Module

APM Fundamentals

elastic

Topics

- Introduction to Elastic APM
- APM Server
- APM Agents
- Learn More

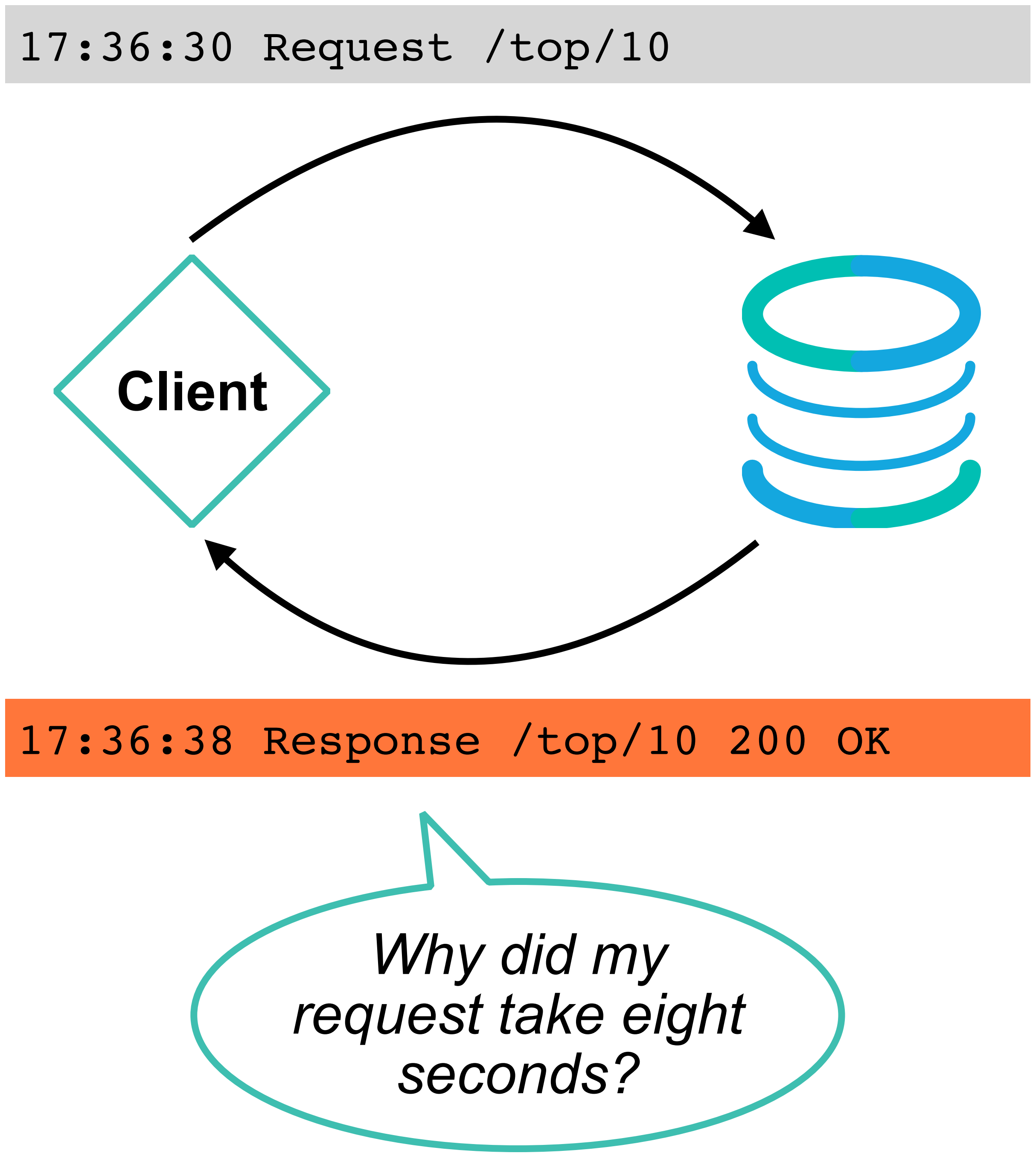
Lesson 1

Introduction to Elastic APM

elastic

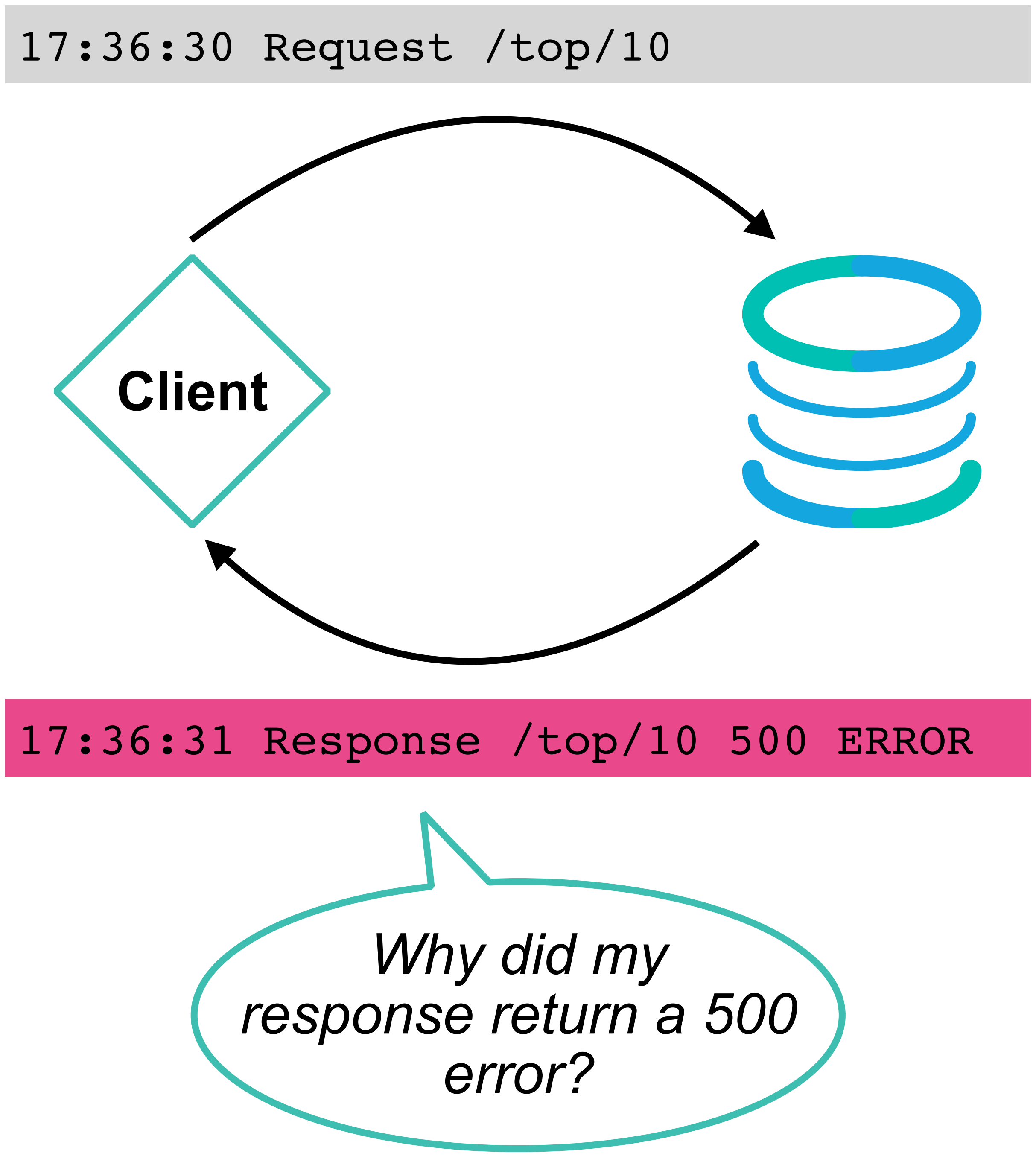
What is APM?

- Application Performance Monitoring



```
17:36:30 Request /top/10
```

```
17:36:38 Response /top/10 200 OK
```



```
17:36:30 Request /top/10
```

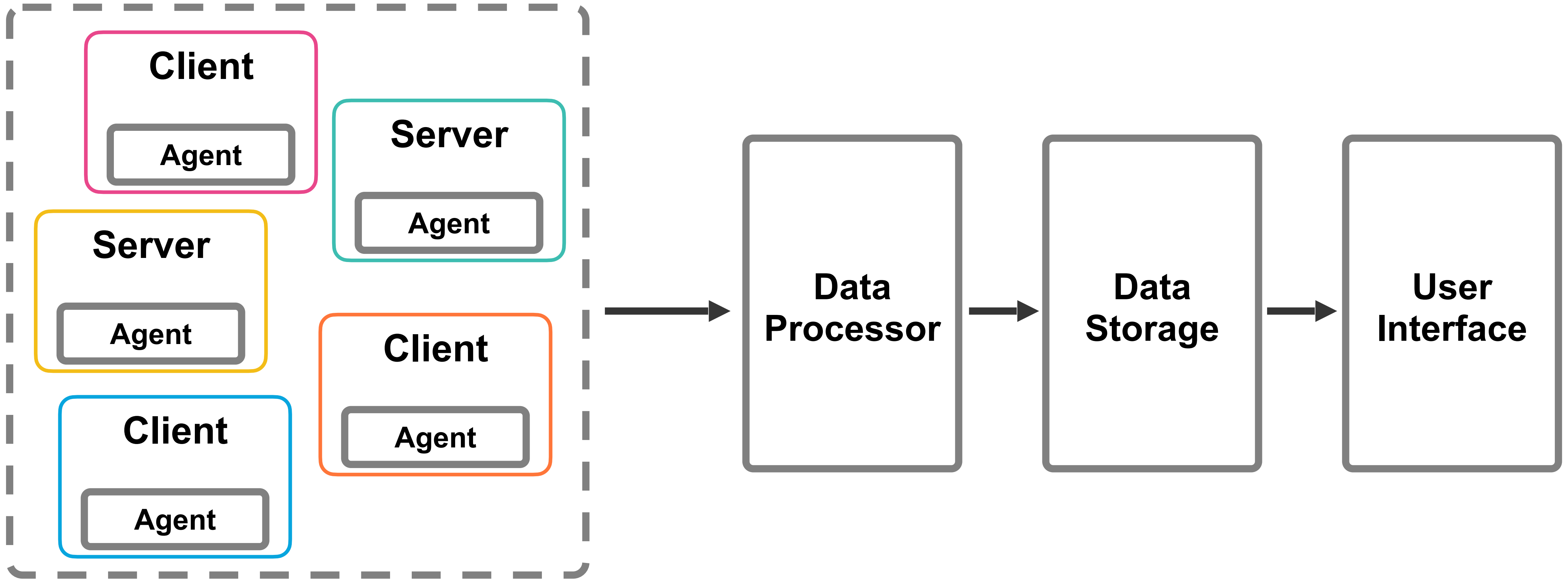
```
17:36:31 Response /top/10 500 ERROR
```

Why APM?

