



Elastic Observability Engineer

Elastic 公式トレーニングコース

elastic.co/training

Elastic バーチャルトレーニングへようこそ



- 皆さんが講師の説明を受講できるよう、音声/ビデオのテストからトレーニングを開始します
- 音声/ビデオの問題を避けるため、次の点をご確認ください：
 - サポート対象のブラウザを利用: Chrome, Firefox
 - ページを "incognito", "プライベート", "シークレット" 画面で開く
 - 広告、スクリプトブロッカー、プロキシ、VPN を無効化
- 問題が発生したら、以下を順にお試しください：
 - 右上のビデオパネルをクリックし、音声を有効化
 - Web ページを更新
 - 他のブラウザを試す
 - 最後の手段、コンピュータの再起動で解決する場合も

Elastic トレーニングへようこそ

- **learn.elastic.co** からログイン
 - 登録 email に記載の手順でアクセス
- "**My Enrollments**" から本日のトレーニングをクリック
- "**Content**" タブからコースの教材ファイルをダウンロード
 - コースで利用するスライドの PDF とラボ手順書が含まれます
- "**Access your virtual class here**" をクリックしてラボ環境にアクセス

Elastic トレーニングについて

- 実施環境
 - Strigo test: app.strigo.io/system-test
- 講師の紹介
- Code of Conduct
 - www.elastic.co/community/codeofconduct

Agenda

- **Module 1: Getting started**
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理

Getting started

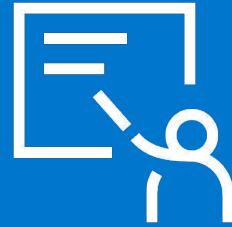
Module 1

Topics

- Elastic Observability
- Uptime
- Discover

Elastic Observability

Module 1 Lesson 1



なぜオブザーバビリティ?

- モダンなアプリケーション
 - 分散
 - マイクロサービス
 - サイロ化したデータ
- アプリケーションやシステムに関する問い合わせを見つけるのはますます困難になってきている
 - 何がこの障害を引き起こした?
 - SLA に準拠できているか?

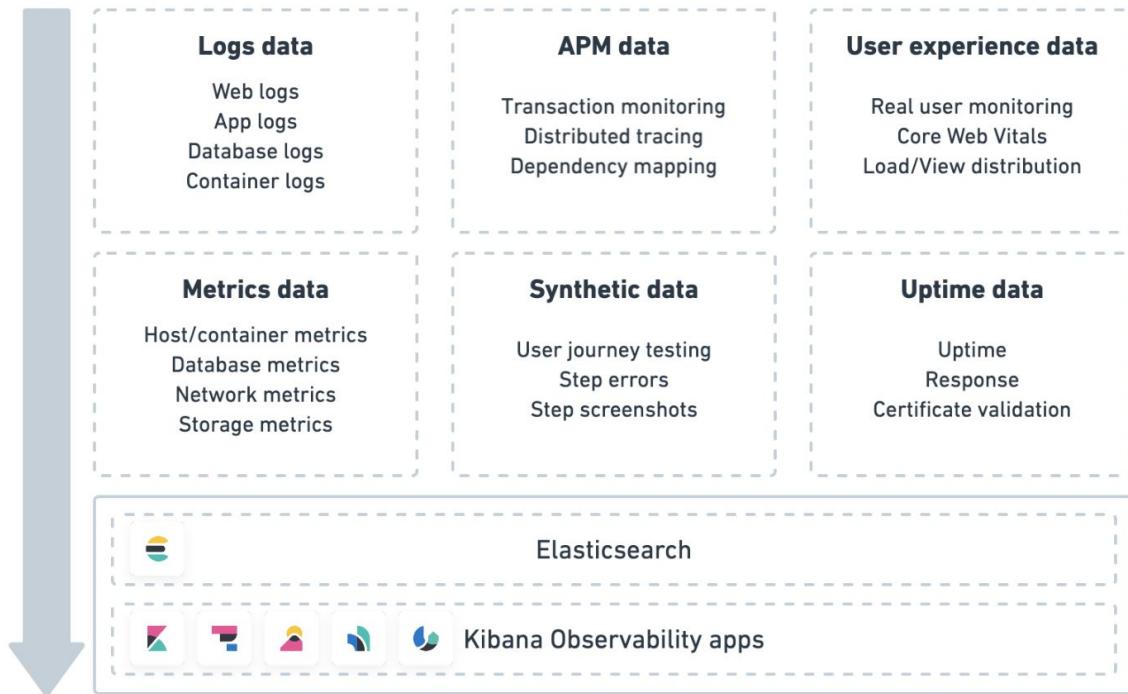
SLA =
Service Level
Agreement

オブザーバビリティとは?

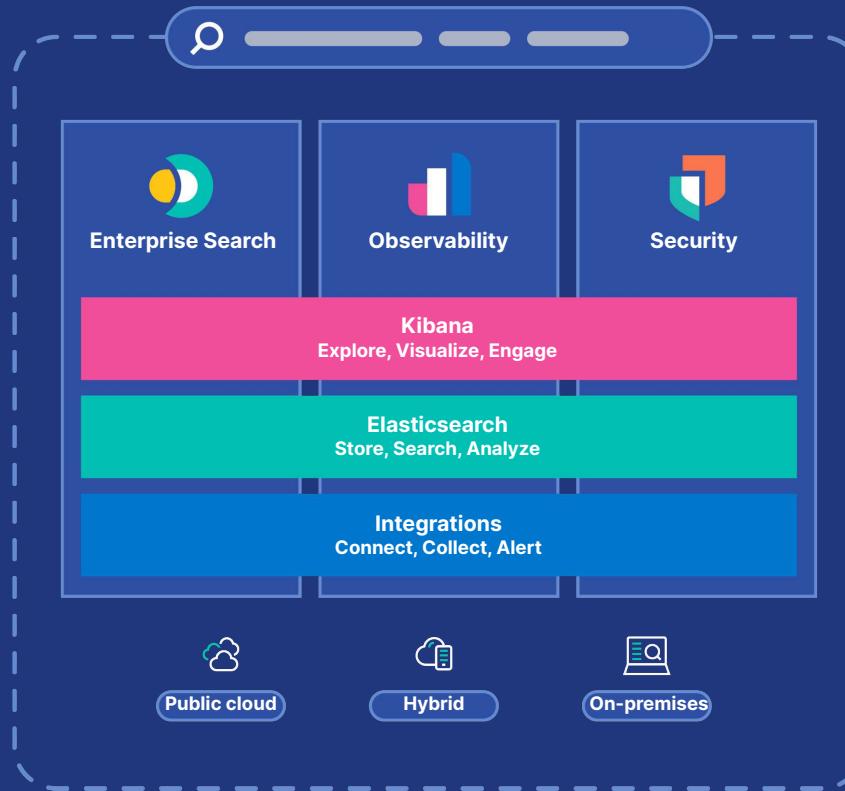
- ソフトウェアシステムの機能要件
- ソフトウェアシステムの非機能要件
 - 操作性、可用性、拡張性、運用性
 - **可観測性 (observability)?**
- モニタリングとは何が違う?
 - 進化?
 - 未知を知る vs 知られることのない未知

Elastic Observability

- ひとつのスタックで全てのオブザーバビリティデータを
- End-to-end のオブザーバビリティとアラート向けの統合ソリューション



The Elastic Search Platform

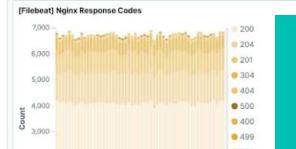
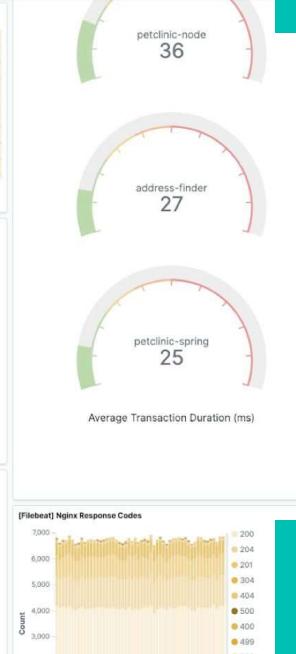


