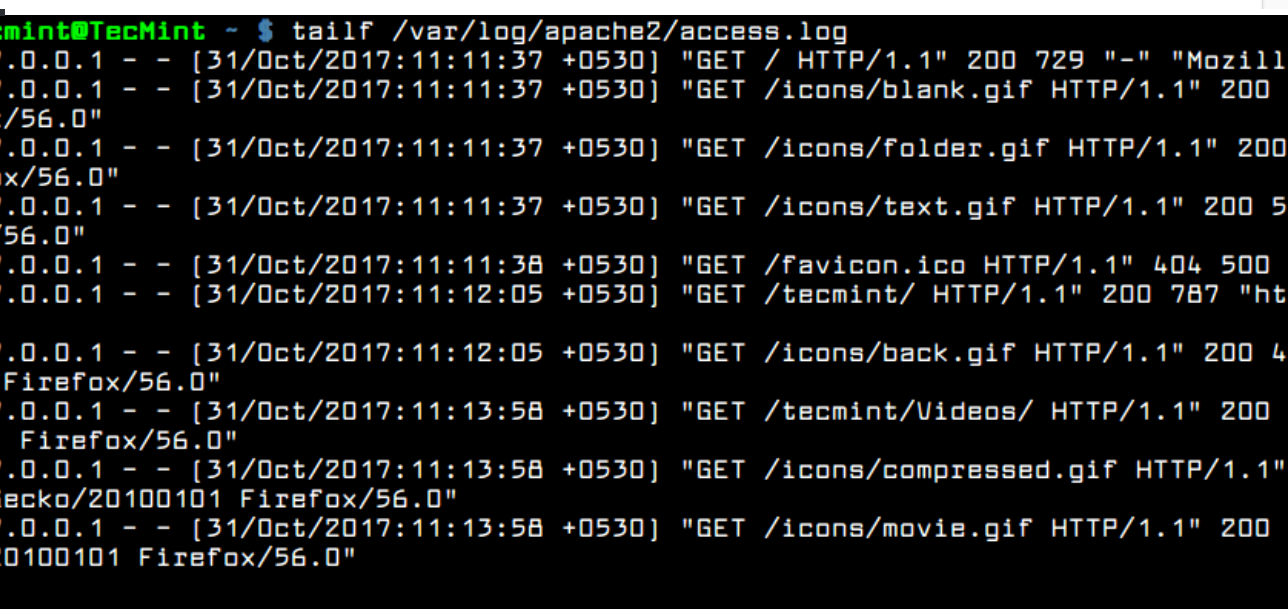
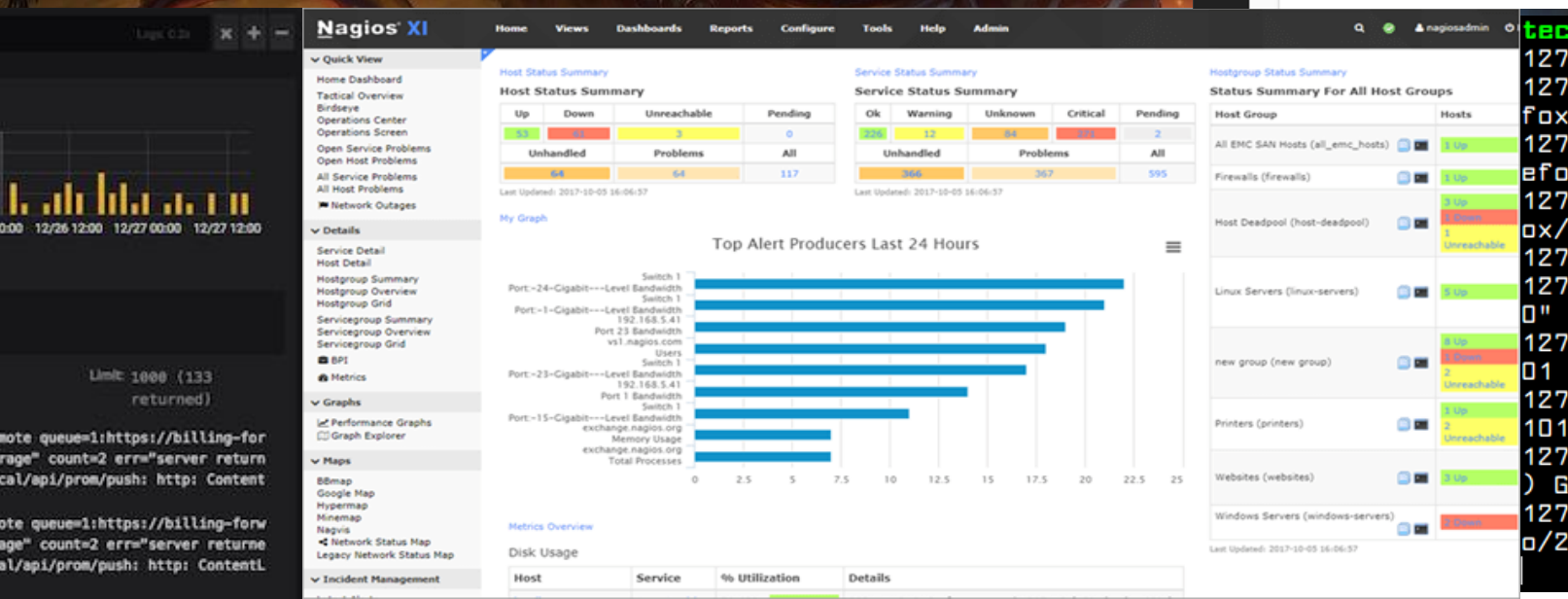