- It records traces for database queries, external HTTP requests, and other slow operations that happen during requests to your application
  - it is easier to see where your application is spending time
- It also collects errors and exceptions that are not handled by your application
  - it is easier to debug errors that might happen in your application
- You can find performance bottlenecks and errors before your customers face them
- You can increase the productivity of your development team

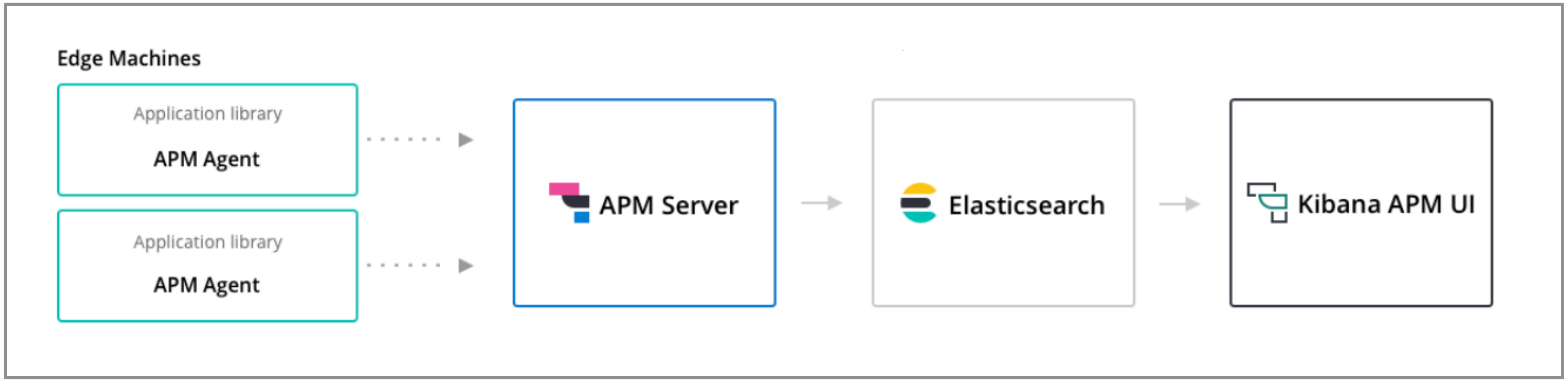
How APM Works?

- This is how Application Performance Monitoring works in general

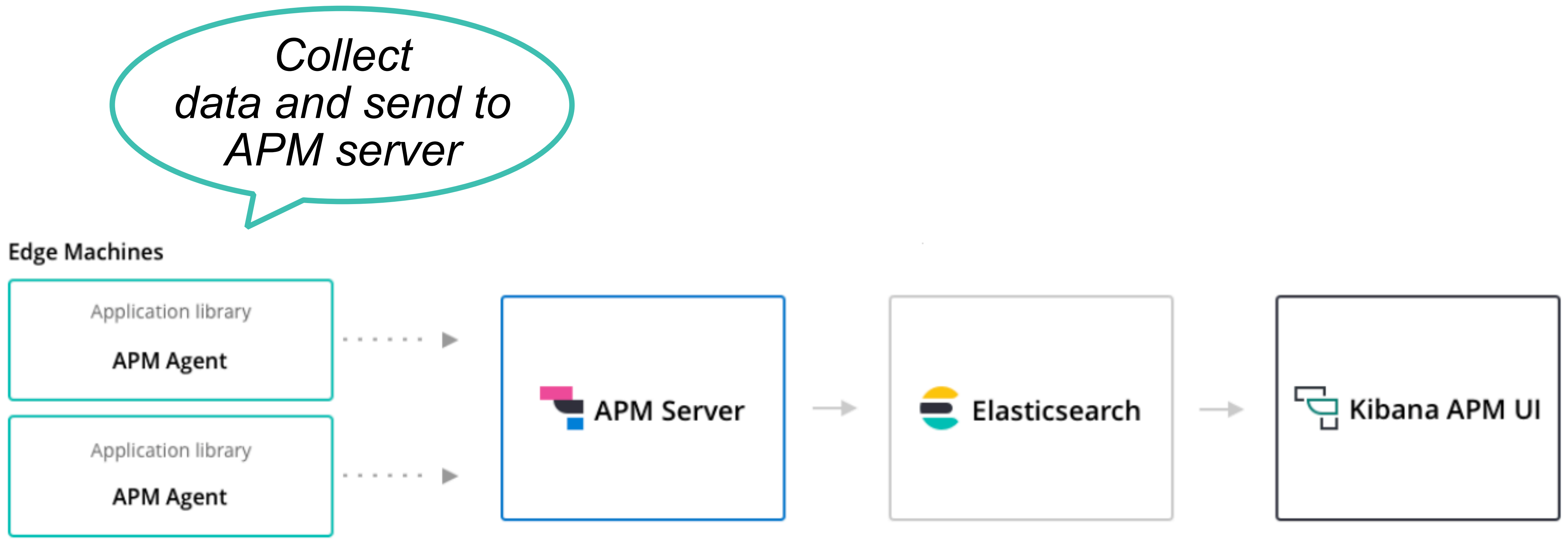


Elastic APM Components

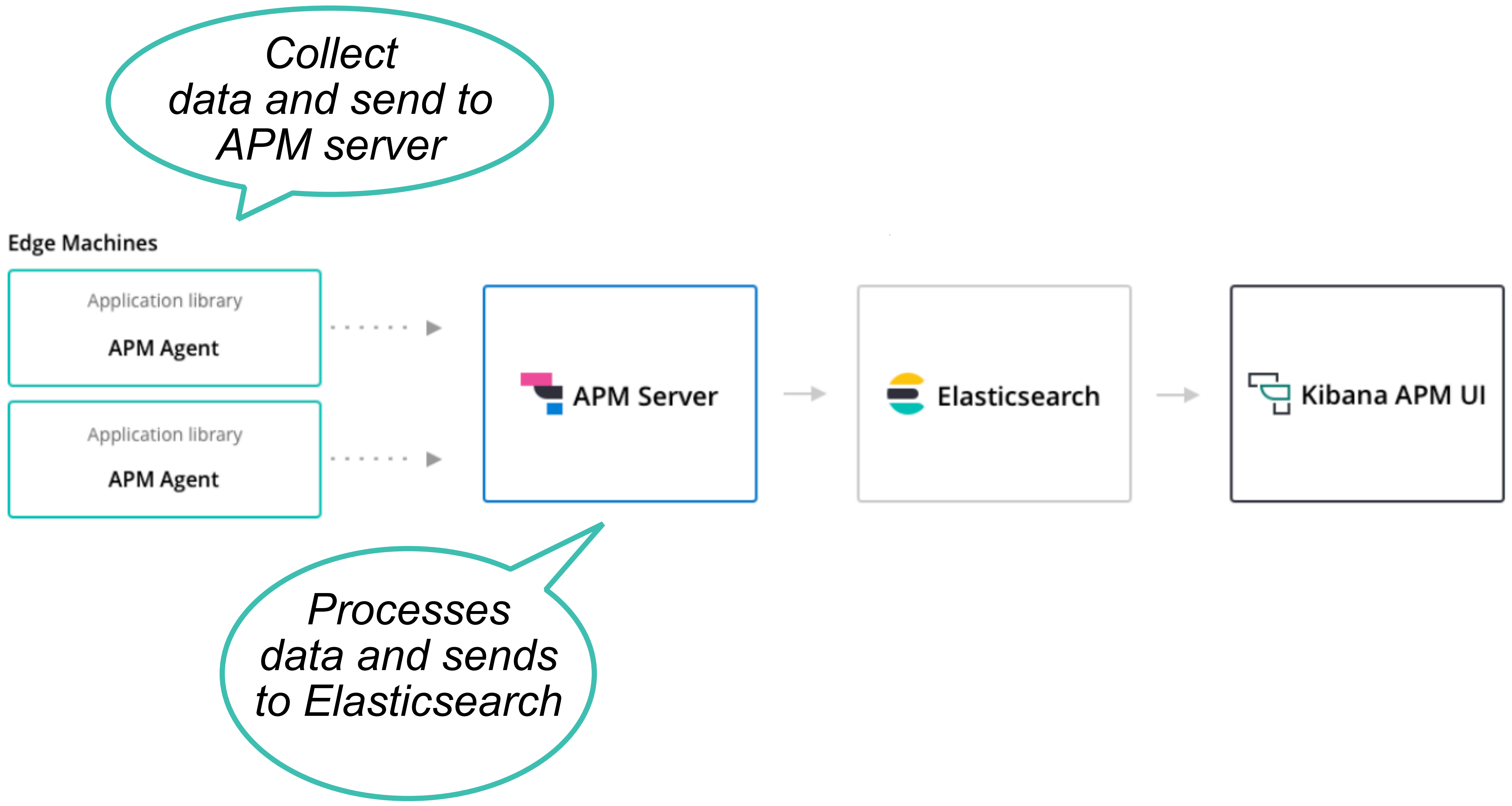
- Elastic APM consists of four components
  - APM Agents
  - APM Server
  - Elasticsearch
  - Kibana APM UI



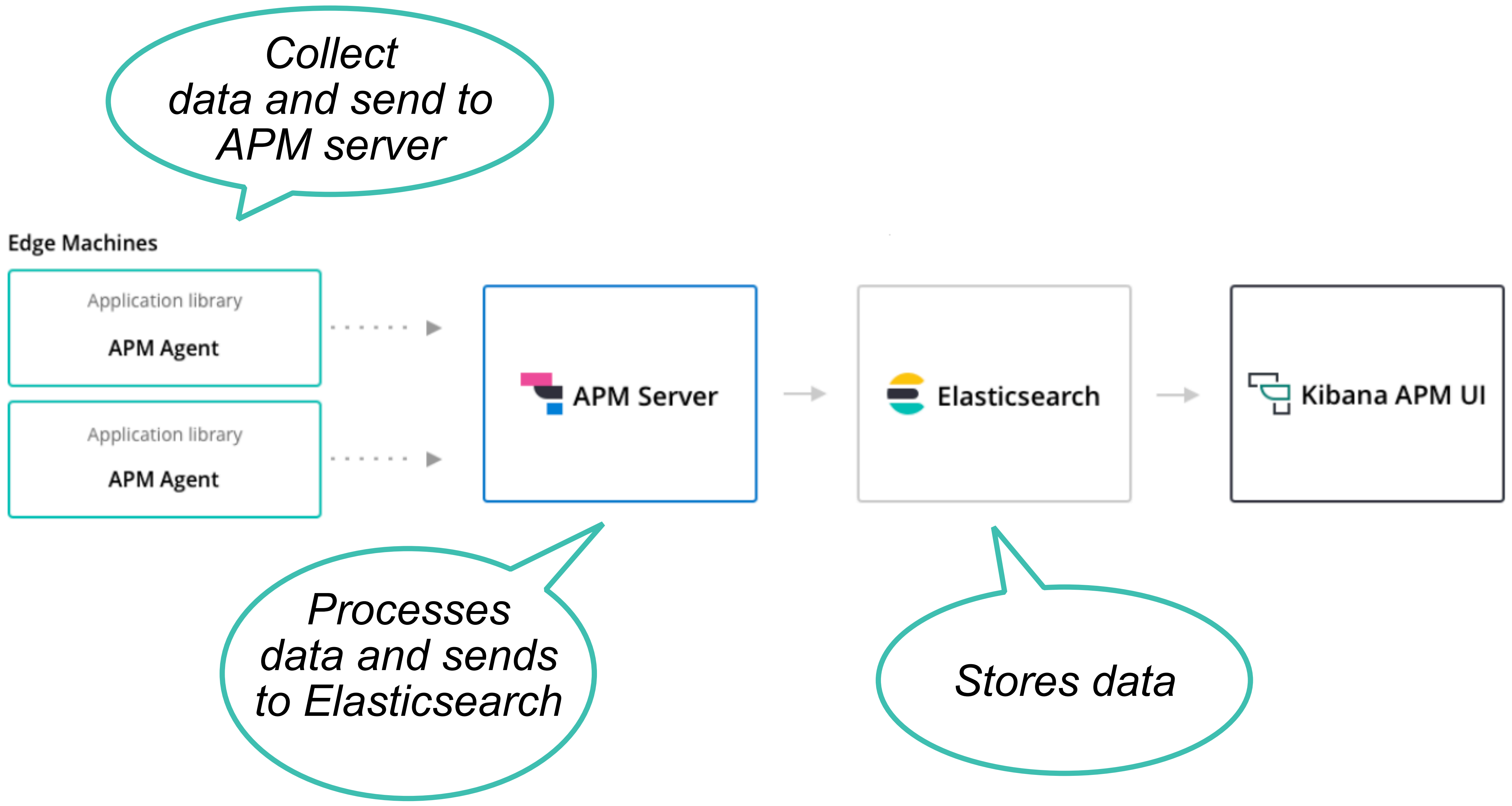
APM Agents


Collect data and send to APM server
Edge Machines
Application library
APM Agent
Application library
APM Agent
APM Server
Elasticsearch
Kibana APM UI