シンプルなユーザインタフェース

- SLI のサマリと根本原因分析を同一の UI で



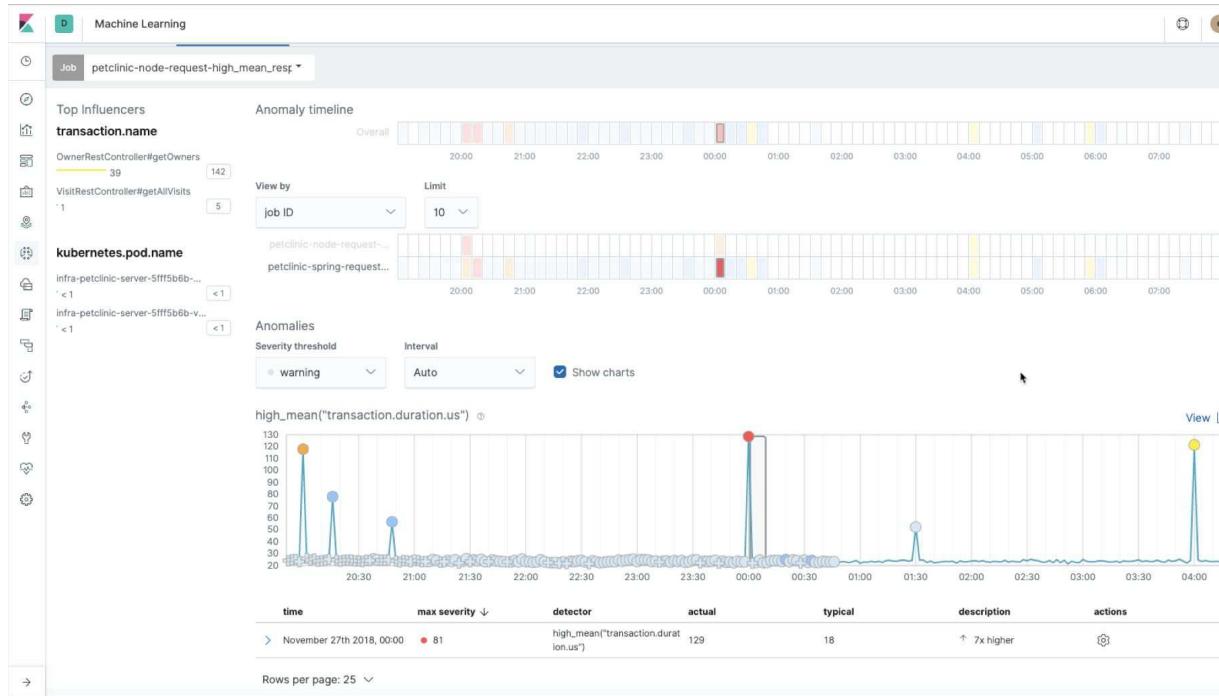
SLI =
Service Level Indicator



SLI はサービスレベルの観点を計測したもの

Machine learning

- 複数のデータソースを関連付け、よりよい異常検知を



Alerting

- 運用データの SLI を SLO と比較してアラートを発報

The screenshot shows the Elasticsearch Watcher interface with the following details:

- Name:** Response Sizes Large
- Indices to query:** apm-7.0.0-span
- Time field:** @timestamp (selected from a dropdown menu)
- Run watch every:** 1 minutes
- Matching the following condition:**

```
WHEN max() OF node_stats.os.cpu.load_average.1m GROUPED OVER top 20 'source_node.name' IS ABOVE 10 FOR THE LAST 100 minutes
```
- Graph:** A line graph showing CPU load average over time. A red horizontal line at approximately 6.5 represents the Service Level Objective (SLO). The graph shows several spikes above this level, particularly around 09:00 and 14:00.

SLI が SLO を満たせなかつたらアラート発報

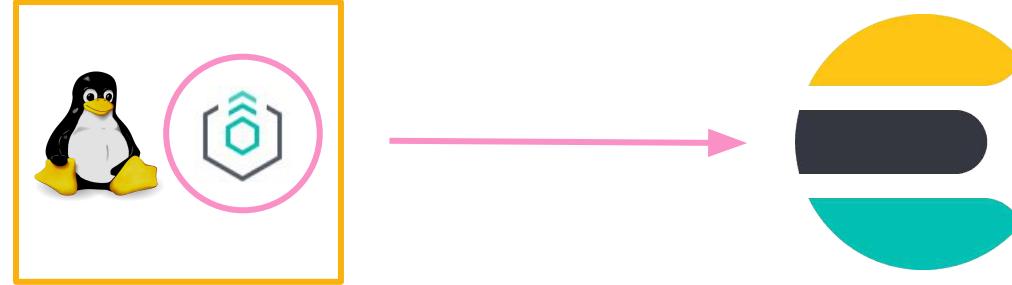
SLO = Service Level Objective

データを Elasticsearch に送信

- さまざまなソースからのデータがある
- 収集、保存、検索、分析、防御、可視化できるのが理想的
- まずは Elastic プラットフォームにこれらのデータを投入する必要がある
 - Elastic Agent
 - Beats

Fleet と Elastic Agent

- ログ、メトリック、uptime、セキュリティデータ、脅威回避を单一のエージェントで
- ひとつの統合エージェントを各ホストにインストールすればよい



Integrations

- Elastic Agentとともに利用するインテグレーションは Elastic と外部のサービスやシステムを簡単に接続する仕組みを提供

The screenshot shows the 'Integrations' page in the Elasticsearch interface. At the top, there's a navigation bar with 'Integrations' selected. Below it, a main heading 'Integrations' with the sub-instruction 'Choose an integration to start collecting and analyzing your data.' Underneath, there are two tabs: 'Browse integrations' (selected) and 'Installed integrations'. The page displays several integration cards:

- Web site crawler**: Adds search to your website with the App Search web crawler.
- Elastic APM**: Monitors, detects, and diagnoses complex performance issues from your application.
- Endpoint Security**: Protects your hosts with threat prevention, detection, and deep security data visibility.

Below these cards is a sidebar with 'All categories' (252) and a search bar. The categories listed are:

- AWS (24)
- Azure (23)
- Cloud (37)
- Communications (3)
- Config management (2)
- Containers (12)
- Custom (22)

There are also several smaller integration cards shown in a grid:

- 1Password Events Reporting**: Collect events from 1Password Events API with Elastic Agent.
- AbuseCH**: Collect threat intelligence from AbuseCH API with Elastic Agent.
- ActiveMQ Logs**: Collect and parse logs from ActiveMQ instances with Filebeat.
- ActiveMQ Metrics**: Collect metrics from ActiveMQ instances with Metricbeat.
- Aerospike Metrics**: Collect metrics from Aerospike servers with Metricbeat.
- Akamai**: Akamai Integration.

Fleet

- Fleet は多数のエージェント群を中央管理するコントロールプレーンとして動作する

The screenshot shows the Fleet interface in the Elasticsearch web UI. The top navigation bar includes a sidebar icon, a 'D' button, a 'Fleet' tab (which is active), and an 'Agents' tab. On the right, there's a 'Send Feedback' link. The main title 'Fleet' is displayed, followed by the subtitle 'Centralized management for Elastic Agents.' Below this, a navigation bar has tabs for 'Agents' (which is selected and underlined), 'Agent policies', 'Enrollment tokens', 'Data streams', and 'Settings'. A search bar with a magnifying glass icon and the placeholder 'Search' is positioned above a table. To the right of the search bar are buttons for 'Status' (dropdown), 'Agent policy' (dropdown showing 3), 'Upgrade available' (button), and '+ Add agent' (button). Below the search bar, it says 'Showing 1 agent'. To the right, status indicators show 1 Healthy, 0 Unhealthy, 0 Updating, and 0 Offline agents. The table below has columns: Host, Status, Agent policy, Version, Last activity, and Actions. One agent is listed: 'b5ffaa9f1cc2' with 'Healthy' status, 'Default agent policy' (with a lock icon and 'rev. 5'), '8.1.2' version, and '12 seconds ago' last activity. At the bottom left, it says 'Rows per page: 20' with a dropdown arrow. At the bottom right, there are navigation arrows for pages 1 and 2.

Host	Status	Agent policy	Version	Last activity	Actions
b5ffaa9f1cc2	Healthy	Default agent policy rev. 5	8.1.2	12 seconds ago	

Beats

- フリーでオープンな単一目的のデータシッパープラットフォーム
- Beats は多様なフレーバーがある軽量なエージェント



Filebeat



Metricbeat



Packetbeat



Winlogbeat



Auditbeat



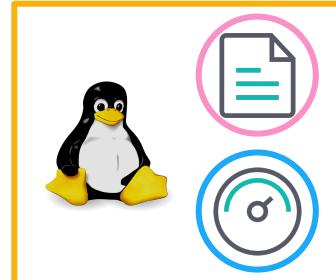
Heartbeat



Functionbeat

Community beat

- 通常各ホストに複数の必要な Beats をインストール

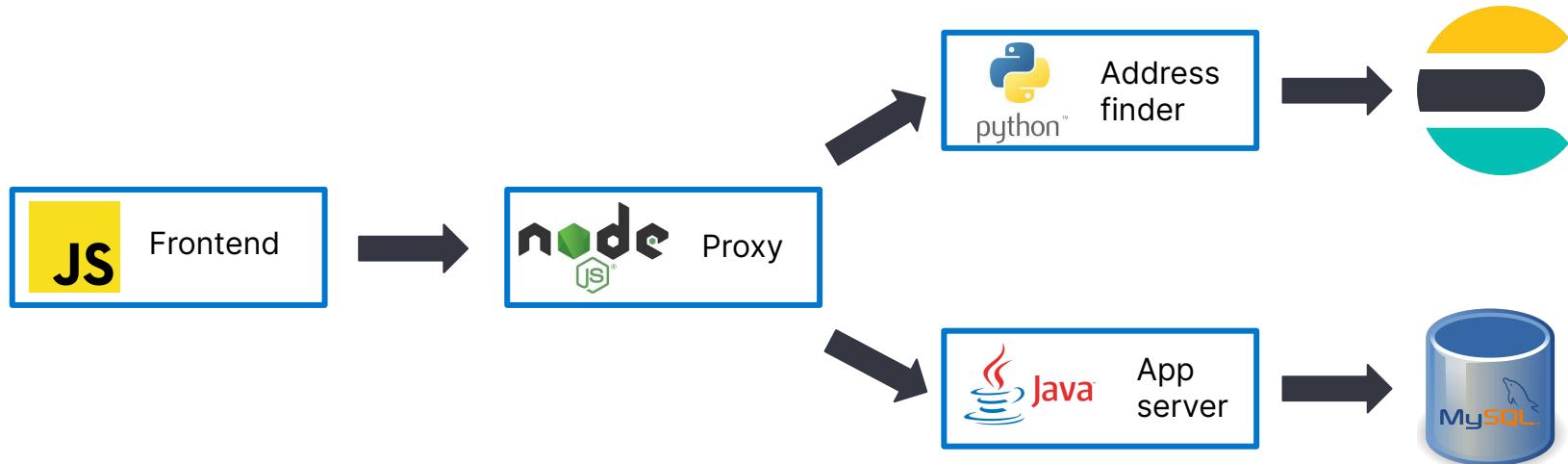


Elastic Agent or Beats?