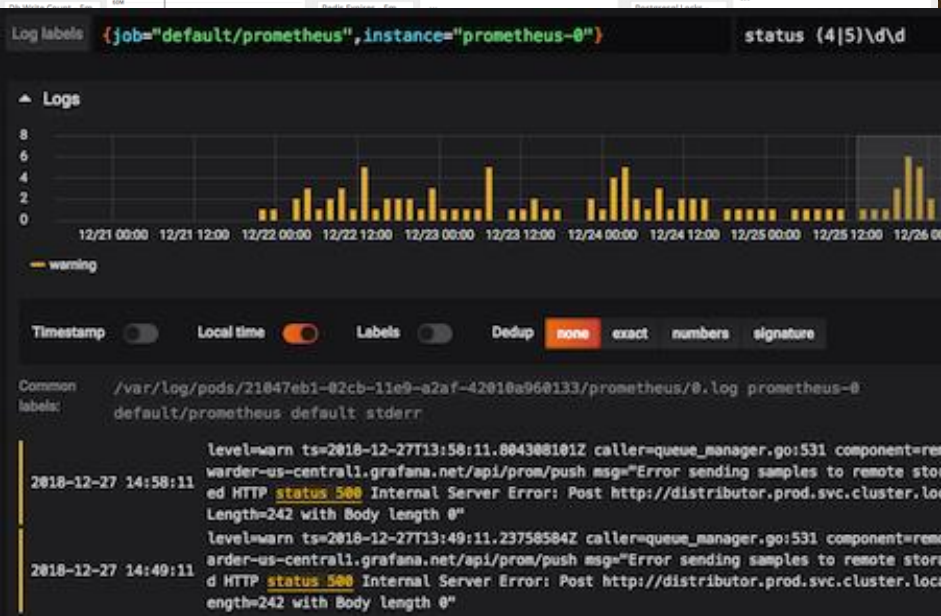
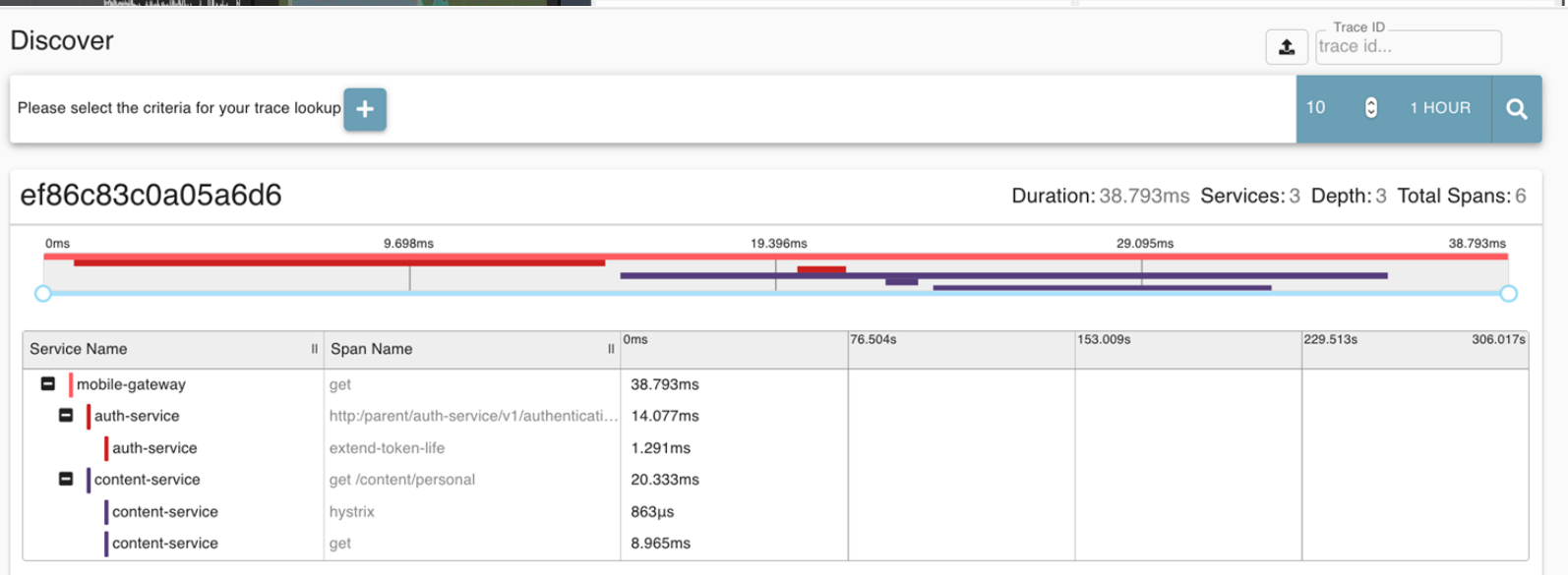
트랜잭션 3