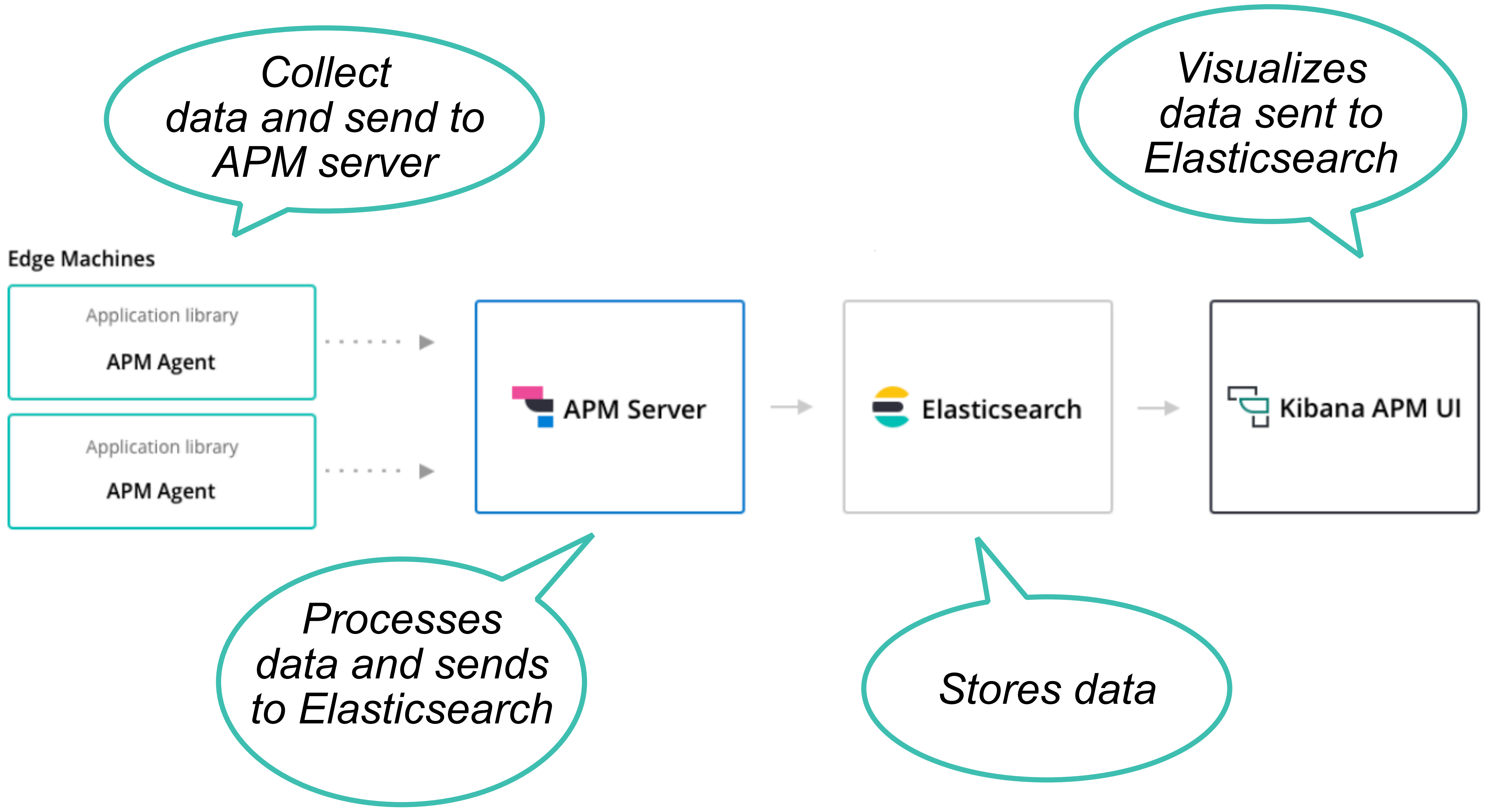
APM Server


Collect data and send to APM server
Edge Machines
Application library
APM Agent
Application library
APM Agent
APM Server
Elasticsearch
Kibana APM UI
Processes data and sends to Elasticsearch

Elasticsearch



Kibana APM UI



Logs x Metrics x APM

- Full-stack monitoring with Elastic APM
- Adds end-user experience and application-level monitoring to the Elastic Stack

APM	Real User Monitoring (RUM)
APM	Application-level monitoring
Beats	Server-level monitoring
Beats Logstash	Logging

Getting Started with Elastic APM

1. Start an Elasticsearch cluster with Kibana (version > 5.6)
2. Start APM Server
3. Setup APM agents in your applications
4. Verify that the data is in Elasticsearch

APM Fundamentals

Lesson 1

Review - Introduction to Elastic APM

elastic

Summary

- Elastic APM allows you to monitor software services and applications in real time, collecting detailed performance information on response time for incoming requests, database queries, calls to caches, external HTTP requests, etc
- Elastic APM also automatically collects unhandled errors and exceptions
- Elastic APM consists of four components: APM agents, APM Server, Elasticsearch, and Kibana APM UI

Quiz

1. **True** or **False**: Elastic APM helps monitor your application by collecting detailed performance information about requests in your application.
2. **True** or **False**: Elastic APM helps debug your application by collecting unhandled errors and exceptions.
3. What are the four components of Elastic APM?