- It depends!
- Elastic Agent は単一のバイナリで Beats と同じ機能を提供
- しかし、まだ機能的なギャップがある場合も
- 以下のガイドラインに従って選択:
 - 必要なインテグレーションがサポートされ GA になっているか
 - インテグレーションが利用できるなら、[サポート対象のアウトプット](#) を確認
 - [機能比較](#) を確認し必要な機能をチェック
- もし上記の3手順を満たすなら、Elastic Agent を利用
- そうでないならレガシーな Beats を使い今後のアップデートに注目

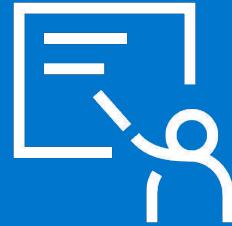
ペットクリニックアーキテクチャ

- トレーニングを通してこのアプリケーションを観測していく



Summary: **Elastic Observability**

Module 1 Lesson 1



Summary

- **Observability** とはシステムの特性
- **Observability** は予期せぬ振る舞いを検知するのに役立つ
- **Observability** は詳細なインサイトとコンテキストを提供
- **Observability** はすばやい根本原因の発見、修正を助ける
- **Elastic Observability** はひとつの統合された実装を提供

Quiz

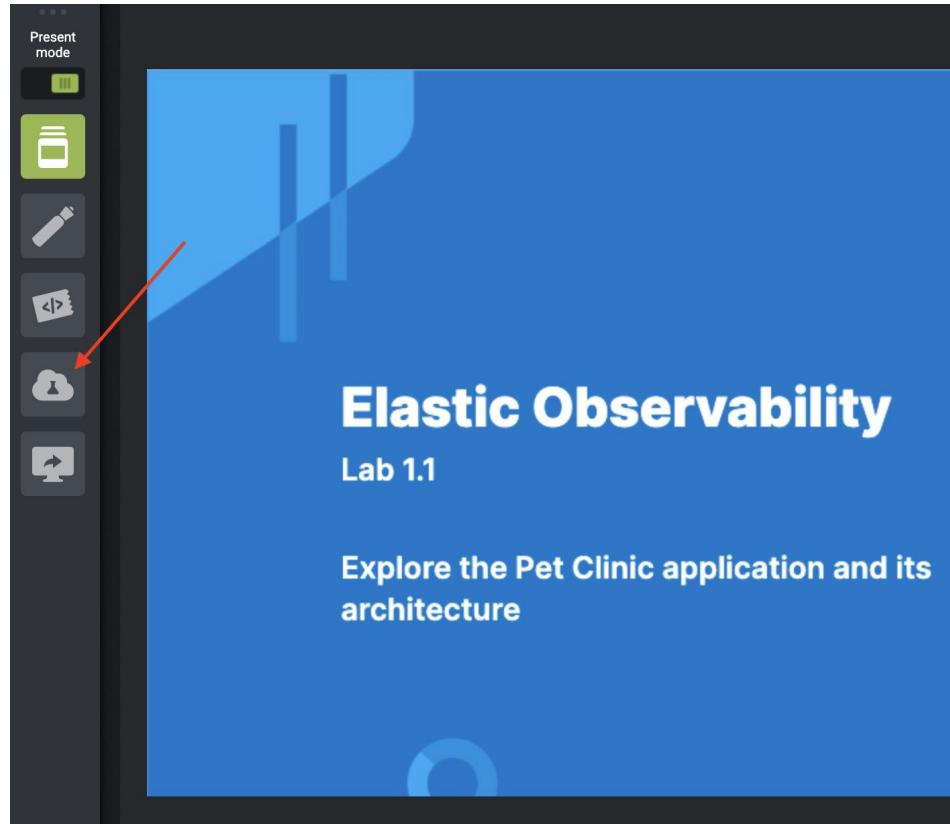
1. **True or False:** Observability は uptime の監視だけに関するものである
2. **True or False:** Observability は全く新しい技術である
3. **True or False:** Elastic Observability は end-to-end の可観測性とアラートを実現する統合されたソリューションを提供

ラボ環境



ラボ環境

- 共有された Strigo のリンク
へアクセス
- 左側の "My Lab" をクリック



ラボ環境

- ラボ環境には二つの Elasticsearch クラスタがある
- Lab Instructions** をクリックして開始

別ウィンドウでラボを表示することもできる

The screenshot shows a web-based course interface for 'Elastic Observability Engineer'. On the left, there's a sidebar with icons for 'Virtual machine' (status: ready), 'Follow presenter', and several other tools. The main header bar includes tabs for 'Virtual machine', 'Terminal', 'Editor', 'Kibana', 'Lab Ins...', and 'Petclinic'. A pink circle highlights the 'Lab Ins...' tab. Below the header, the title 'Elastic Observability Engineer' is displayed, along with a search bar and navigation links for 'Home', 'Labs', 'Lab Environment', 'Troubleshooting', and 'Quickstart'. The central content area is titled 'Home' and features the 'elastic' logo. To the right, a 'Table of contents' sidebar lists 'Requirements', 'Navigation Tips', 'Global Shortcuts', and 'Search Bar'. At the bottom, the text 'Course Lab Book' is visible.

Elastic Observability

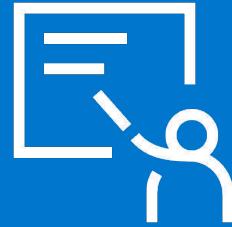
Lab 1.1



ペットクリニックアプリケーションとそのアーキテクチャを探索してみましょう

Uptime

Module 1 Lesson 2

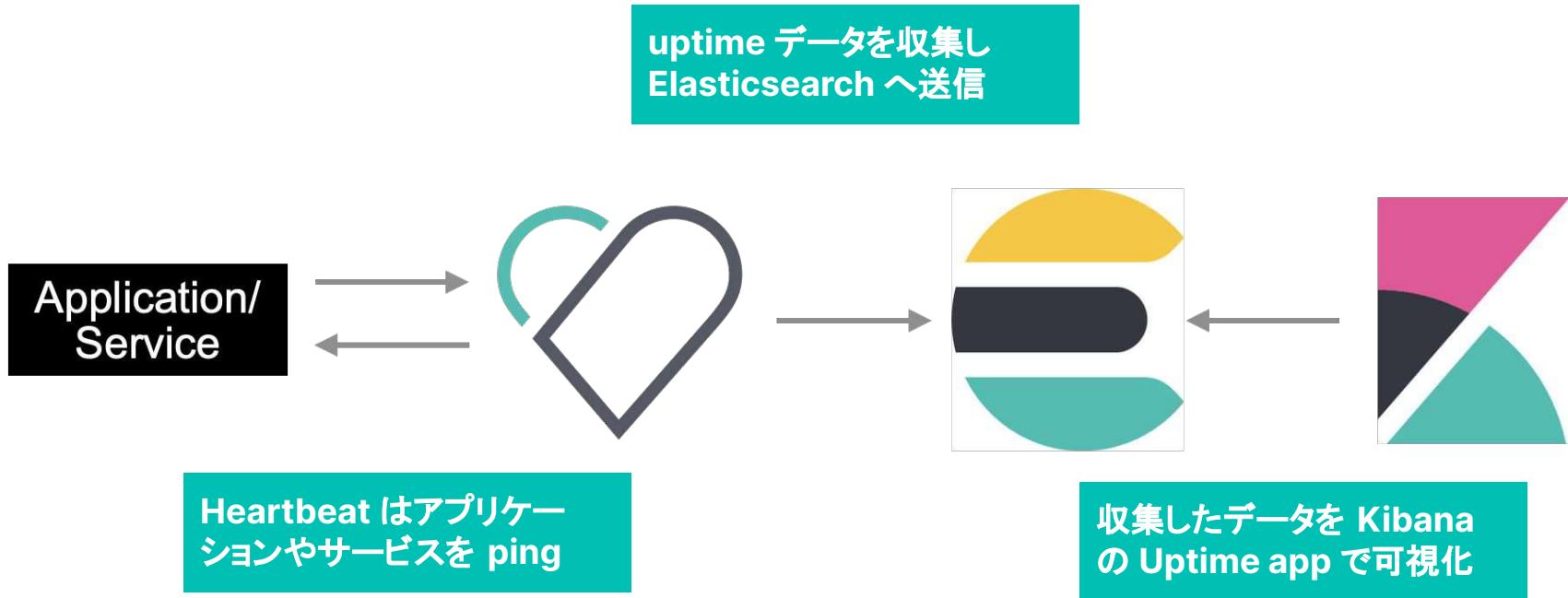


なぜ Uptime?