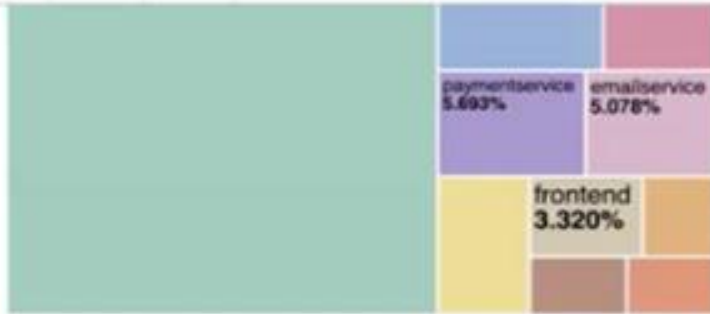
스팬

스팬

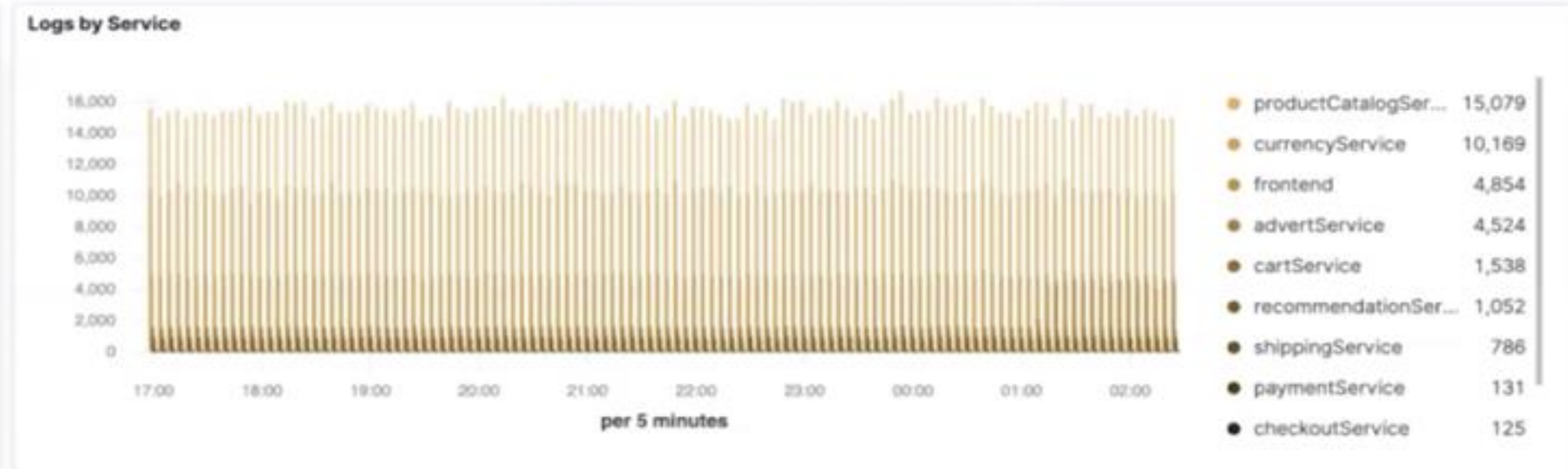
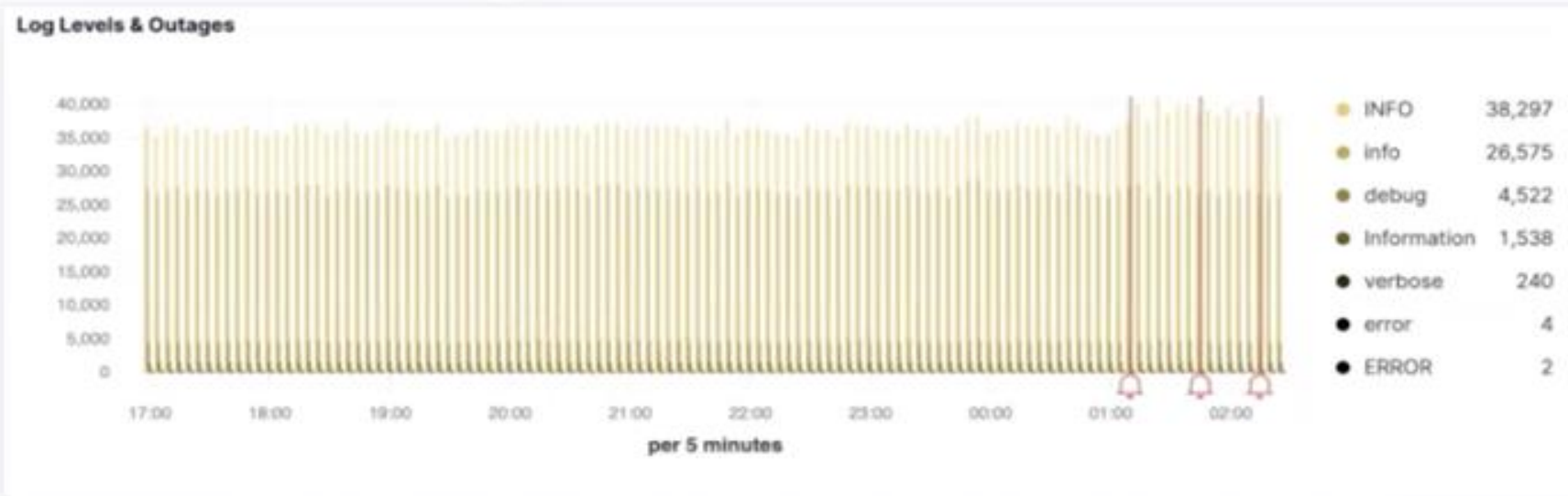