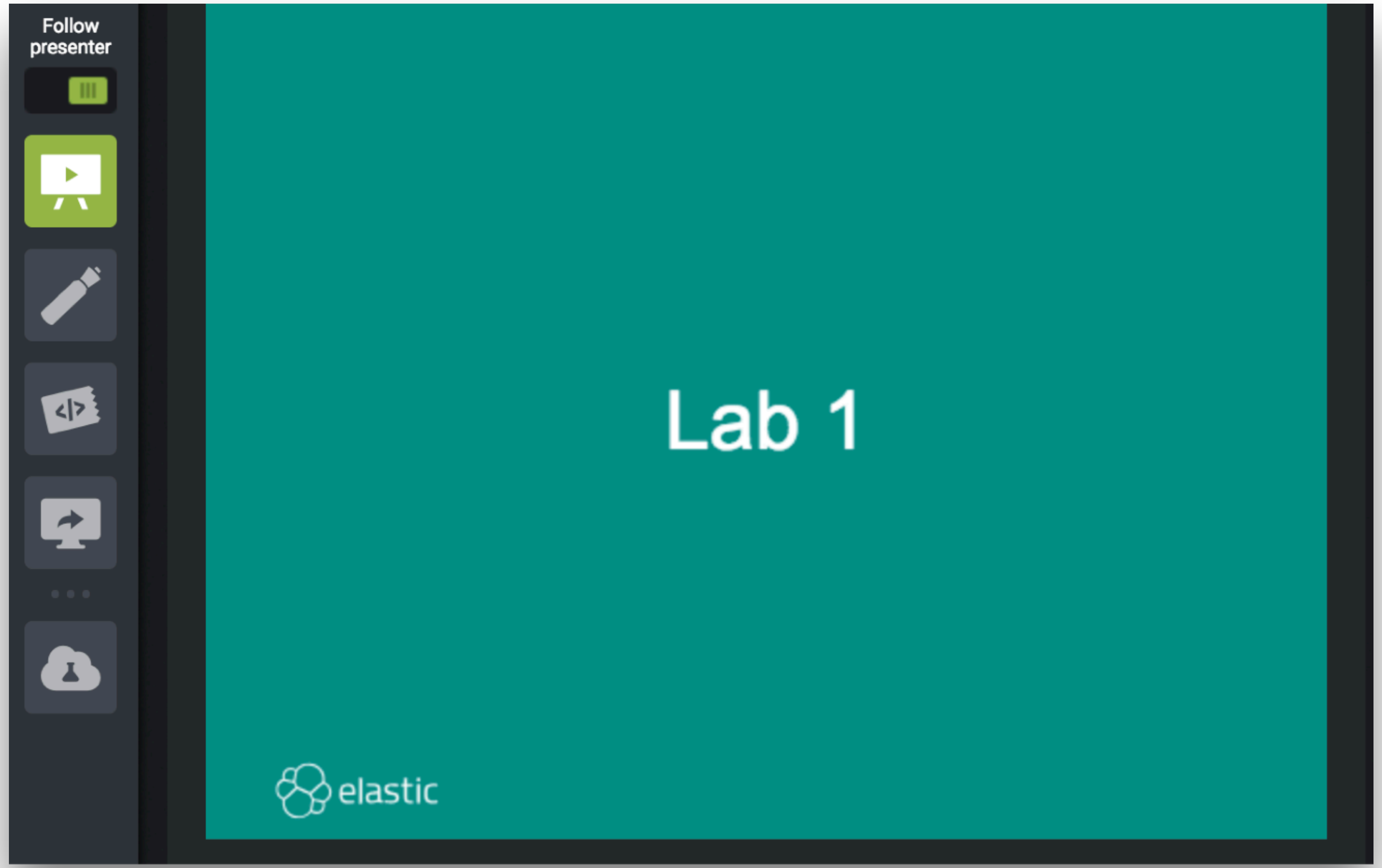
Lesson 1

Lab - Introduction to Elastic APM

elastic

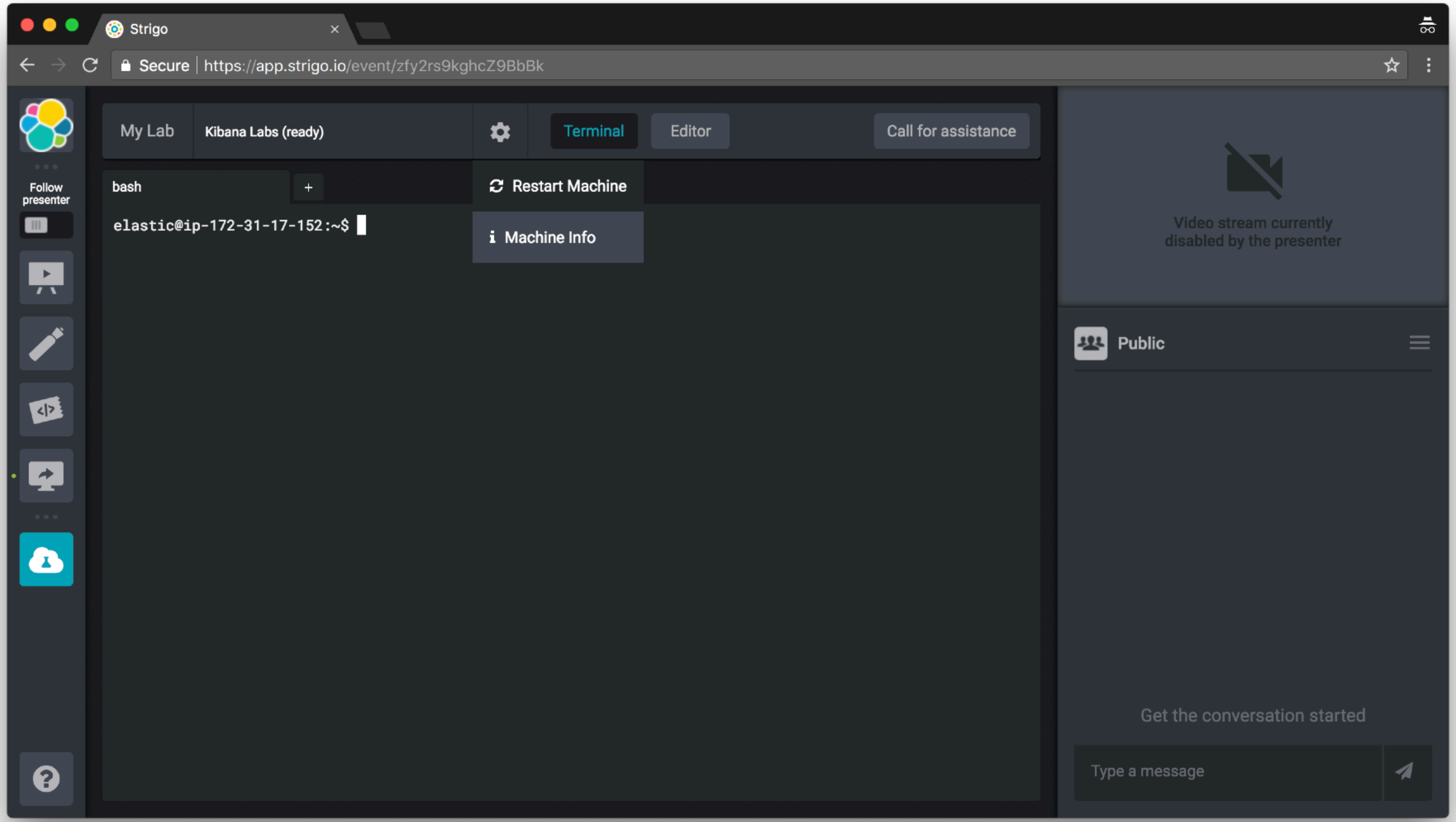
Lab Environment

- Visit Strigo using the link that was shared with you, and log in if you haven't already done so
- Click on "**My Lab**" on the left



Lab Environment

- Click on the gear icon next to "**My Lab**" and select "**Machine Info**"

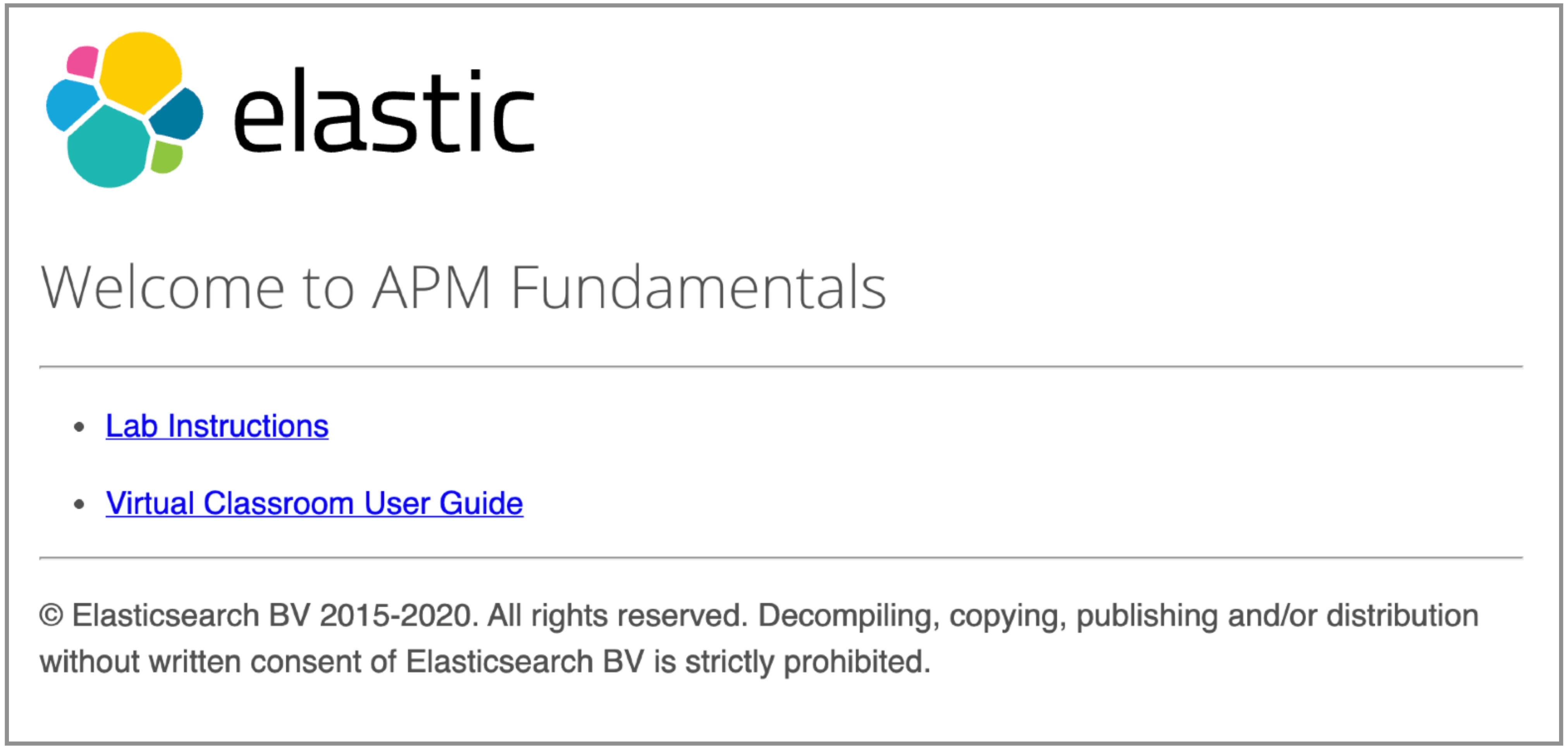


```
elastic@ip-172-31-17-152:~$
```

elastic

Lab Environment

- From here you can access lab instructions and guides
  - You also have them in your .zip file, but it is easier to access and use the lab instructions from here:



elastic

Welcome to APM Fundamentals

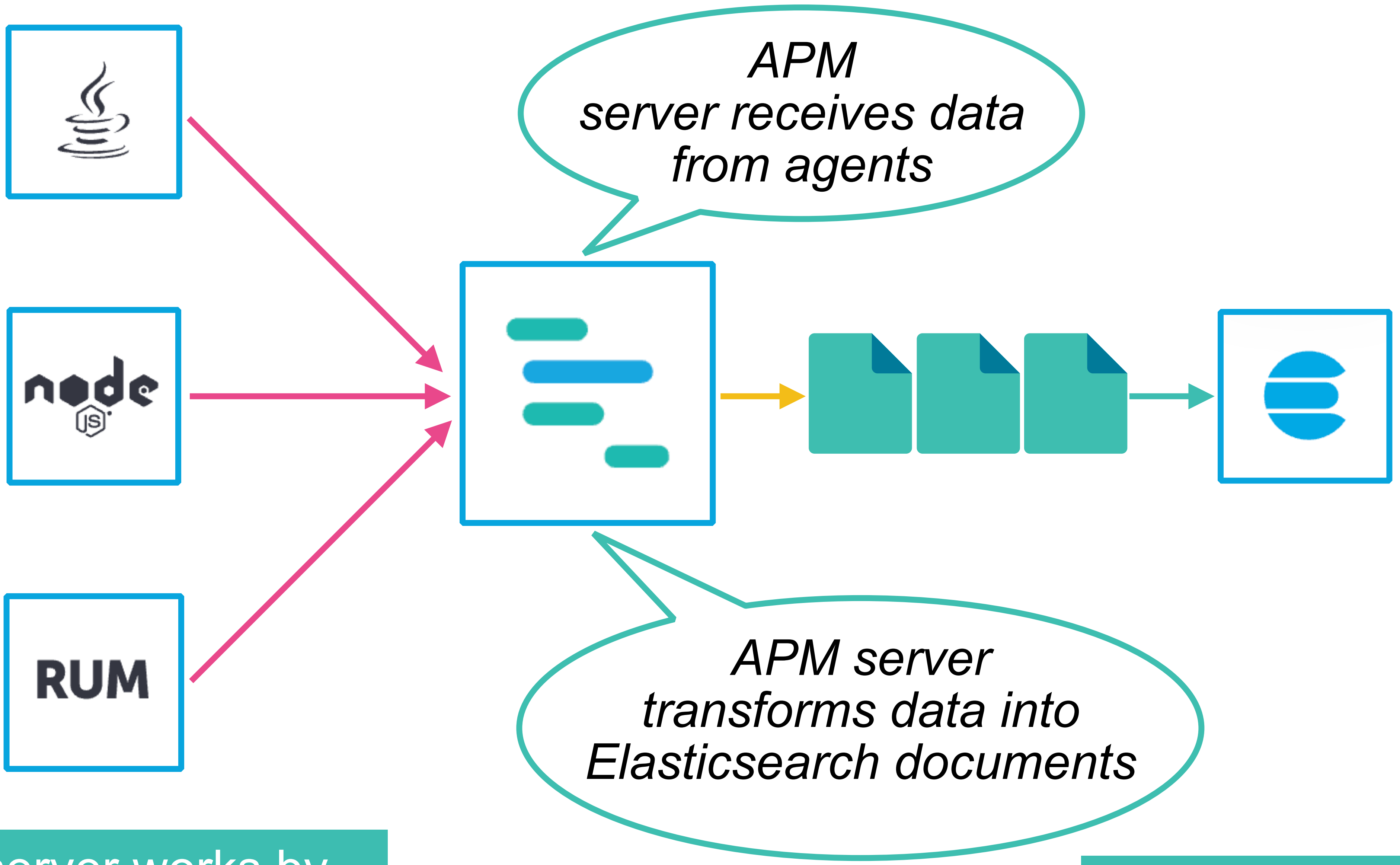
- Lab Instructions
- Virtual Classroom User Guide

© Elasticsearch BV 2015-2020. All rights reserved. Decompiling, copying, publishing and/or distribution without written consent of Elasticsearch BV is strictly prohibited.

Lesson 2

APM Server

APM Server Overview


RUM
APM server receives data from agents
APM server transforms data into Elasticsearch documents

APM server works by exposing an HTTP server to which agents ship the APM data they collect

APM server is built with the Beats framework, and as such it leverages its functionality

Why is APM Server a Separate Component?