- システムが生きていないと観測もできない
- よって通常はここから始める:
 - システムは利用可能?

Heartbeat

- Uptime モニタリングの軽量なシッパー



Heartbeat のモニタリング

- ICMP
 - v4 と v6 (echo requests)
 - root アクセスが必要
- TCP
 - カスタムペイロードを送信/受信してエンドポイントを検証
- HTTP
 - 期待するレスポンスが返るかでホストを検証
 - e.g. ステータスコード、レスポンスヘッダ、コンテンツ
- TCP と HTTP はどちらも SSL/TLS に対応
 - プロキシ設定もサポート

サンプル設定

```
heartbeat.monitors:  
- type: icmp  
  id: ping-myhost  
  name: My Host Ping  
  hosts: ["myhost"]  
  schedule: '*/5 * * * *'  
- type: tcp  
  id: myhost-tcp-echo  
  name: My Host TCP Echo  
  hosts: ["myhost:777"]  # default TCP Echo Protocol  
  check.send: "Check"  
  check.receive: "Check"  
  schedule: '@every 5s'  
- type: http  
  id: service-status  
  name: Service Status  
  hosts: ["http://localhost:80/service/status"]  
  check.response.status: [200]  
  schedule: '@every 5s'
```

单一の Heartbeat インスタンスで複数エンドポイントを ping できる

ICMP で myhost を 5s ごとに ping
(10:00:00, 10:00,05, ...)

TCP で myhost:777 の送受信文字列を 5s ごとに監視

HTTP で localhost:80 のレスポンステータスを監視

Heartbeat の利用開始手順

1. Heartbeat をインストール
2. Heartbeat を設定
3. 設定をテスト (任意)
4. Output をテスト (任意)
5. Elasticsearch インデックステンプレートと Kibana ダッシュボードをセットアップ
6. Heartbeat を開始
7. Elasticsearch 内のデータを確認

```
./heartbeat test config
```

```
./heartbeat test output
```

```
./heartbeat setup -e
```

```
./heartbeat -e
```

The `-e` フラグは任意の設定、
`syslog` の代わりに標準エラー
へ出力

Uptime データの確認

- Heartbeat はデータを Elasticsearch に送信
- Discover で data view を確認 (次のレッスンで詳細を説明)
- または Uptime アプリ で詳細なインサイトを得る



Overview

Alerts

Cases

Logs

Metrics

APM

Uptime

User Experience

Uptime アプリ

Monitors

Search by monitor ID, name, or url (E.g. http://)

モニターをステータス、
URL、名前、ポートでグル
ープ化

1 m

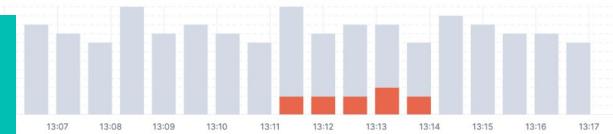
Refresh

5 Monitors



Pings over time

アベイラビリティでフィルタ、
ホストの応答時間履歴



Monitors

All Up Down

Status	Name	Url	Tags	TLS Certificate	Downtime history	Status alert
Up	Elastic.co Homepage	https://elastic.co		Expires in 3 months	--	<input type="checkbox"/>
Up	Elasticsearch REST API	https://elasticsearch:9200		Expires in 3 years	--	<input type="checkbox"/>
Up	MySQL Service	tcp://mysql:3306		--	--	<input type="checkbox"/>
Up	Pet Clinic Servers	icmp://petclinic-server		--	<input type="checkbox"/>	
Up	Pet Clinic Servers	icmp://petclinic-client		--	<input type="checkbox"/>	

Rows per page: 10

全サービスのステータス概
要をチェック

さらなるインサイトを取得

Monitors

Last 15 minutes 1 m Refresh

Search by monitor ID, name, or url (E.g. http://)

Location 0 Port 3 Scheme 3 Tag 0

5 Monitors

Pings over time

up/down だけではない

Monitors All Up Down

Status	Name	Url	Tags	TLS Certificate
Up	Elastic.co Homepage	https://elastic.co		Expires in 3 months
Up	Elasticsearch REST API	https://elasticsearch:9200		Expires in 3 years
Up	MySQL Service	tcp://mysql:3306	--	--
Up	Pet Clinic Servers	icmp://petclinic-server		
Up	Pet Clinic Servers	icmp://petclinic-client		

Rows per page: 10 >

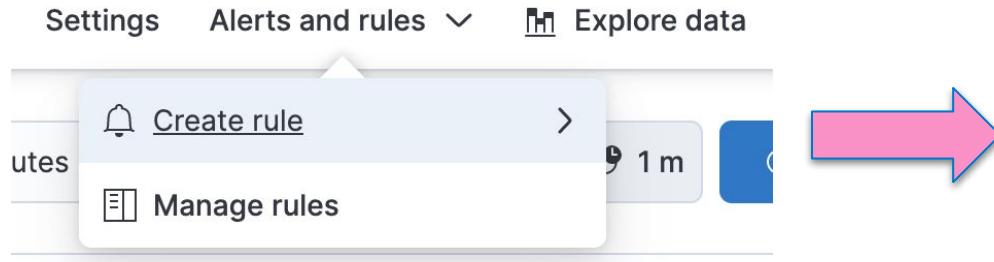
TLS 証明書期限をモニタリング

利用可能なインテグレーションを確認

アラートを受け取る

- アラートはトレーニングの後半で詳細に解説

エラーや障害に基づく通知を受信



Create rules

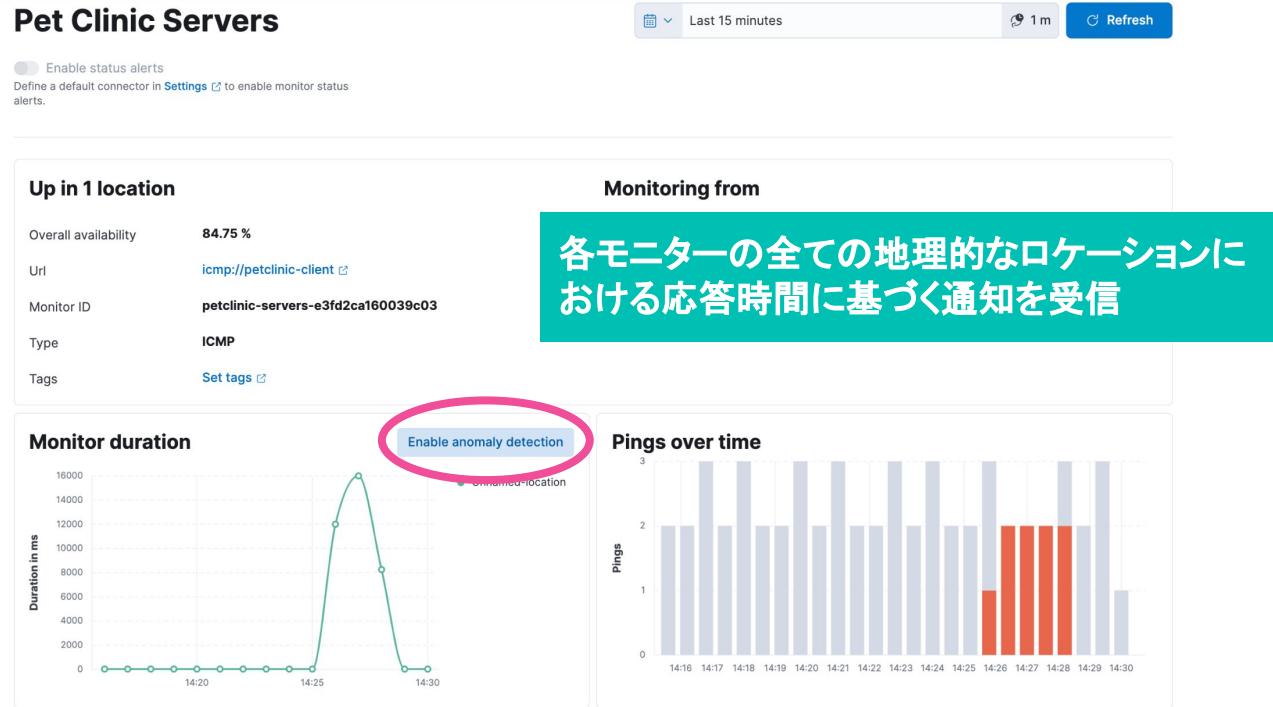
Monitor status rule

TLS rule

TLS 証明書の期限が近づいたら通知を受信

問題発生を未然に検知

- マシンラーニングはトレーニングの後半で扱う



Summary: Uptime

Module 1 Lesson 2



Summary

- **Heartbeat** は ICMP, TCP, HTTP でサービスの可用性をチェック
- 単一のインスタンスで複数エンドポイントを監視可能
- **Uptime app** は Heartbeat データのクリアなビュー
- Uptime はトレンドのモニタリングや証明書失効のインサイトを高めることができる
- Uptime はモニターや TLS ステータスに基づくアラートを設定可能