- 7015c90 - (HEAD, origin/master, origin/HEAD, master) Updated iPhones graphic with new screenshot (11 days ago) <@Nutter>
- 9bfa321 - Updated README (2 weeks ago) <@Nutter>
- dd18305 - Updated to new design (2 weeks ago) <@Nutter>
- b2e36c6 - Added fallback for missing avatar (2 weeks ago) <@Nutter>
- 8678978 - Removed Disk caching of avatar (6 weeks ago) <@Nutter>
- 39d2c3c - Preparing for App Store submission (9 weeks ago) <@Nutter>
- 4fd7261 - Fixed cell bg when user only has 1 badge. (9 weeks ago) <@Nutter>
- 13bf363 - Fix image (10 weeks ago) <@Nutter>
- 705363e - Try screenshots with transparent background (10 weeks ago) <@Nutter>
- bd4602e - Correct image location in README (10 weeks ago) <@Will McKenzie>
- 483d05b - Added iPhone screenshots to README (10 weeks ago) <@Nutter>
- 2d7a6d7 - Adding license (10 weeks ago) <@Nutter>
- d6c6660 - Adding to .gitignore (10 weeks ago) <@Nutter>
- 0e2af71 - Still trying to ignore UserInterfaceState file (10 weeks ago) <@Nutter>
- cca3d3b - ignore interfaceState file (10 weeks ago) <@Nutter>
- 2b1e812 - Added first draft of README (10 weeks ago) <@Nutter>
- 38586ba - Bug fixes and UI Improvements. New Icons added (2 months ago) <@Nutter>
- 968d04c - Removed extra app icons (3 months ago) <@Nutter>
- ecf783d - Various updates (3 months ago) <@Nutter>
- 6ce53d9 - Removed SDURL files from root (3 months ago) <@Nutter>
- 62e0679 - Made sure images cache (3 months ago) <@Nutter>
- 1397dF2 - Ran Arc conversion to improve memory management. Refactoring and bug fixes (3 months ago) <@Nutter>
- c3061b8 - Added pull to refresh functionality (3 months ago) <@Nutter>
- 926616b - Made sure iPhone badge and accomplishments view controllers reload when user changed. (3 months ago) <@Nutter>
- 83cd921 - Stopped badges images using separate threads. (3 months ago) <@Nutter>

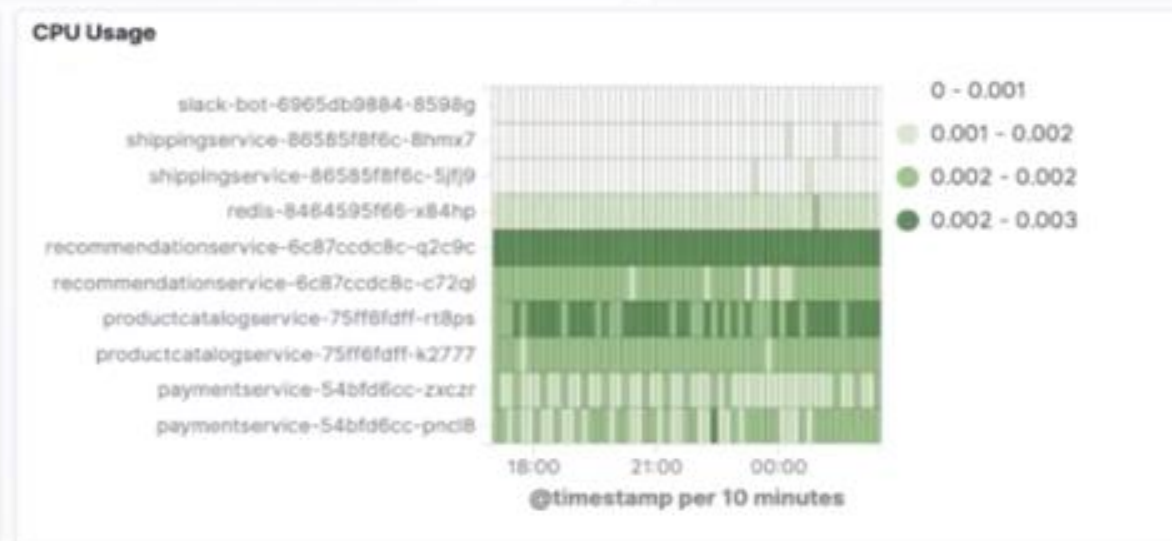
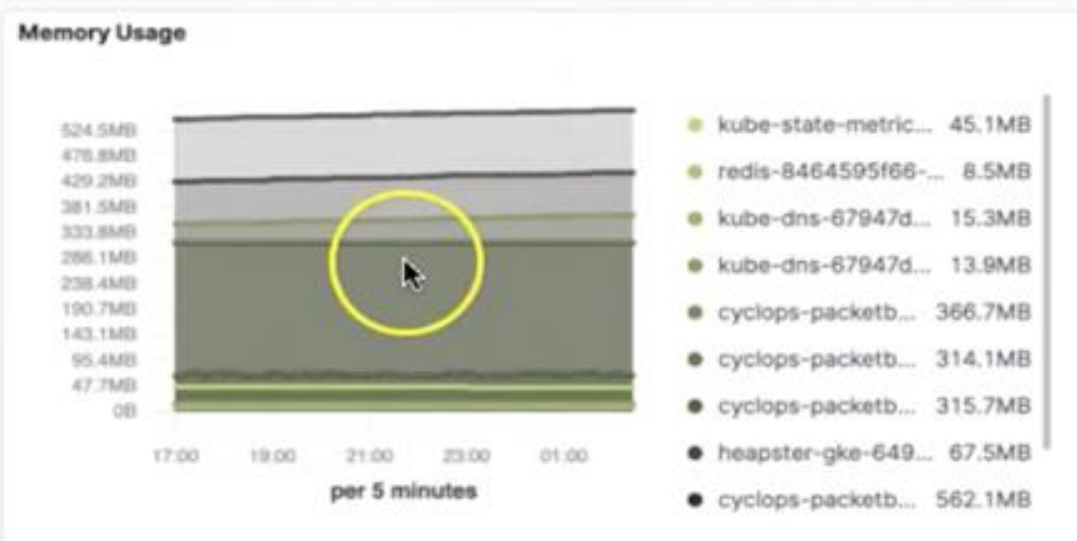