- Performance
  - it can be scaled independently since it is a stateless component
  - it controls the amount of data flowing into Elasticsearch
  - it can buffer data when Elasticsearch becomes unresponsive
- Security
  - it prevents browsers from interacting with Elasticsearch
- Interoperability
  - it acts as a middleware for JavaScript source mapping
  - it provides a JSON API for agents

Data Model

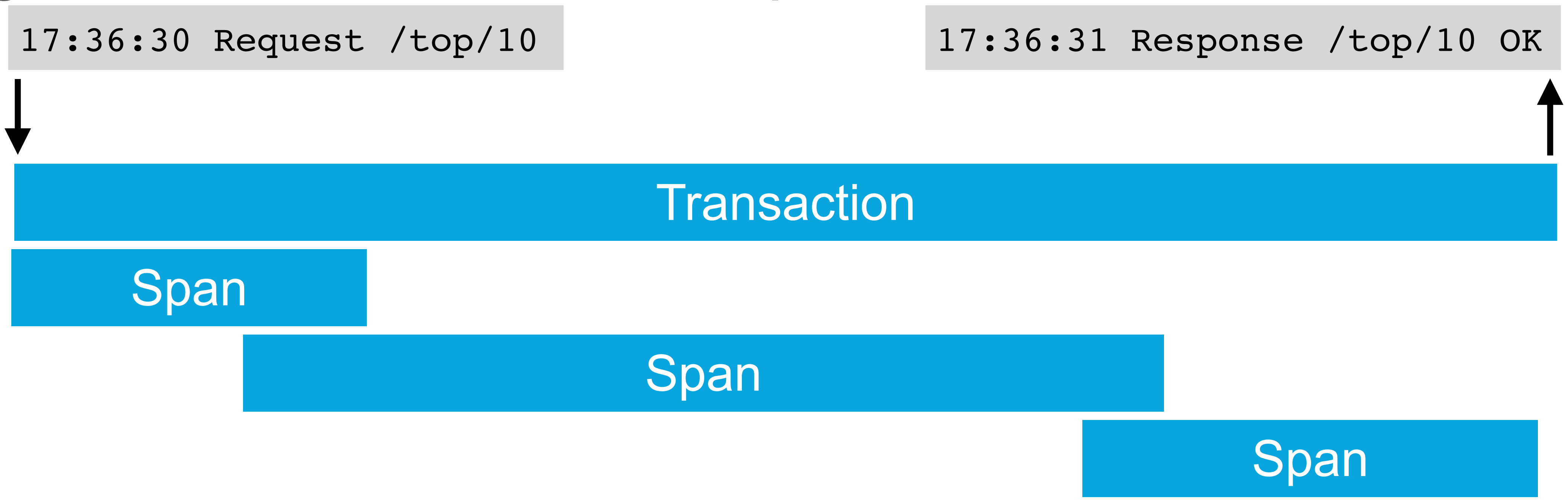
- What kind of data agents send to the APM server?
  - spans, transactions, errors or metrics
  - which are known as events
  - and can contain additional metadata
  - such as labels, custom context and user context

Spans

- Spans contain information about a specific code path that has been executed
- They measure from the start to end of an activity
  - and they can have a parent/child relationship with other spans
- A span contains
  - a transaction id attribute that refers to their parent transaction
  - a parent id attribute that refers to their parent span or transaction
  - start time and duration
  - name
  - type
  - stack trace (optional)

Transactions

- Transactions are a special kind of span that have additional attributes associated with them
- They describe an event captured by an Elastic APM agent instrumenting an application
  - e.g., a HTTP request or an asynchronous background job
  - it includes the timestamp of the event, a unique id, type, name, data about the environment and other relevant information
- Together, transactions and spans form a trace



elastic

Errors

- An error is either a captured **exception** or a captured **log**
  - it can contain a **stack trace**, which is helpful for debugging
  - the **culprit** of the error indicating where it originated
  - and might relate to the transaction during which it happened
- For simplicity, errors are represented by a unique ID

```
server/top.js in <anonymous> at line 27

25.
26. app.get('/top/10', function (req, res) {
27. apm.captureError('this is a string', function (err) {
28. if (err) {
29.   res.status(500).send('could not capture error: ' + err.message)
```

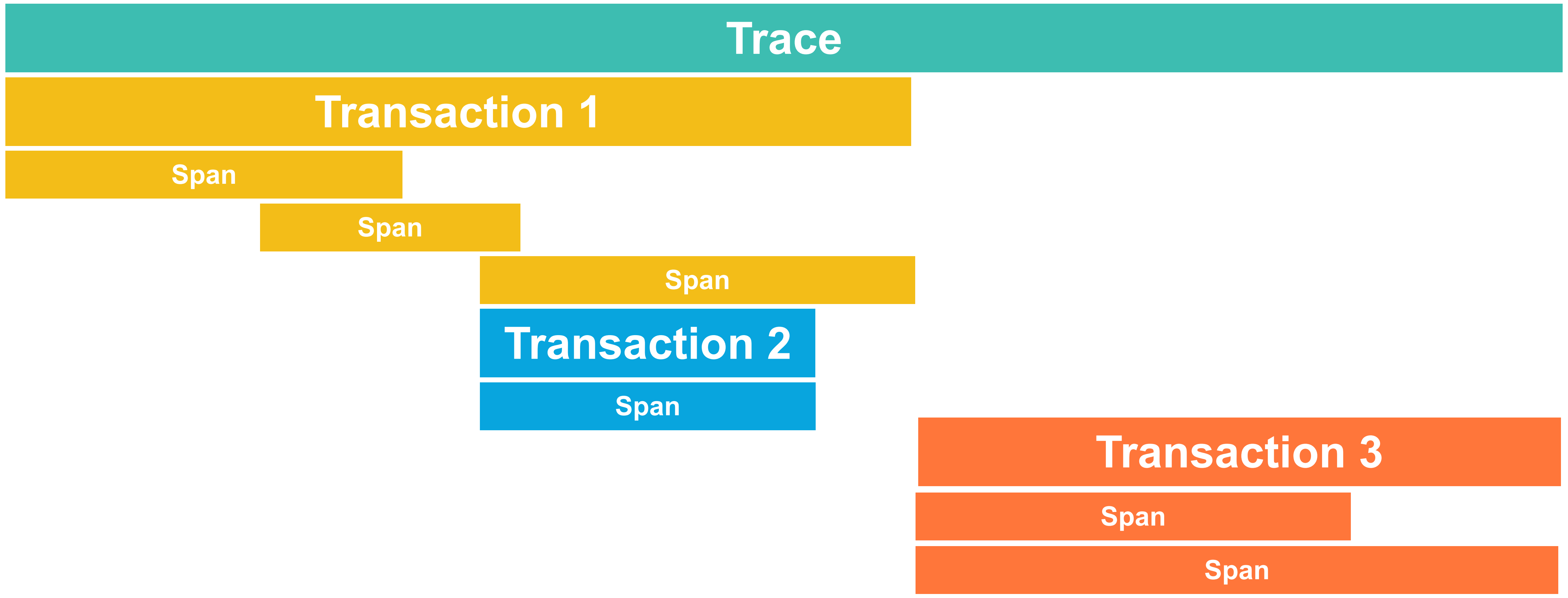
```
server.js in <anonymous> at line 75

73. })
74.
75.  next()
76. })
77.
```

Metrics

- APM agents automatically pick up basic host-level metrics
  - including system and process-level CPU and memory metrics
- Agents specific metrics are also available
  - like JVM metrics in the Java agent
  - and Go runtime metrics in the Go agent

Distributed Tracing

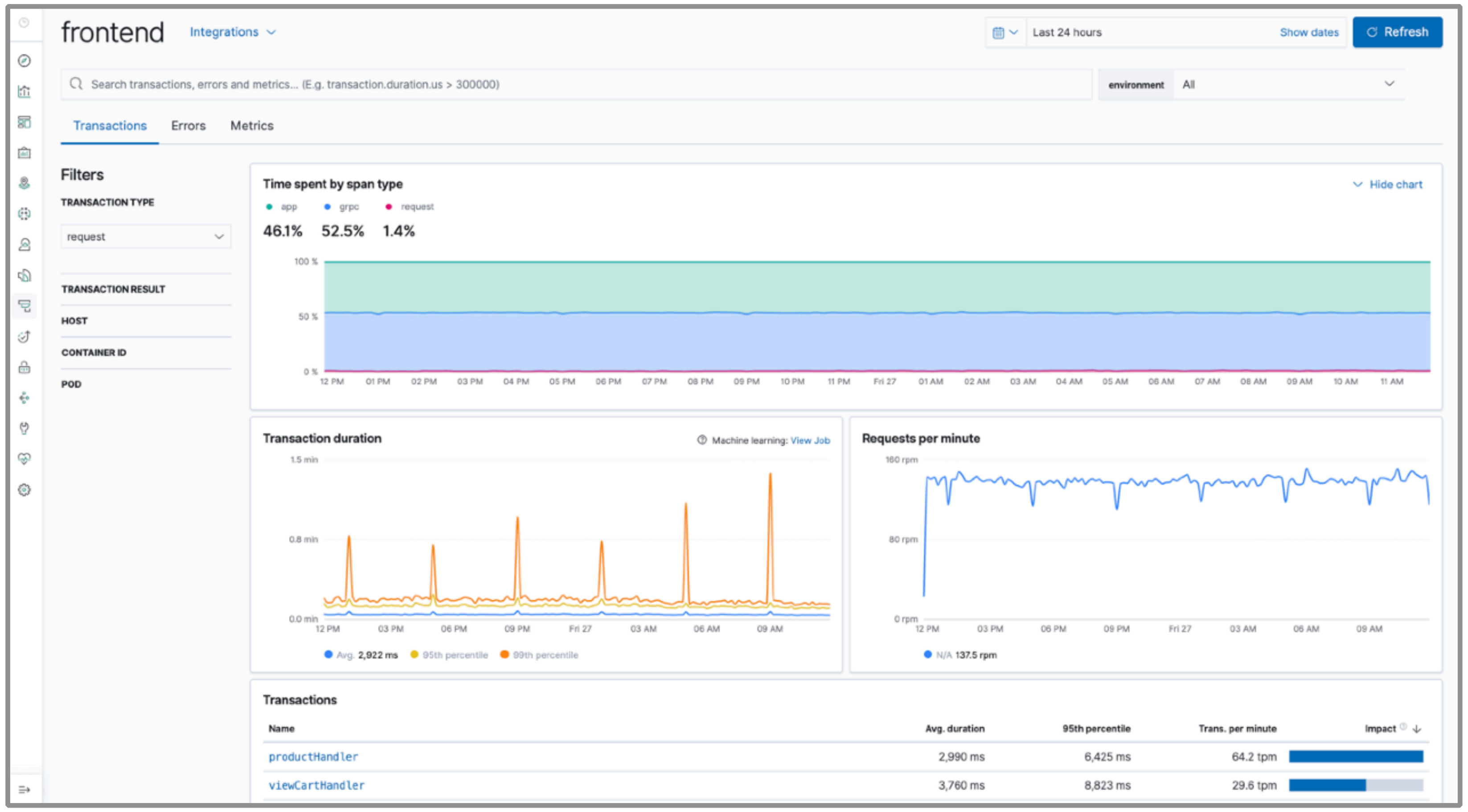
- One trace that spans multiple transactions
- APM does that by giving each trace a unique ID
  - it uses an unique trace ID
  - and passes that ID on through the transactions
  - thereby creating an end-to-end trace for multiple applications



elastic

Visualizing APM Data

- All these data are stored in different Elasticsearch indices
  - `apm-[version]-span-00000X`
  - `apm-[version]-transaction-00000X`
  - `apm-[version]-error-00000X`
  - `apm-[version]-metric-00000X`



Transactions

Name	Avg. duration	95th percentile	Trans. per minute	Impact
productHandler	2,990 ms	6,425 ms	64.2 tpm	
viewCartHandler	3,760 ms	8,823 ms	29.6 tpm	