Quiz

1. Heartbeat で設定可能な三種類のモニターとは?
2. **True or False:** Uptimeではどのサービスがいつ利用可能だったかしか表示しない
3. **True or False:** モニタしたい全てのシステムに Heartbeat をデプロイする必要がある

Uptime

Lab 1.2

ペットクリニックアーキテクチャの uptime を監
視しましょう



Discover

Module 1 Lesson 3



Elasticsearch と Kibana の概要

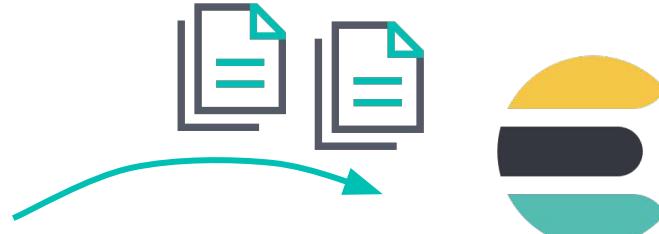
- Elasticsearch
 - オブザーバビリティデータを保存する、検索、分析エンジン
- Kibana
 - Elasticsearch にインデックスされたデータの検索、データ可視化機能を提供
- Discover
 - オブザーバビリティデータに関する疑問への答えを探索するのに役立つ Kibana アプリ

Elasticsearch ドキュメント

- Elastic プラットフォームではデータは Elasticsearch に保存される
- Elasticsearch は ドキュメントストア
 - ドキュメントとよばれる JSON オブジェクトを保存する

Timestamp	Favorite Prc	Favorite Marvel	Favorite Op	Current company	industry	Country	th	How long	Current po	Wha	Wha	Wha	Wha	Fa
2017/02/17 5:26:18	Java	spider-man	OSX	Software Products	Web	US	11+	Engineer	1	3	4	4	El	
2017/02/17 5:28:41	HTML	Thor	OSX	Software Products	Netherlands	11+		Manager	0	1	1	2	Ds	
2017/02/17 5:28:47	Javascript	don't have one	OSX	Software Products	Canada	2-5		Developer	0	0	2	3	ES	
2017/02/17 5:29:30	Java		OSX	Software Products	Britain	11+			1	2	4	2	MI	
2017/02/17 5:30:27	Python	Dr. Doom												
2017/02/17 5:30:31	Go													
2017/02/17 5:30:52	Java	Iron Man												
2017/02/17 5:33:10	Fortran	Wolverine												
2017/02/17 5:42:14	Python	Ghost Rider												


```
Caused by: java.lang.ExceptionInInitializationError
1   at org.elasticsearch.common.logging.ESLogger.error(ESLogger.java:110)
2   at org.elasticsearch.common.xcontent.XContentRegistry$Builder.build(XContentRegistry.java:110)
3   at org.elasticsearch.common.xcontent.XContentRegistry$Builder.build(XContentRegistry.java:109)
4   at org.elasticsearch.common.xcontent.XContentRegistry$Builder.build(XContentRegistry.java:109)
5   at org.elasticsearch.common.xcontent.XContentRegistry$Builder.build(XContentRegistry.java:109)
6   at org.elasticsearch.common.settings.Settings$Builder.build(Settings.java:109)
7   at org.elasticsearch.common.settings.Settings$Builder.build(Settings.java:109)
8   at org.elasticsearch.common.settings.Settings$Builder.build(Settings.java:109)
9   at org.elasticsearch.common.settings.Settings$Builder.build(Settings.java:109)
10  at org.elasticsearch.common.network.NetworkModule$Builder.build(NetworkModule.java:109)
11  at org.elasticsearch.client.transport.TransportClient$Builder.build(TransportClient.java:109)
12  at org.elasticsearch.client.transport.TransportClient$Builder.build(TransportClient.java:109)
13  at org.elasticsearch.client.transport.TransportClient.init(TransportClient.java:268)
14  at org.elasticsearch.transport.client.PreBuiltTransportClient.<init>(PreBuiltTransportClient.java:125)
15  at org.elasticsearch.transport.client.PreBuiltTransportClient.<init>(PreBuiltTransportClient.java:111)
16  at org.elasticsearch.transport.client.PreBuiltTransportClient.<init>(PreBuiltTransportClient.java:101)
17  at com.regioncom.bpo.rcease.util.TransportClientFactory.configureClients(TransportClientFactory.java:81)
```



フィールドとバリュー

- ドキュメントには複数のフィールドがある
- 各フィールドは:
 - 0個以上の バリューを持つ
 - データ型を持つ

```
{  
    "category": [  
        "Women's Clothing",  
        "Women's Shoes"  
    ],  
    "currency": "EUR",  
    "customer_full_name": "Pia Carr",  
    "customer_id": 45,  
    "order_date": "2021-12-09T11:11:02+00:00",  
    "order_id": 569968,  
    ...  
}
```

フィールド

バリュー

データに関する疑問をお持ちでしょう

- どんなデータが Elasticsearch に保存される?
- どの web ページに特定の単語が含まれる?
- 最近ログにはどんなイベントが出力されている?
- どの処理が 500 ms より長くかかっている?

Discover

- 素早くデータを検索、フィルタして答えの発見に役立つ

The screenshot shows the Elasticsearch Discover interface. On the left, there's a sidebar with a 'Analytics' icon and a 'Discover' button highlighted with a pink arrow pointing from the 'Discover' section above. Below it are other options: Dashboard, Canvas, Maps, Machine Learning, Graph, and Visualize Library. The main area has a 'Discover' tab selected. A search bar at the top contains 'heartbeat-*'. The results show 242 hits. A histogram on the right shows document counts for time intervals. Below the histogram is a 'Field statistics' section. A modal window titled 'A better way to explore' provides information about field statistics and includes a 'Learn more' button.

Analytics

Discover

Dashboard

Canvas

Maps

Machine Learning

Graph

Visualize Library

Discover

Analytics

Search

+ Add filter

heartbeat-*

242 hits

Documents Field statistics (BETA)

Apr 18, 2022 @ 10:53:27.613 - Apr 18, 2022 @ 11:08:27.613

A better way to explore

Experience the new Document Explorer. Understand the shape of your data with Field Statistics.

Learn more

1 field sorted

@timestamp

Document

Apr 18, 2022 @ 11:08:20.723

@timestamp Apr 18, 2022 @ 11:08:20.723 agent.ephemeral_id 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname ip-172-31-12-220 agent.id 717b0798-3bad-445f-be09-ed85d1e70d10 agent.name ip-172-31-12-220 agent.type heartbeat agent.version 8.2.0 ecs.version 8.0.0 event.dataset http://response.body.bytes 539 http.response.body...

Apr 18, 2022 @ 11:08:20.686

@timestamp Apr 18, 2022 @ 11:08:20.686 agent.ephemeral_id 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname ip-172-31-12-220 agent.id 717b0798-3bad-445f-be09-ed85d1e70d10 agent.name ip-172-31-12-220 agent.type heartbeat agent.version 8.2.0 ecs.version 8.0.0 event.dataset icmp_icmp.requests 1 icmp.rtt.us 185 monitor.check...

Apr 18, 2022 @ 11:08:20.686

@timestamp Apr 18, 2022 @ 11:08:20.686 agent.ephemeral_id 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname ip-172-31-12-220 agent.id 717b0798-3bad-445f-be09-ed85d1e70d10 agent.name ip-172-31-12-220 agent.type heartbeat agent.version 8.2.0 ecs.version 8.0.0 event.dataset icmp_icmp.requests 1 icmp.rtt.us 294 monitor.check...

Apr 18, 2022 @ 11:08:20.631

@timestamp Apr 18, 2022 @ 11:08:20.631 agent.ephemeral_id 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname ip-172-31-12-220 agent.id 717b0798-3bad-445f-be09-ed85d1e70d10 agent.name ip-172-31-12-

しかし、まずは **data view** が必要

- インテグレーションは自動で設定してくれる
- Kibana で独自の設定も可能
 - メインメニュー → Stack Management → Data Views

The screenshot shows the Kibana navigation menu. On the left, under the 'Management' section, the 'Stack Management' item is highlighted with a blue background and white text. A large pink arrow points from this item towards the right side of the screen. On the right, a vertical list of items is displayed: 'Kibana' (with a question mark icon), 'Data Views', 'Saved Objects', 'Tags', 'Search Sessions', 'Spaces', and 'Advanced Settings'. The 'Data Views' item is also highlighted with a blue background and white text.

- Management
- Dev Tools
- Integrations
- Fleet
- Osquery
- Stack Monitoring
- Stack Management

Kibana ?

- Data Views
- Saved Objects
- Tags
- Search Sessions
- Spaces
- Advanced Settings

Kibana には data view が必要

- 探索したい Elasticsearch のデータにアクセスするため

You have data in Elasticsearch.
Now, create a data view.