Logs



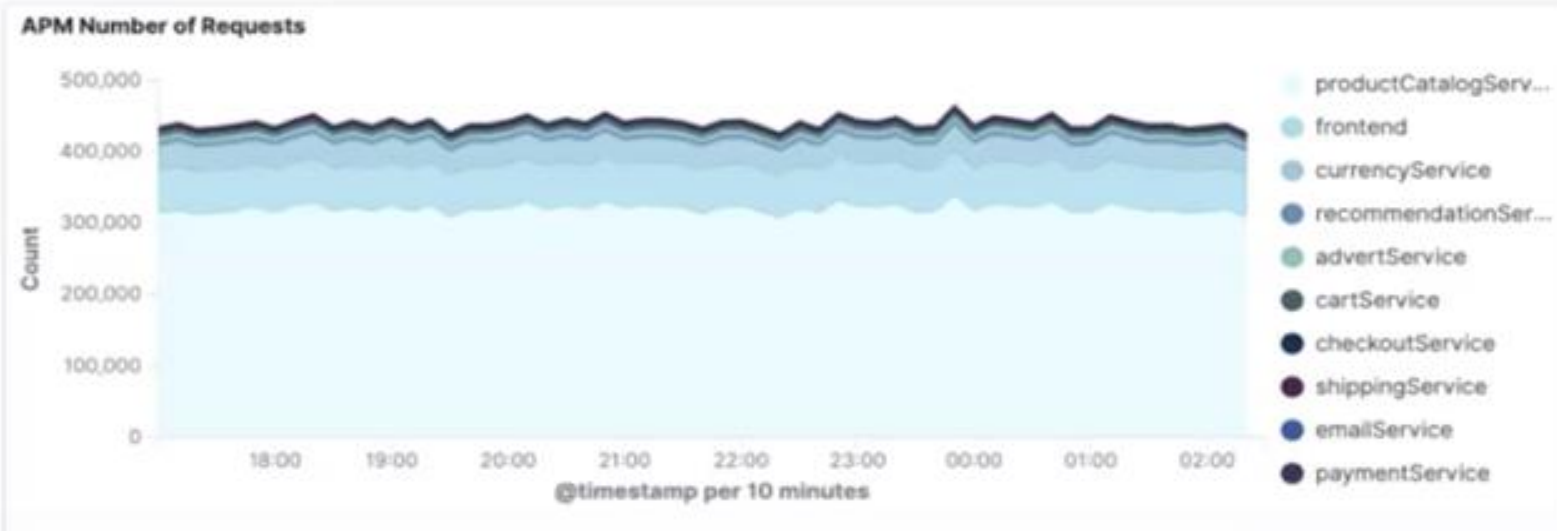
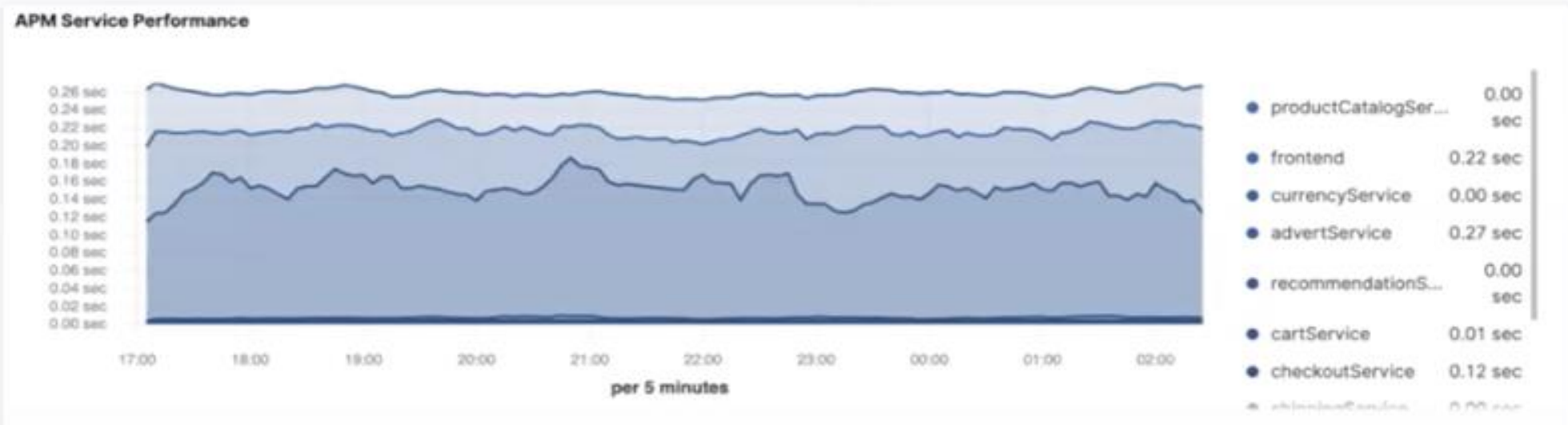
Metrics