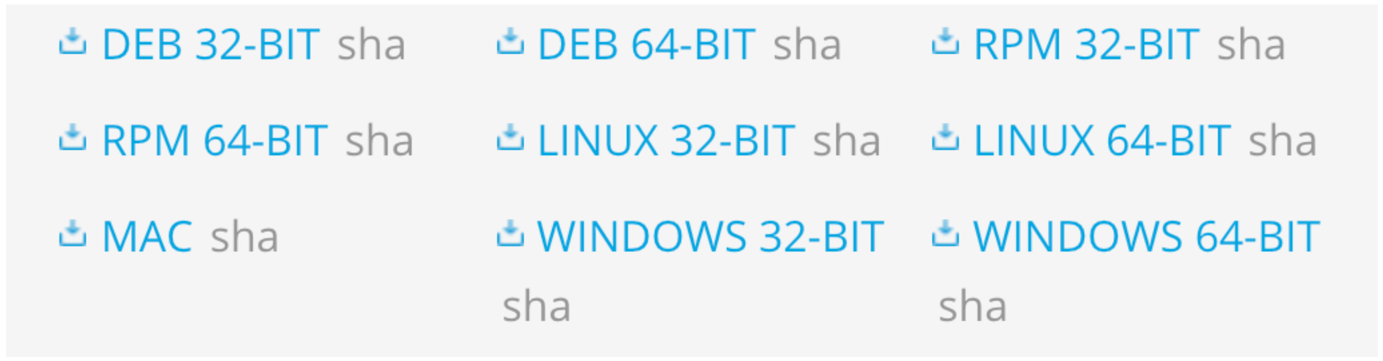
Visualize the data after it is indexed in Elasticsearch with the dedicated APM UI

Install and Run the APM Server

elastic

Download APM Server

- Download the APM Server for your operating system
  - https://www.elastic.co/downloads/apm



DEB 32-BIT sha	DEB 64-BIT sha	RPM 32-BIT sha
RPM 64-BIT sha	LINUX 32-BIT sha	LINUX 64-BIT sha
MAC sha	WINDOWS 32-BIT sha	WINDOWS 64-BIT sha

APM, YUM and homebrew repositories are also available

- APM Server is also available on Docker
  - https://www.elastic.co/guide/en/apm/server/current/running-on-docker.html
- The easiest way to get started is by using our hosted Elasticsearch Service on Elastic Cloud
  - https://www.elastic.co/guide/en/apm/get-started/current/install-and-run.html

Install APM Server

- MacOS

```
tar xzvf apm-server-7.6.1-darwin-x86_64.tar.gz
```

- Linux

```
tar xzvf apm-server-7.6.1-linux-x86_64.tar.gz
```

- DEB

```
sudo dpkg -i apm-server-7.6.1-amd64.deb
```

- RPM

```
sudo rpm -vi apm-server-7.6.1-x86_64.rpm
```

- Windows

1. Extract the contents of the zip file into **`C:\Program Files`**
2. Rename the **`apm-server-7.6.1-windows`** directory to **`APM-Server`**
3. Open a PowerShell prompt as an Administrator

```
PS > cd 'C:\Program Files\APM-Server'
PS C:\Program Files\APM-Server> .\install-service-apm-server.ps1
```

elastic

Configure APM Server

- Edit the **`apm-server.yml`** configuration file
  - MacOs/Linux/Windows: look in the archive that you extracted
  - DEB/APT/RPM/YUM: **`/etc/apm-server/apm-server.yml`**

```
apm-server:
  host: "localhost:8200"

output:
  elasticsearch:
    hosts: "localhost:9200"
```

- You must specify credentials if you have Elastic security
- Besides Elasticsearch, other outputs are also available
  - Logstash, Kafka, File, Console, and Elastic Cloud
- When installing agents on your applications also check agent and server compatibility

Start APM Server

- MacOS

```
cd apm-server-7.6.1-darwin-x86_64/
./apm-server -e
```

- Linux

```
cd apm-server-7.6.1-linux-x86_64/
./apm-server -e
```

- DEB/APT

```
sudo service apm-server start
```

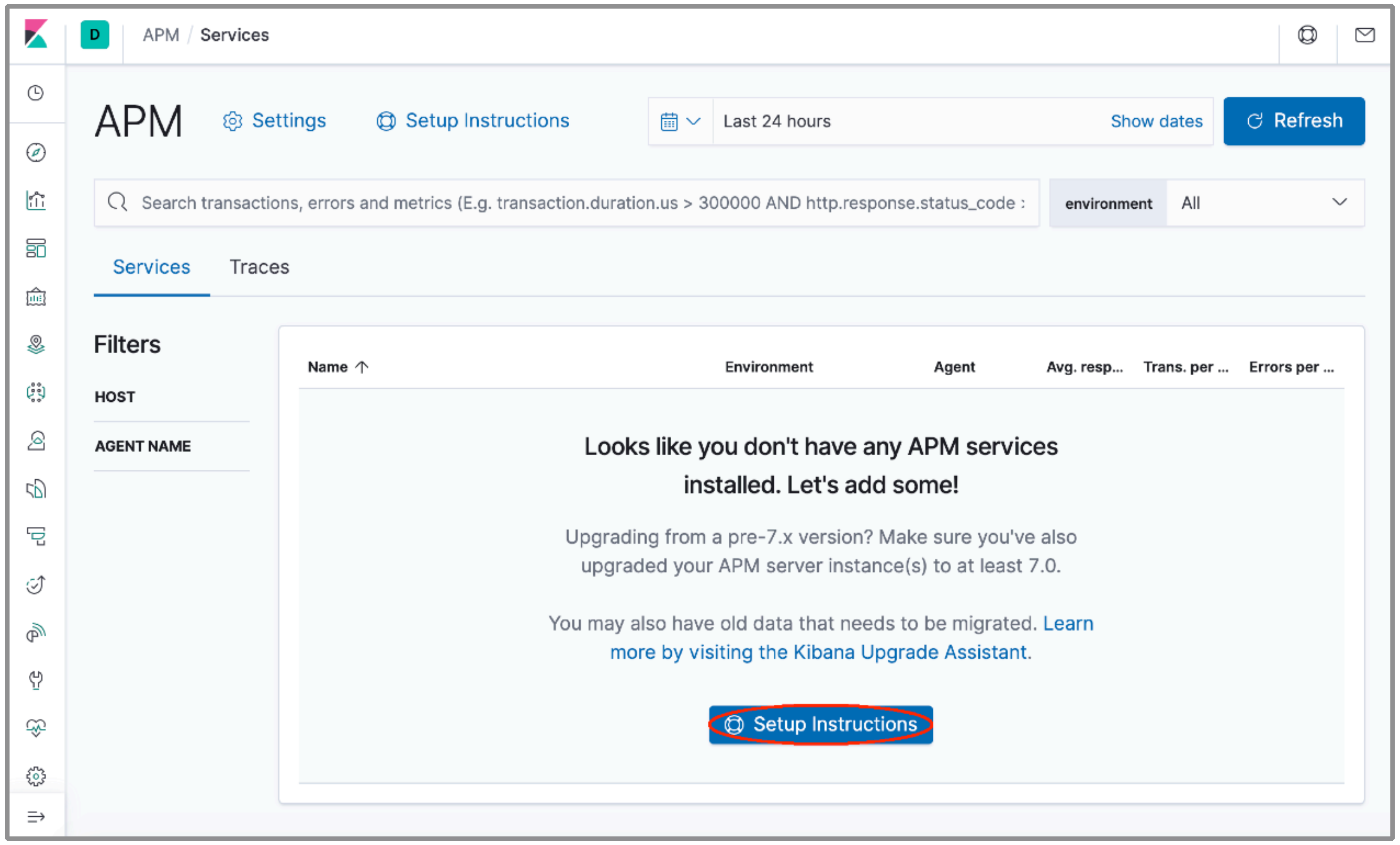
- RPM/YUM

```
sudo systemctl start apm-server
```

- Windows

```
PS C:\Program Files\APM-Server> Start-Service apm-server
```

Load Kibana Objects



APM / Services

APM Settings Setup Instructions

Last 24 hours Show dates Refresh

Search transactions, errors and metrics (E.g. transaction.duration.us > 300000 AND http.response.status_code :

environment All

Services Traces

Filters

HOST

AGENT NAME

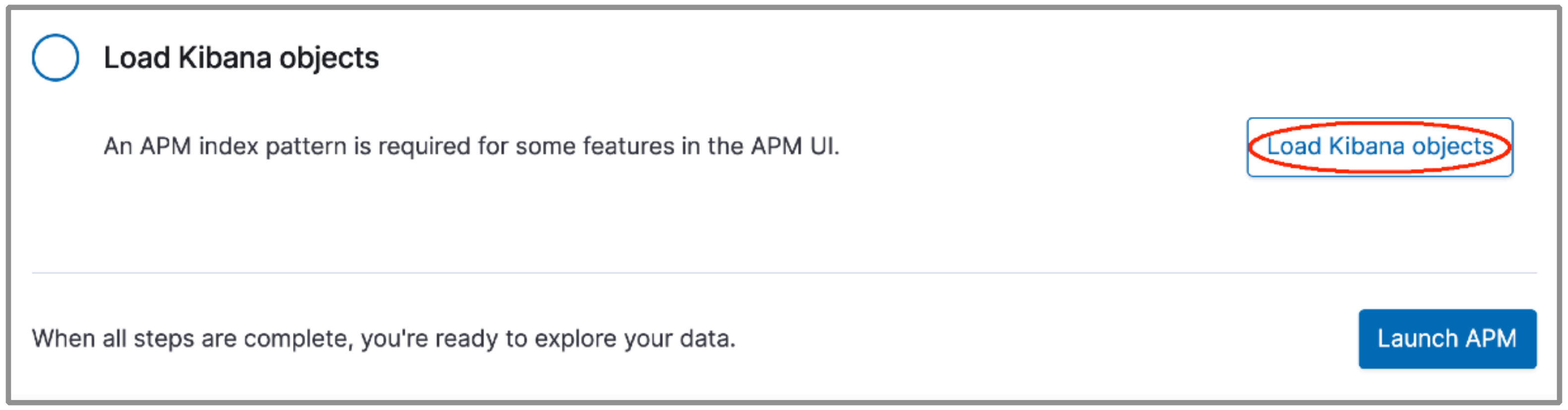
Name	Environment	Agent	Avg. resp...	Trans. per ...	Errors per ...

Looks like you don't have any APM services installed. Let's add some!

Upgrading from a pre-7.x version? Make sure you've also upgraded your APM server instance(s) to at least 7.0.

You may also have old data that needs to be migrated. Learn more by visiting the Kibana Upgrade Assistant.

Setup Instructions



Load Kibana objects

An APM index pattern is required for some features in the APM UI.

Load Kibana objects

When all steps are complete, you're ready to explore your data.