Kibana requires a data view to identify which data streams, indices, and index aliases you want to explore. A data view can point to a specific index, for example, your log data from yesterday, or all indices that contain your log data.

[+ Create Data View](#)



Want to learn more? [Read the docs](#)

Data views

- data view で利用するデータを選択
- フィールドの属性を定義可能

Create data view

Name
heartbeat-*
Enter an index pattern that matches one or more data sources. Use an asterisk (*) to match multiple characters. Spaces and the characters . , ? , " , < , > , | are not allowed.

Timestamp field
@timestamp
Select a timestamp field for use with the global time filter.
Show advanced settings

Rows per page: 10

✓ Your index pattern matches 1 source.

heartbeat-8.2.0 Data stream

Create data view

Data view はひとつ以上のインデックス、
data stream、インデックスエイリアスを指す

Data views は以前
index patterns と呼
ばれていた

heartbeat-*

Time field: @timestamp Default

View and edit fields in heartbeat-. Field attributes, such as type and searchability, are based on field mappings in Elasticsearch.

Fields (1456) Scripted fields (0) Field filters (0)

Name	Type	Format	Searchable	Aggregatable	Excluded
@timestamp	date		●	●	
_id	_id		●		
_index	_index		●	●	
_score					
_source	_source				
_type					
agent.build.original	keyword		●	●	
agent.ephemeral_id	keyword		●	●	
agent.hostname	keyword		●	●	

データを見つける

そして閲覧するデータの
時間範囲を指定する

探索したいデータがある
場所を Kibana に伝える

The screenshot shows the Kibana interface with the following elements:

- Top Bar:** Includes "Discover", "Options", "New", "Open", "Share", "Inspect", and "Save" buttons.
- Search Bar:** Contains a "Search" input field and a "KQL" dropdown set to "Last 15 minutes".
- Filter Bar:** Shows a selected filter "heartbeat-*".
- Document List:** Displays 242 hits. A histogram at the top shows document counts across time intervals from 53' to 8'. The main area lists documents sorted by @timestamp, with rows per page set to 100.
- Bottom Navigation:** Includes a "Rows per page" dropdown and a page navigation bar with buttons for <, 1, 2, 3, >.

どんなフィールドが設定されているか確認する

The screenshot shows the Elasticsearch Settings interface for the 'heartbeat-*' index pattern. It highlights the 'monitor' field type and the 'any' type filter.

Available fields:

- _id
- _index
- _score
- @timestamp
- agent.ephemeral_id
- agent.hostname
- agent.id
- agent.name
- agent.type
- agent.version
- ecs.version
- event.dataset
- http.response.body.bytes
- http.response.body.hash
- http.response.headers.Age

Selected field: monitor

Filter by type: 0

Available fields (for monitor type):

- monitor.check_group
- monitor.duration.us
- monitor.id
- monitor.ip
- monitor.name
- monitor.status
- monitor.timespan
- monitor.type

Search results: 242 hits

Filter by type: 0

Available fields:

- monitor.check_group
- monitor.duration.us
- monitor.id
- monitor.ip
- monitor.name
- monitor.status
- monitor.timespan
- monitor.type

Type filter: any

- any
- string
- number
- _source
- date
- geo_point
- ip
- boolean
- unknown
- date_range

Page footer:

Copyright Elasticsearch BV 2015-2022 Copying, publishing and/or distributing without written permission is strictly prohibited

Elastic logo:

どんな種類のデータが収集されているか確認する

The screenshot shows the Elasticsearch Kibana interface. On the left, a search bar contains the query "monitor". Below it, a "Filter by type" dropdown is set to 0. A sidebar titled "Available fields" lists various monitoring fields: monitor.check_group, monitor.duration.us, monitor.id, monitor.ip, monitor.name, monitor.status, monitor.timespan, and monitor.type. A red circle highlights the "+" button in the "Available fields" sidebar.

In the center, a chart titled "242 hits" displays a histogram of values from 53' to 56'. The x-axis shows dates from April 18, 2022, at 10h. The y-axis ranges from 0 to 10. The distribution is roughly uniform across the bins.

Below the chart, a table titled "monitor.name" shows the top 5 values:

monitor.name	Value
Pet Clinic Servers	49.6%
Elasticsearch REST API	37.2%
MySQL Service	12.4%
Elastic.co Homepage	0.8%

At the bottom of this section, it says "Exists in 242 / 242 records".

On the right, a table titled "http.response.status_code" shows the status codes for the same timestamp range:

@timestamp	monitor.name	http.response.status_code
Apr 18, 2022 @ 11:08:20.723	Elasticsearch REST API	200
Apr 18, 2022 @ 11:08:20.686	Pet Clinic Servers	-
Apr 18, 2022 @ 11:08:20.686	Pet Clinic Servers	-
Apr 18, 2022 @ 11:08:20.631	MySQL Service	-
Apr 18, 2022 @ 11:08:10.722	Elasticsearch REST API	200

At the bottom of this table, it says "Rows per page: 100".

特定のレコードにズームする

Search

+ Add filter

heartbeat-*

Search field names

Filter by type 0

tcp.rtt.connect.us
tls.certificate_not_valid_after
tls.certificate_not_valid_before
tls.cipher
tls.established
tls.rtt.handshake.us
tls.server.hash.sha1
tls.server.hash.sha256
tls.server.x509.issuer.common_name
tls.server.x509.issuer.distinguished_name
tls.server.x509.not_after
tls.server.x509.not_before
tls.server.x509.public_key_algorithm
tls.server.x509.public_key_exponent
tls.server.x509.public_key_size
tls.server.x509.serial_number

240 hits

Apr 18, 2022, 13h

10
5
0

22' 23' 24' 25' 26'

1 field sorted

↓ @timestamp

	@timestamp	Document
✓	Apr 18, 2022 @ 13:36:51.315	@timestamp Apr 18, 2022 13:36:51.315 31-12-220 agent.id 7220 agent.type heartbeat
✓	Apr 18, 2022 @ 13:36:51.070	@timestamp Apr 18, 2022 13:36:51.070 31-12-220 agent.id 7220 agent.type heartbeat
✓	Apr 18, 2022 @ 13:36:51.070	@timestamp Apr 18, 2022 13:36:51.070 31-12-220 agent.id 7220 agent.type heartbeat
✓	Apr 18, 2022 @ 13:36:50.819	@timestamp Apr 18, 2022 13:36:50.819 31-12-220 agent.id 7220 agent.type heartbeat
✓	Apr 18, 2022 @ 13:36:41.313	@timestamp Apr 18, 2022 13:36:41.313 31-12-220 agent.id 7220 agent.type heartbeat

Rows per page: 100

Expanded document

View: Single document Surrounding documents

K < 3 of 240 >

Actions	Field	Value
①	_id	EzKIPYAB3qFky30Ht5_0
①	_index	.ds-heartbeat-8.2.0-2022.04.18-000001
②	_score	-
③	@timestamp	Apr 18, 2022 @ 13:36:51.070
④	agent.ephemeral_id	2d974062-870d-40b4-a44f-b9e2aa926648
④	agent.hostname	ip-172-31-12-220
④	agent.id	7f7b0798-3bad-445f-be09-ed85d1e70d10
④	agent.name	ip-172-31-12-220
④	agent.type	heartbeat
④	agent.version	8.2.0
④	ecs.version	8.0.0
④	event.dataset	icmp
④	icmp.requests	1

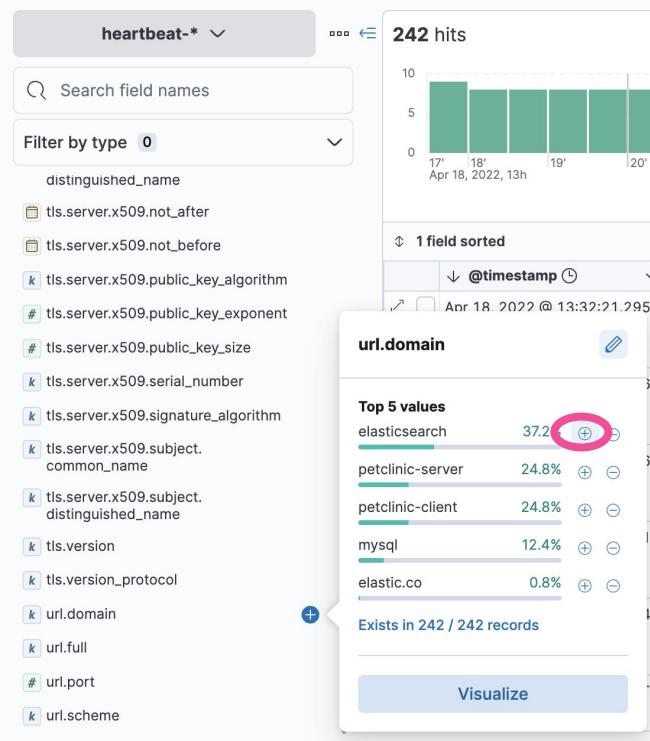
シンプルなレコード検索を実行する

The screenshot shows the Elasticsearch interface with a search bar at the top containing the query "elasticsearch". A pink circle highlights the search bar. Below the search bar, there are buttons for "KQL" and "Refresh".