### Hosts [Metricbeat Redis] ECS

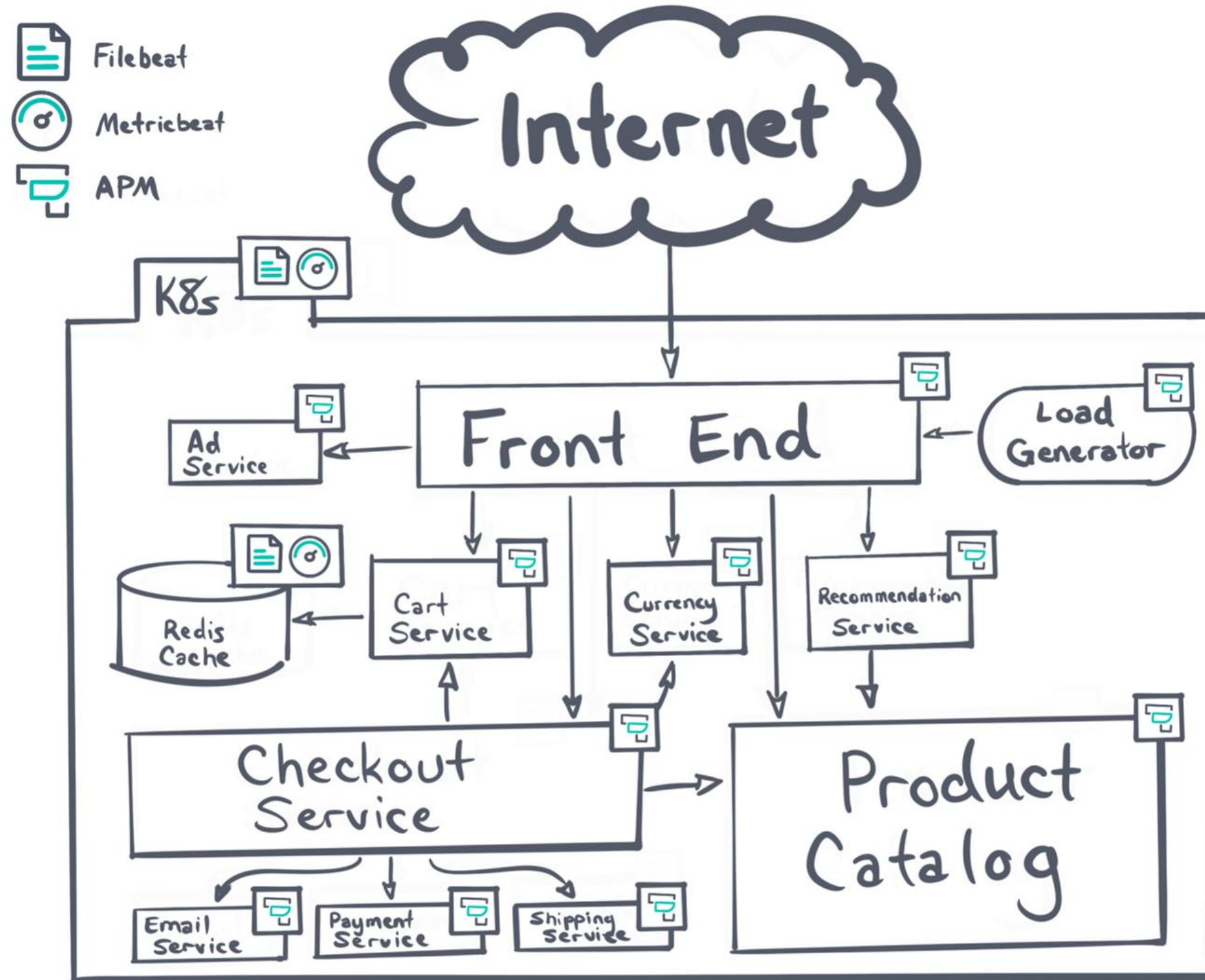
service.address: Descending	Uptime (s)	PID	Memory	CPU used (user)	CPU used (system)
10.80.9.59:6379	42,046	1	1,448,096	61.744	72.956

APM



# Demo

# Architecture



# Service Map

The screenshot displays the Elastic Service Map interface. At the top, the Elastic logo is on the left, and a search bar contains the text "Find apps, content, and more. Ex: Discover". The navigation bar includes "Observability", "APM", and "Service Map". On the right of the navigation bar are links for "Settings", "Anomaly detection", "Alerts and rules", and "Add data".

The main content area is titled "Service Map" and includes a dropdown menu for "Environment" set to "All". Below this, there are controls for "Comparison" (checked), "Day before", a date selector for "Last 1 hour", and a "Refresh" button.

The central part of the interface shows a dependency graph on a grid background. A red arrow points to a node labeled "JS frontend-rum". This node connects to "frontend-node". From "frontend-node", arrows point to "Enterprise Search", "advertService", "4 resources", and "GO". "Enterprise Search" connects to "elasticsearch". "advertService" connects to "elasticsearch". "GO" connects to "cartService", "currencyService", and "emailService". "cartService" connects to "redis".

The left sidebar contains a navigation menu with sections: "Observability" (Overview, Alerts, Cases), "Logs" (Stream, Anomalies, Categories), "Infrastructure" (Inventory, Metrics Explorer), "APM" (Services, Traces, Dependencies, **Service Map**), "Uptime" (Monitors, TLS Certificates), and "User Experience".

# Service Map

The screenshot displays the Elastic Service Map interface. At the top, the Elastic logo and a search bar are visible. The navigation bar includes 'Observability', 'APM', and 'Service Map'. The main content area is titled 'Service Map' and features a dropdown for 'Environment' set to 'All'. Below this, there are controls for 'Comparison' (checked), 'Day before', 'Last 1 hour', and a 'Refresh' button.

The central part of the interface shows a network diagram of services. A red arrow points to a 'Service Details' panel for the 'frontend-rum' service. This panel displays the following metrics:

- Latency (avg.): 2,150 ms
- Throughput (avg.): 9.3 tpm

The 'Service Details' panel also includes a 'Service Details' button (highlighted in blue) and a 'Focus map' button (highlighted in light green). The network diagram shows 'frontend-rum' connected to various services, including 'Enterprise Search', 'advertisService', 'elasticsearch', 'cartService', 'currencyService', 'emailService', and 'redis'. A '4 resources' node is also visible, connected to 'frontend-rum'.