Launch APM

Stop APM Server

- MacOS

```
Ctrl + C
```

- Linux

```
Ctrl + C
```

- DEB/APT

```
sudo service apm-server stop
```

- RPM/YUM

```
sudo systemctl stop apm-server
```

- Windows

```
PS C:\Program Files\APM-Server> Stop-Service apm-server
```

Lesson 2

Review - APM Server

Summary

- The APM Server receives data from APM agents and transforms them into Elasticsearch documents
- The APM Server classifies the data collected by agents into spans, transactions, errors, and metrics, which can be visualized through the Kibana APM UI
- The APM Server is a separate component because it helps with scalability, controlling and buffering data that is sent to Elasticsearch, preventing browsers from interacting directly with Elasticsearch, and improving compatibility across agents and the Elastic Stack

Quiz

1. **True** or **False**: The APM Server transforms data collected by agents and ships them to Elasticsearch.
2. Which tool can be used to visualize the APM data stored in Elasticsearch?
3. **True** or **False**: The APM server improves the APM architecture because it adds resiliency to the APM architecture and allows fine-grained control over the amount of data flowing into Elasticsearch.

elastic

Lesson 2

Lab - APM Server

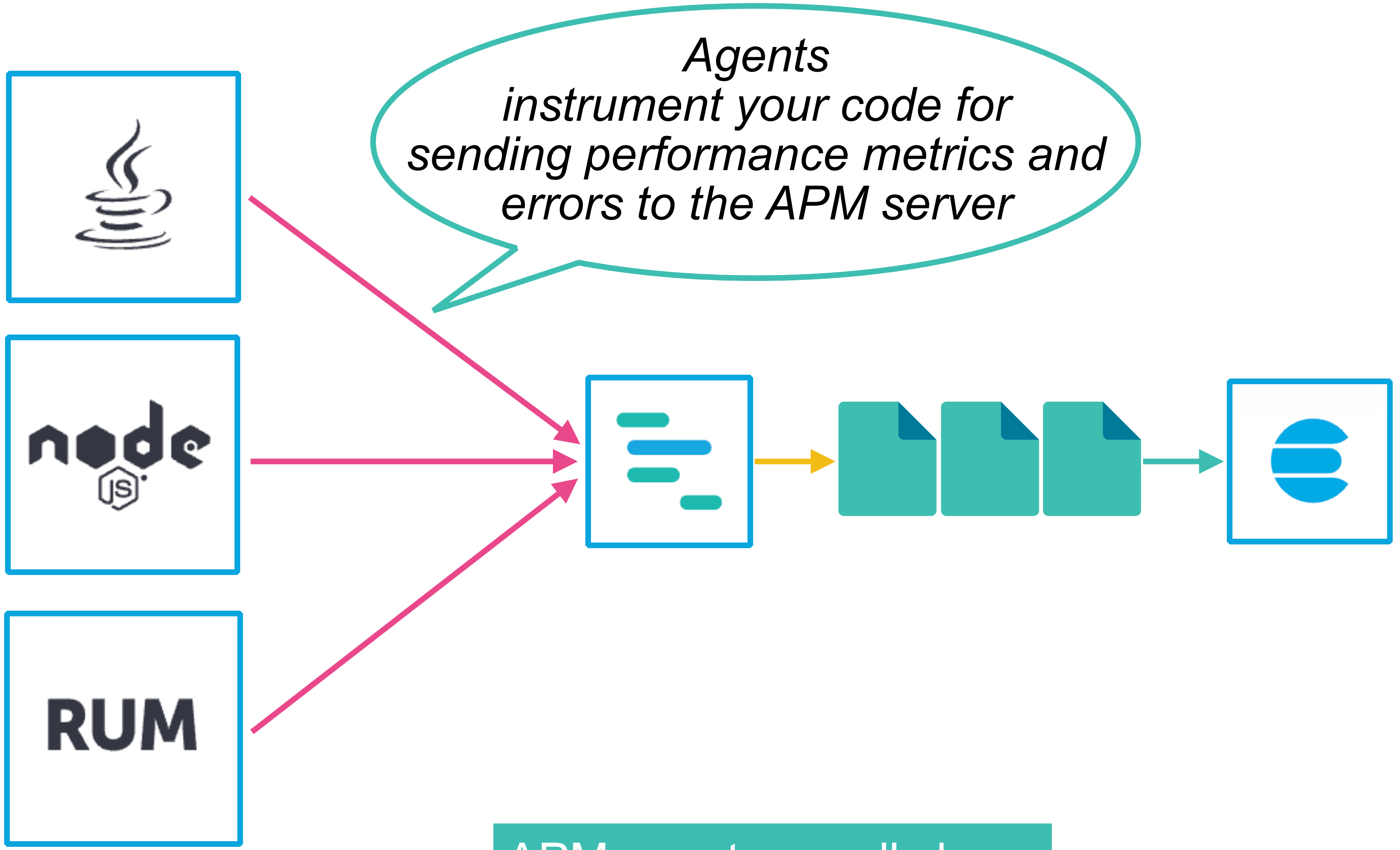
elastic

Lesson 3

APM Agents

elastic

APM Agents Overview


Agents instrument your code for sending performance metrics and errors to the APM server
node JS
RUM

APM agents are written in the same language as your service

APM agents usually have automatic instrumentation for the most popular frameworks and routers

APM agents also allow you to manually instrument your application

Available Agents

node JS

RUM

GO

.NET

These are the Elastic agents, but there are community agents too.

Choose a Service Name in your Application

- The **service.name** setting is configured in the agent
  - it is used to group data coming from different services
- It can only contain alphanumeric characters, spaces, underscores, and dashes

```
^[a-zA-Z0-9 _-]+$
```

- Elastic APM includes the **service.name** field on every document that it saves in Elasticsearch
  - if you change the it you will see two services
  - make sure you choose a good name before getting started
  - e.g., Training Search Page, training-search-page, training_search_page, ...

Java Agent

Install the Java Agent

- The first step is downloading the latest release of the Java agent jar file from maven central
  - don't declare a dependency to the agent in your application

Configure the Java Agent

- Add the JVM flag **-javaagent** when starting your application and set:
  - service name
  - custom APM Server URL (default **http://localhost:8200**)
  - the base package of your application

```
java -javaagent:/path/to/elastic-apm-agent-<version>.jar \
     -Delastic.apm.service_name=my-application \
     -Delastic.apm.server_url=http://localhost:8200 \
     -Delastic.apm.application_packages=org.example \
     -jar my-application.jar
```

- After starting your application, the Java agent will automatically collect measures on its usage and also some error reporting

elastic

Supported Technologies

- Java versions
  - Oracle JDK, Open JDK, and IBM J9 VM
- Web Frameworks
  - Servlet API, Spring Web MVC, JavaServer, Spring Boot, JAX-RS, and JAX-WS
- Application Servers/Servlet Containers
  - Tomcat, WildFly, JBoss EAP, Jetty, WebSphere Liberty, Undertow Servlet, Payara, and Oracle WebLogic
- Many other APIs and always more support to come
  - data stores, networking frameworks, and much more...
  - https://www.elastic.co/guide/en/apm/agent/java/current/supported-technologies-details.html

Node.js Agent

elastic

Install the Node.js Agent

- Install the Node.js agent as a dependency to your application

```
npm install elastic-apm-node --save
```

Configure the Node.js Agent

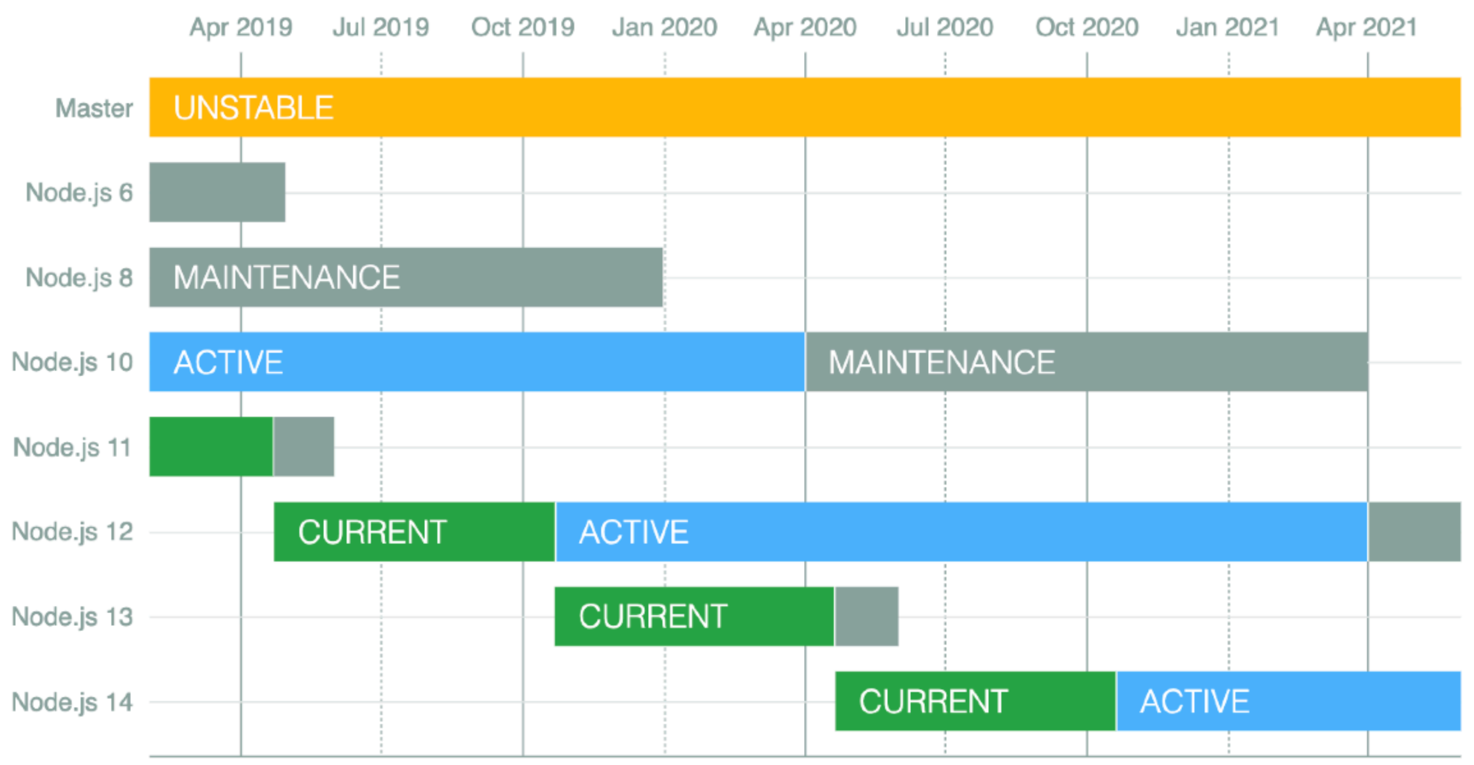
- Configure the Node.js agent to start before any other modules in your application

```
// Add this to the VERY top of the first file loaded in your app
const apm = require('elastic-apm-node').start({
  serviceName: 'my-application',
  serverUrl: 'http://localhost:8200',
  environment: 'production'
})
```

- After starting your application, the Node.js agent will automatically collect measures on its usage and also some error reporting

Supported Technologies

- The agent follows the support schedule of Node.js itself



- Web Frameworks
  - Express, hapi, Koa, Restify, and Fustily
- Many other modules and always more support to come
  - https://www.elastic.co/guide/en/apm/agent/nodejs/current/supported-technologies.html

elastic

RUM Agent

Real User Monitoring (RUM)

- It captures user **interaction with clients** such as web browsers
  - the JavaScript agent is the RUM agent
  - the RUM endpoint needs to be enabled in the APM server
- It can provide **valuable insights** into your applications
  - performance issues within your client-side application
  - latency of your server-side application
  - identify most used browsers, devices and platforms
  - regional performance of your website based on location information and individual user performance

Enable the RUM Endpoint in the APM Server

1. Enable the RUM endpoint in your **apm-server.yml**

```
apm-server.rum.enabled: true
```

2. Set the APM Server host

```
apm-server:
  host: "172.31.38.42:8200"
```

   – you can also use 0.0.0.0, the host name, or ${HOSTNAME}

3. Make sure that the server is open to the Internet

Install and Configure the RUM Agent

- Install the RUM agent as a dependency to your application
  - **npm** manages the agent version

```
npm install @elastic/apm-rum --save
```

- Configure the agent
  - load the RUM agent library and initialize the service

```
import { init as initApm } from '@elastic/apm-rum'
const apm = initApm({
  serviceName: 'my-application',
  serverUrl: 'http://172.31.38.42:8200',
  serviceVersion: '1.0'
})
```

It is important that clients can connect to the APM Server.