On the left side, there is a sidebar titled "Available fields" which lists various document fields such as @timestamp, _id, _index, _score, _type, _version, agent, ecs, event, http, monitor, and url.

The main area displays a histogram with 90 hits, spanning from April 18, 2022, 13h to April 18, 2022, 13:30:06.455. The histogram bars are teal-colored. Below the histogram, a table titled "1 field sorted" shows the results. The table has columns for "Document" and "@timestamp". The results list multiple entries for April 18, 2022, at various times, all with the same URL and timestamp. The table includes a "Rows per page" dropdown set to 100 and navigation arrows.

特定の値をフィルター



The screenshot shows the "Edit filter" dialog box. At the top right is a blue button labeled "+ Add filter" with a pink circle around it. The dialog has sections for "Field" (set to "url.domain") and "Operator" (set to "is"). The "Value" field contains "elasticsearch". There is also a "Create custom label?" checkbox. At the bottom right are "Cancel" and "Save" buttons.

フィルターの管理

Search KQL Last 15 minutes Refresh

url.domain: elasticsearch + Add filter

Pin across all apps Edit filter Exclude results Temporarily disable Delete

90 hits Documents Field statistics BETA

Apr 18, 2022 @ 13:18:46.698 - Apr 18, 2022 @ 13:33:46.698

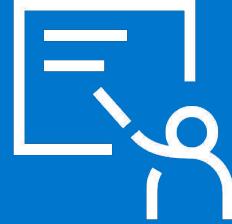
1 field sorted

@timestamp	Document
Apr 18, 2022 @ 13:33:41.301	url.domain: elasticsearch @timestamp: Apr 18, 2022 @ 13:33:41.301 agent.ephemeral_id: 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname: ip-172-31-12-220 agent.id: 7f7b0798-3bad-445f-be09-ed85d1e70d10 agent.name: ip-172-31-12-220 agent.type: heartbeat agent.version: 8.2.0 ecs.version: 8.0.0 event.dataset: http http.response.body.bytes: 539 http.response.bo...
Apr 18, 2022 @ 13:33:31.300	url.domain: elasticsearch @timestamp: Apr 18, 2022 @ 13:33:31.300 agent.ephemeral_id: 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname: ip-172-31-12-220 agent.id: 7f7b0798-3bad-445f-be09-ed85d1e70d10 agent.name: ip-172-31-12-220 agent.type: heartbeat agent.version: 8.2.0 ecs.version: 8.0.0 event.dataset: http http.response.body.bytes: 539 http.response.bo...
Apr 18, 2022 @ 13:33:21.299	url.domain: elasticsearch @timestamp: Apr 18, 2022 @ 13:33:21.299 agent.ephemeral_id: 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname: ip-172-31-12-220 agent.id: 7f7b0798-3bad-445f-be09-ed85d1e70d10 agent.name: ip-172-31-12-220 agent.type: heartbeat agent.version: 8.2.0 ecs.version: 8.0.0 event.dataset: http http.response.body.bytes: 539 http.response.bo...
Apr 18, 2022 @ 13:33:11.298	url.domain: elasticsearch @timestamp: Apr 18, 2022 @ 13:33:11.298 agent.ephemeral_id: 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname: ip-172-31-12-220 agent.id: 7f7b0798-3bad-445f-be09-ed85d1e70d10 agent.name: ip-172-31-12-220 agent.type: heartbeat agent.version: 8.2.0 ecs.version: 8.0.0 event.dataset: http http.response.body.bytes: 539 http.response.bo...
Apr 18, 2022 @ 13:33:01.297	url.domain: elasticsearch @timestamp: Apr 18, 2022 @ 13:33:01.297 agent.ephemeral_id: 2d974062-870d-40b4-a44f-b9e2aa926648 agent.hostname: ip-172-31-12-220 agent.id: 7f7b0798-3bad-445f-be09-ed85d1e70d10 agent.name: ip-172-31-12-220 agent.type: heartbeat agent.version: 8.2.0 ecs.version: 8.0.0 event.dataset: http http.response.body.bytes: 539 http.response.bo...

Rows per page: 100 < 1 >

Summary: Discover

Module 1 Lesson 3



Summary

- **Discover** はデータを理解するために、最初に Kibana で利用するツール
- **Discover** どんなデータが収集できているか (あるいはできていないか) を明らかにする
- **Discover** では Elasticsearch のデータ探索のために Data view が必要

Quiz

1. **Discover** が Observability データを扱う際に役立つことを3つあげましょう
2. **True or False: Discover** は Elasticsearch のデータを探索するために Data view が必要
3. **True or False: Discover** では特定のフィールド値でフィルターが可能

Discover

Lab 1.3

Discover で Heartbeat データを探索しましょ
う



Agenda

- **Module 1: Getting started**
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理

ログとメトリックの収集

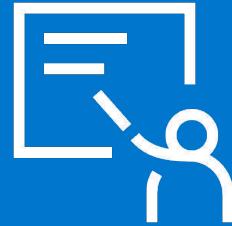
Module 2

Topics

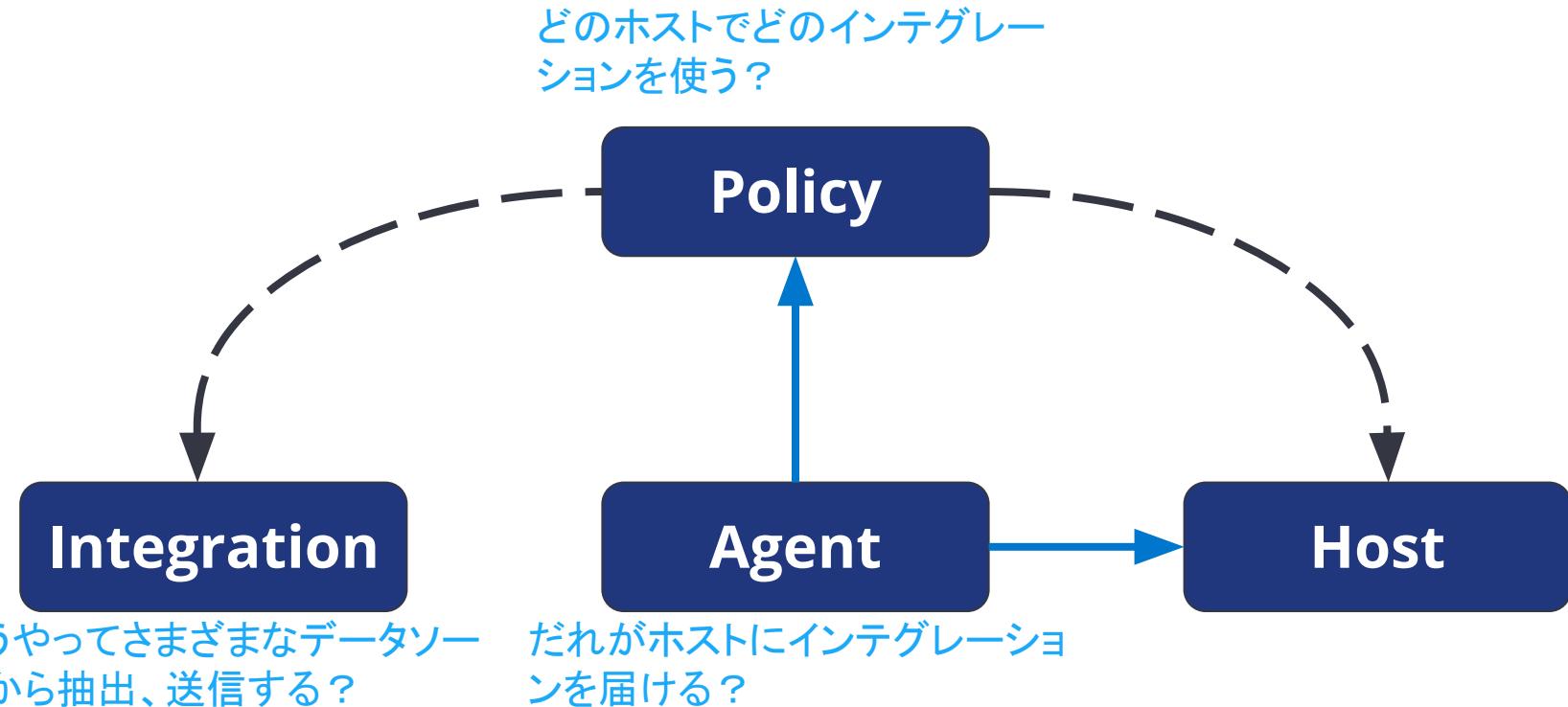
- Elastic Agent
- Logs
- Metrics

Elastic Agent

Module 2 Lesson 1



Fleet と Elastic Agent



Integrations

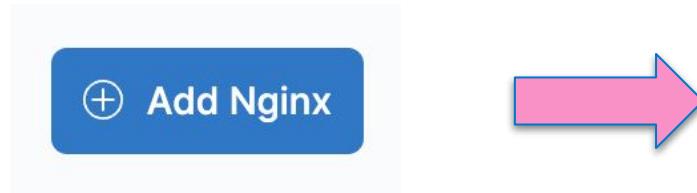
- Elastic を外部のサービスやシステムと接続する
- 迅速なインサイトの取得、アクションの実行
- 新しいデータソースを収集
- 多くはアセットを同梱
 - ダッシュボード
 - ビジュアリゼーション
 - パイプライン

Kibana からインテグレーションを追加、管理

The screenshot shows the Kibana Integrations interface. At the top, there's a navigation bar with 'Integrations' and a breadcrumb 'Browse integrations'. Below it is a main heading 'Integrations' with a sub-instruction 'Choose an integration to start collecting and analyzing your data.' There are two tabs: 'Browse integrations' (selected) and 'Installed integrations'. The interface is divided into several sections. On the left, there's a sidebar with 'All categories' (252) and a search bar. The main area contains cards for specific integrations: 'Web site crawler' (Add search to your website with the App Search web crawler.), 'Elastic APM' (Monitor, detect and diagnose complex performance issues from your application.), and 'Endpoint Security' (Protect your hosts with threat prevention, detection, and deep security data visibility.). Below these are more detailed cards for various services like '1Password Events Reporting', 'AbuseCH', 'ActiveMQ Logs', etc.