# Service Details

The screenshot shows the Elastic APM interface for the 'frontend-rum' service. The left sidebar contains navigation options like Overview, Alerts, Cases, Logs, Stream, Anomalies, Categories, Infrastructure, Inventory, Metrics Explorer, APM, Services, Traces, Dependencies, Service Map, Uptime, Monitors, TLS Certificates, User Experience, and Dashboard. The main content area is titled 'frontend-rum JS' and includes tabs for Overview, Transactions, Errors, Infrastructure, Service Map, and Alerts. A dropdown menu is open, showing transaction types: http-request, user-interaction (highlighted with a green bar and a red arrow), route-change, and page-load. Below this is a search bar for transactions, errors, and metrics, with a 'Comparison' checkbox checked and a 'Day before' time range selected. A 'Latency' chart shows average latency over time, with a 'Day before' comparison line. A 'Throughput' chart shows transaction volume over time. A 'Transactions' table lists specific transactions with their latency, throughput, and failed transaction rate.

**Observability**

frontend-rum JS

Environment: All

Overview Transactions Errors Infrastructure Service Map Alerts

http-request  
✓ user-interaction  
route-change  
page-load

Search transactions, errors and metrics (E.g. transaction.duration.us > 300000 AND http.response.status\_code >= 400)

Comparison Day before Last 1 hour Refresh

**Latency** Metric Average

1,500 ms  
1,000 ms  
500 ms  
0 ms

14:20:00 14:25:00 14:30:00 14:35:00 14:40:00 14:45:00 14:50:00 14:55:00 15:00:00 15:05:00 15:10:00 15:15:00

● Average ● Day before

**Throughput** 40 tpm 20 tpm

**Transactions** View transactions

Name	Latency (avg.)	Throughput	Failed transaction rate	Impact
Click - button["Add To Cart"]	3,119 ms	6.1 tpm	N/A	Progress bar
Click - header	510 ms	22.0 tpm	N/A	Progress bar
Click - button["Checkout"]	704 ms	2.4 tpm	N/A	Progress bar



# Service Details

elastic  🌐 🔔 👤

Observability APM Services frontend-rum Transactions Click - button["Add To Cart"] Settings ⚠️ Anomaly detection Alerts and rules + Add data

### Observability

- Overview
- Alerts
- Cases
- Logs
- Stream
- Anomalies
- Categories

### Infrastructure

- Inventory
- Metrics Explorer

### APM

- Services**
- Traces
- Dependencies
- Service Map

### Uptime

- Monitors
- TLS Certificates

### User Experience

- Dashboard

## Latency distribution

367 total transactions ✕ Selection: 4,817 ms - 12 s

Transactions

Latency

● All transactions

### Trace sample

1 of 133 Investigate View full trace

a minute ago | 8,454 ms (100% of trace) | <http://frontend-ecom-nginx-7/products/women/T-shirt/EV421DA...> | Mobile Safari (15.4)

### Timeline

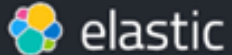
Metadata Logs

Services ● frontend-rum ● frontend-node ● cartService

0 ms 1,000 ms 2,000 ms 3,000 ms 4,000 ms 5,000 ms 6,000 ms 7,000 ms 8,454 ms

21	Click - button["Add To Cart"]	8,454 ms
1	POST /api/add-cart	8,453 ms
1	HTTP 2xx POST /api/add-cart	8,353 ms
3	OK /gallivant.CartService/AddItem	8,340 ms

# Service Details



Observability APM Services frontend-rum Transactions Click - button["Add To Cart"] Settings Anomaly detection Alerts and rules Add data

## Transaction details

Investigate

Service: **frontend-node** Transaction: **POST /api/add-cart**

2 minutes ago | 8,353 ms (99% of trace) | POST http://frontend-node-7:3000/api/add-cart | 200 OK | Mobile Safari (15.4)

### Metadata

 [How to add labels and other data](#)

### labels

labels.userId	synthetic-new-york-620713
---------------	---------------------------

### @timestamp

@timestamp	2022-09-05T06:17:07.136Z
------------	--------------------------

### agent

agent.name	nodejs
agent.version	3.10.0

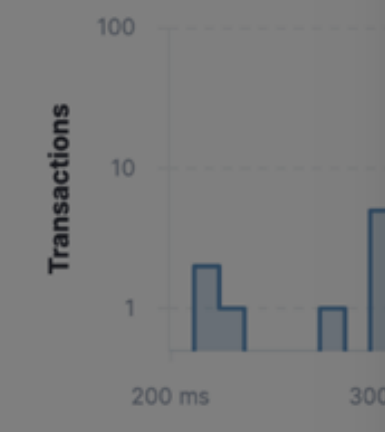
### client

client.geo.city_name	Boydton
client.geo.continent_name	North America
client.geo.country_iso_code	US

### Observability

- Overview
- Alerts
- Cases
- Logs
- Stream
- Anomalies
- Categories
- Infrastructure
  - Inventory
  - Metrics Explorer
- APM
  - Services
    - Traces
    - Dependencies
    - Service Map
- Uptime
  - Monitors
  - TLS Certificates
- User Experience
  - Dashboard

### Latency distribution



Transactions

200 ms 300 ms

All transactions

### Trace sample

2 minutes ago | 8,454 ms

Timeline Metadata

Services frontend-rum

0 ms

- 21 Click -
- 1 POST /api
- 1 HTTP 2
- 3 OK /g

# Service Details

The screenshot displays the Elastic APM interface for the 'frontend-node' service. The breadcrumb navigation shows the path: Observability > APM > Services > frontend-node > Transactions > POST /api/add-cart. The main content area is titled 'frontend-node' and includes an 'Environment' dropdown set to 'All'. A navigation bar contains tabs for Overview, Transactions (selected), Dependencies, Errors, Metrics, Infrastructure, Service Map, Logs, and Alerts. A search bar for transactions is present, along with filters for 'request' type, 'Comparison' (checked), 'Day before' period, and 'Last 1 hour' time range. A 'Refresh' button is also visible.