- The RUM agent will automatically collect measures on the client's usage and also some error reporting

Supported Technologies

- Platforms
  - Android, Chrome, Edge, Firefox, Internet Explorer, Safari, and iPhone
- Frameworks
  - integrations for React, Angular, and Vue
- Single page applications
- User interactions through click event listeners that are registered by the application
- Many other features support and always more to come
  - https://www.elastic.co/guide/en/apm/agent/rum-js/current/supported-technologies.html

Lesson 3

Review - APM Agents

Summary

- APM agents usually automatically collect performance information about your application, but you can also manually instrument your application
- APM agents are language dependent, that is, you must install the APM agent according to the programming language your application is written
- APM agents use the **service.name** setting to report and group data based on the application name

Quiz

1. What are the two kinds of instrumentation that APM agents provide?
2. **True** or **False**: You should choose the APM agent language according to the programming language the application is written in.
3. Which setting is used to group data based on the application name?

APM Fundamentals

Lesson 3

Lab - APM Agents

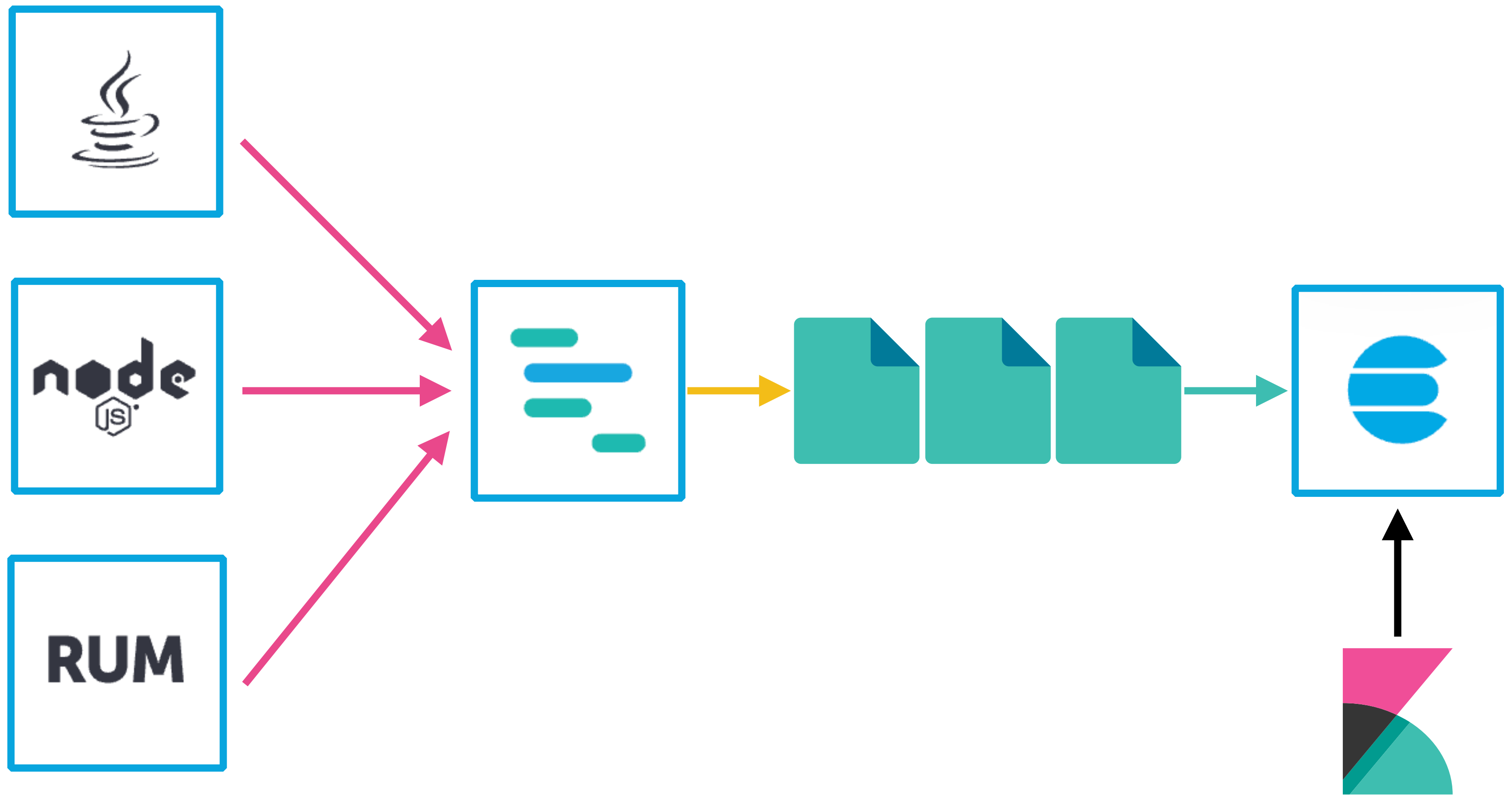
elastic

Lesson 4

Learn More

APM Fundamentals

- The automatic measures make it possible to pinpoint and fix performance issues and unhandled errors quickly
- It also makes it easier to monitor how many times specific issues are happening in the monitored application
- This is what you built during this training:



Next Steps

- This is an introductory course to APM
- Next, use Elastic APM to monitor your applications
  - check the response time of requests to your applications
  - check whether your applications have unhandled errors
  - visualize where your applications are spending time
- References
  - https://www.elastic.co/solutions/apm
  - https://www.elastic.co/guide/en/apm/get-started/current/index.html
- Join us on other trainings and deep dive into APM according to your needs

Other APM Courses

- Application Performance Monitoring (coming soon)
  - dive deep into the Elastic APM architecture
  - setup and configure Java, Node.js, and RUM agents
  - monitor both backend and frontend applications
  - use distributed tracing to monitor micro services architectures
- Analyzing APM Data
  - dive deep into your APM data with the Kibana APM UI
  - monitor how long your application is taking to answer requests
  - explore the spans of your transactions
  - find and fix errors in your applications
  - use the search bar to filter transactions or errors in real time

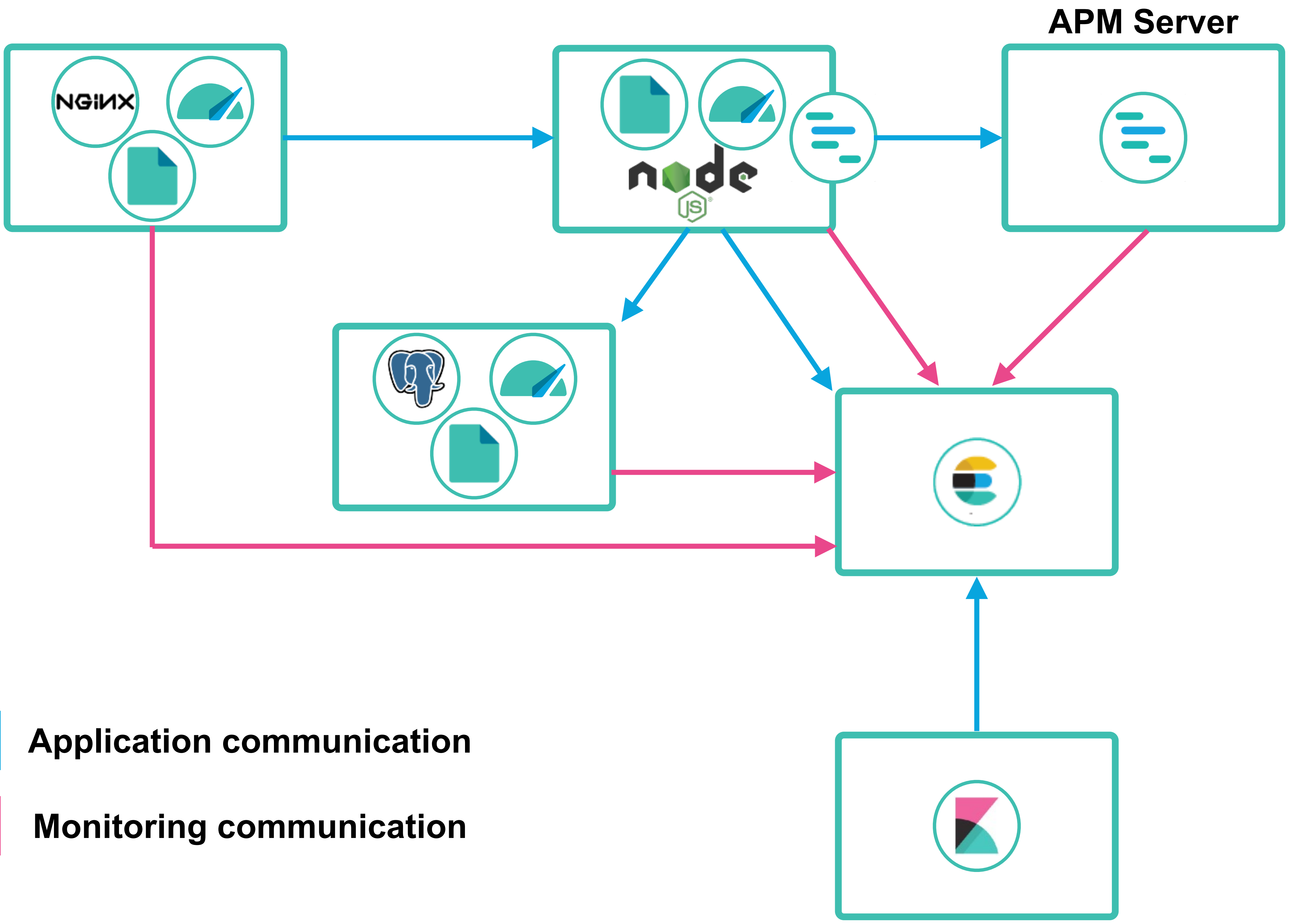
Other APM Courses

- Node.js Application Performance Monitoring
  - setup and configure a Node.js agent in your application
  - add custom transactions and spans to your applications
  - troubleshoot your application and the Node.js agent
  - tune the performance of your Node.js agent
- Real User Monitoring with Elastic APM
  - enable Real User Monitoring in the APM server
  - setup and configure a RUM JS agent in your client application
  - generate source maps to help debugging minified JavaScript
  - troubleshoot your application and the RUM JS agent

APM is Just Another Piece of the Puzzle

- APM gives you end-user experience and application-level monitoring
- APM is also crucial for a unified observability strategy
- But you can get even more data:
  - Logging
  - Metrics
- To really understand what's happening to your application you should combine them all:
  - Logging Fundamentals
  - Metrics Fundamentals
  - Observability Fundamentals

The Bigger Picture



🟦 **Application communication**

🟥 **Monitoring communication**

APM Fundamentals

Lesson 4

Review - Learn More

elastic

Summary

- Elastic APM makes it easier to find performance bottlenecks and errors in your application
- With Elastic APM you can improve end-user experience based on your application-level monitoring
- You can combine **APM**, **Logging**, and **Metrics** to have a full understanding of your applications and systems

Quiz

1. **True** or **False**: Elastic APM makes it easier to visualize performance bottlenecks and errors.
2. What can you improve in your application with the application-level monitoring provided by Elastic APM?
3. Which solutions are complementary to APM when it comes to monitoring an application?

APM Fundamentals

Lesson 4

Lab - Learn More

elastic

Quiz Answers

Introduction to Elastic APM

1. True
2. True
3. APM agents, APM server, Elasticsearch and Kibana APM UI

APM Server

1. True
2. You can use the Kibana APM UI
3. True

APM Agents

1. Automatic and manual
2. True
3. Agents use the service.name setting to report data based on the application name, so they can be grouped in the APM UI

Learn More

1. True
2. End-user experience
3. Logging and Metrics