The primary chart is 'POST /api/add-cart', which includes three sub-charts:

- Latency:** A line chart showing average latency in milliseconds over time. The y-axis ranges from 0 to 4,000 ms. The x-axis shows time from 14:30:00 to 15:15:00. A legend indicates 'Average' (blue) and 'Day before' (light blue).
- Throughput:** A line chart showing transactions per minute (tpm) over time. The y-axis ranges from 0 to 60 tpm. The x-axis shows time from 14:30:00 to 15:15:00. A legend indicates 'Throughput' (green) and 'Day before' (light green). A red arrow points to a sharp spike in throughput at approximately 15:00:00.
- Time spent by span type:** A stacked bar chart showing the percentage of time spent on different span types. The y-axis ranges from 0% to 100%. The x-axis shows time from 14:33:20 to 15:06:40. A legend indicates 'app' (green).

At the bottom of the page, there are sections for 'Failed transaction rate' and 'Top 5 errors'.

# Service Details

- Observability
- Overview
- Alerts
- Cases
- Logs
- Stream
- Anomalies
- Categories
- Infrastructure
- Inventory
- Metrics Explorer
- APM
- Services**
- Traces
- Dependencies
- Service Map
- Uptime
- Monitors
- TLS Certificates
- User Experience


## frontend-node 🟢 ⚙️ Environment All ▾

- Overview
- Transactions
- Dependencies
- Errors
- Metrics
- Infrastructure 🔗
- Service Map
- Logs**
- Alerts 🔔

📅 Last 30 minutes ↻ Refresh

Timestamp	Message
15:27:09.772	EV421CA4NQ11
15:27:11.395	[frontend-node.log][info] /ads - fetching ads
15:27:12.589	[frontend-node.log][info] /cart - fetching cart for user: synthetic-new-york-967632
15:27:14.969	[frontend-node.log][info] /add-cart - adding product to cart
15:27:15.462	[frontend-node.log][info] /cart - fetching cart for user: synthetic-new-york-370944
15:27:15.745	[frontend-node.log][info] /recommendations - fetching recommendations
15:27:16.047	EV421BA1FQ11
15:27:16.157	PI922HA0IN11
15:27:16.534	PI922EA0IC11
15:27:16.644	AX821BA00Q11
15:27:16.751	EV421DA5ZG11
15:27:17.264	[frontend-node.log][info] /checkout - checking out
15:27:17.267	Error: 13 INTERNAL: failed to charge card: could not charge the card: rpc error: code = Code(401) desc = Your credit card (ending 0454) expired on 1/19
15:27:17.267	at Object.exports.createStatusError (/usr/src/frontend-gateway/node_modules/grpc/src/common.js:91:15)
15:27:17.268	[][access] 37.69.84.52 "POST /api/checkout? HTTP/1.0" 500
15:27:17.268	at Object.onReceiveStatus (/usr/src/frontend-gateway/node_modules/grpc/src/client_interceptors.js:1209:28)
15:27:17.268	at InterceptingListener._callNext (/usr/src/frontend-gateway/node_modules/grpc/src/client_interceptors.js:568:42)
15:27:17.268	at InterceptingListener.onReceiveStatus (/usr/src/frontend-gateway/node_modules/grpc/src/client_interceptors.js:618:8

# Service Details



Observability APM Services frontend-rum Transactions Click - button["Add To Cart"] Settings Anomaly detection Alerts and rules Add data

### Observability

- Overview
- Alerts
- Cases
- Logs
- Stream
- Anomalies
- Categories
- Infrastructure
  - Inventory
  - Metrics Explorer
- APM
  - Services**
  - Traces
  - Dependencies
  - Service Map
- Uptime
  - Monitors
  - TLS Certificates
- User Experience
  - Dashboard

**Trace samples** Latency correlations Failed transaction correlations

### Latency distribution

 182 total transactions  
Selection: 4,690 ms - 11 s  
95p  
Current sample

Trace sample 1 of 66 Investigate View full trace

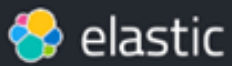
7 minutes ago 9,508 ms (100% of trace) http://frontend-ecom-nginx-9/products/women/Dress/AN621CA... Android (4.1)

Timeline Metadata Logs

Services frontend-rum frontend-node cartService

Service	Operation	Duration
frontend-rum	Click - button["Add To Cart"]	9,508 ms
frontend-rum	POST /api/add-cart	9,507 ms
frontend-node	HTTP 2xx POST /api/add-cart	9,394 ms

# Service Details

Find apps, content, and more. Ex: Discover⌕ ⚙️ 👤


☰ D Observability APM Services frontend-rum Transactions Click - button["Add To Cart"] Settings ⚠️ Anomaly detection Alerts and rules ➕ Add data

## Observability

- Overview
- Alerts
- Cases
- Logs
- Stream
- Anomalies
- Categories
- Infrastructure
  - Inventory
  - Metrics Explorer
- APM
  - Services**
  - Traces
  - Dependencies
  - Service Map
- Uptime
  - Monitors
  - TLS Certificates
- User Experience
  - Dashboard

**Trace samples** Latency correlations Failed transaction correlations

### Latency distribution ? | 186 total transactions Help



● All transactions ● client.geo.country\_iso\_code:US

### Correlations

Progress: 100%

Refresh

Correlation <span data-bbox="833 1547 882 1575">?</span> ↓	Impact	Field name	Field value	Filter
0.11	Very low	client.geo.country_iso_code	US	<span data-bbox="2865 1613 2915 1641">+</span> <span data-bbox="2932 1613 2982 1641">-</span>

Rows per page: 10 v

< 1 >



**감사합니다**