Elastic Agent ポリシー

- どのインテグレーションをどのホストで実行するか指定
- Elastic Agent ポリシーは複数のエージェントに設定できる
- 大規模な設定も簡単に管理できる



Configure an integration for the selected agent policy.

1 Configure integration

Integration settings

Choose a name and description to help identify how this integration will be used.

Integration name: nginx-1
Description: Optional

[Advanced options](#)

Collect logs from Nginx instances

Collect logs from third-party REST API (experimental)

Collect metrics from Nginx instances

2 Where to add this integration?

New hosts Existing hosts

Agent policy

Agent policies are used to manage a group of integrations across a set of agents.

Agent policy: Agent policy 1

1 agent is enrolled with the selected agent policy.

Elastic Package Registry

- Kibana で利用できる Elastic Agent インテグレーション向けのオンラインパッケージホスティングサービス
- Kibana が EPR に接続、最新のインテグレーションをダウンロード、アセットを Elasticsearch に保存
- インテグレーションは定期的に更新、リリースされるので、このプロセスはインターネット接続が必要

Fleet

- Fleet は Elastic Agent へのコミュニケーションチャネルとして働く
- Agent は定期的に最新の更新をチェック
- 各 Agent ポリシーには Agent を何台でも登録することができる

< View all agent policies

Agent policy 1

Revision 7 Integrations 5 Agents 1 agent Last updated on Apr 26, 2022 Actions ▾

Integrations Settings

Search... Namespace ▾ Add integration

Name ↑	Integration	Namespace	Actions
docker-1	Docker Metrics v1.2.0	default	...
log-1	Custom Logs v1.0.0	default	...
mysql-1	MySQL v1.2.1	default	...
nginx-1	Nginx v1.3.1	default	...
system-1	System v1.6.4	default	...

Fleet での中央集中管理

- Fleet ではすべての Elastic Agent のステータスを確認できる

Fleet

Centralized management for Elastic Agents.

Agents Agent policies Enrollment tokens Data streams Settings

Search Status Agent policy 2 Upgrade available Add agent

Showing 2 agents

Host	Status	Agent policy	Version	Last activity	Actions
ip-172-31-12-220	Healthy	Agent policy 1 rev. 7	8.2.0	25 seconds ago	...
3d94d8c87fa7	Healthy	Elastic Cloud agent policy rev. 4	8.2.0	25 seconds ago	...

Rows per page: 20 < 1 >

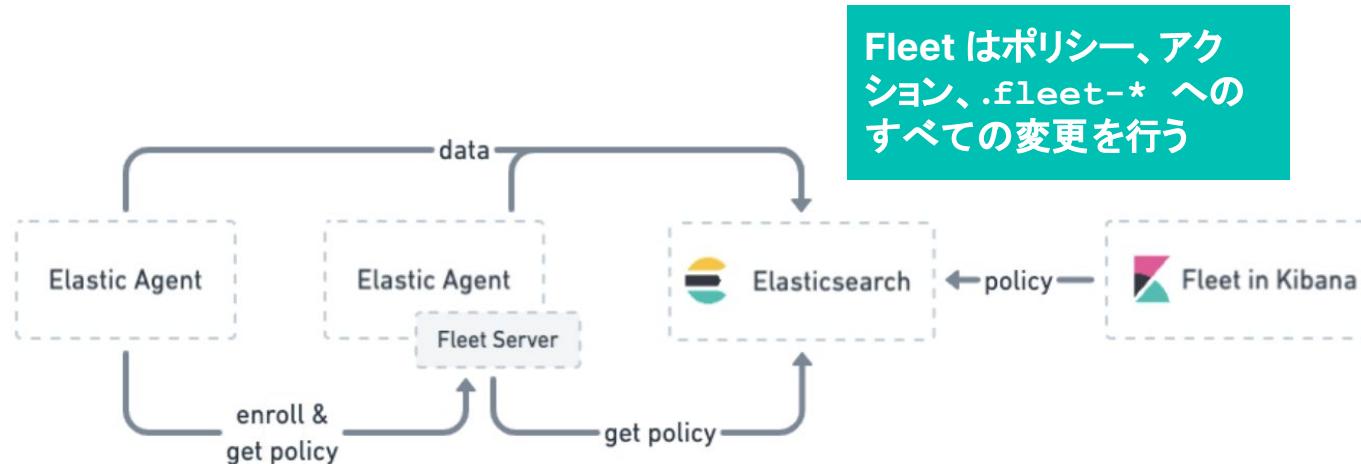
Kibana で Elastic Agent と設定されたポリシーを管理

設定とアップグレード

- Agent ポリシーの設定を変更するには
 - すべての agent は次のチェックイン時に更新を受信する
 - ポリシーの更新を手動で分配する必要はない
- Elastic Agent のバイナリ、インテグレーションを更新するには
 - Fleet からアップグレードを開始
 - すると各ホスト上で稼働している Elastic Agent が自動的にアップグレードされる

Fleet Server

- Elastic Agents を Fleet へつなげる
- スケーラブルなインフラを可能に
- Elastic Cloud、セルフマネージドで利用可能
- デプロイされた Elastic Agent と通信する独立したプロセス



Elastic Agents のインストール

- Elastic Agent のインストール、管理にはいくつか方法がある
 - Fleet-managed の Elastic Agent をインストール
 - スタンドアロンの Elastic Agent をインストール
 - コンテナライズされた環境に Elastic Agent をインストール

Fleet-managed の Elastic Agent をインストール

- これが推奨される手法
- エージェントの管理、アップグレードが簡単
- モニタリングしたい各ホスト上に Elastic Agent をインストール
- Kibana の Fleet を使ってエージェントを設定、集中管理
- トレーニングではこの手法を利用

スタンドアロンの **Elastic Agent** をインストール

- Fleet でエージェントを集中管理するメリットがない場合は便利
 - 会社のセキュリティ要件
 - 他の構成管理システムを利用したい
- このアプローチは上級者向け
- エージェントの管理、アップグレードは自分で実施する必要がある
- 監視対象の各ホストに Elastic Agent をインストール
- インストールした場所でローカルに、手動でエージェントを設定

コンテナライズ環境に **Elastic Agent** をインストール

- コンテナの内部で Elastic Agent を動作させることができる
- Fleet Server、スタンドアロンモードに対応

Getting started

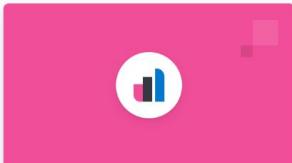
≡ P Home

Welcome home



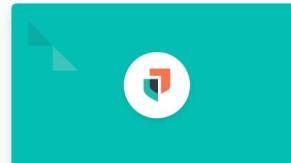
Enterprise Search

Create search experiences with a refined set of APIs and tools.



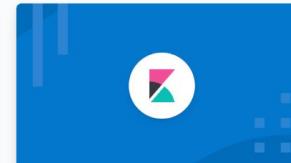
Observability

Consolidate your logs, metrics, application traces, and system availability with purpose-built UIs.



Security

Prevent, collect, detect, and respond to threats for unified protection across your infrastructure.



Analytics

Explore, visualize, and analyze your data using a powerful suite of analytical tools and applications.

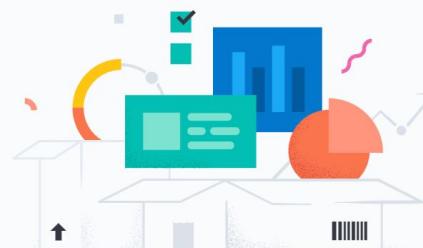
Get started by adding integrations

To start working with your data, use one of our many ingest options. Collect data from an app or service, or upload a file. If you're not ready to use your own data, add a sample data set.

[+ Add integrations](#)

[Try sample data](#)

[Upload a file](#)



最初のインテグレーションを追加

The screenshot shows the Elasticsearch interface for managing integrations. On the left, a search bar highlights the term "nginx". On the right, the "Nginx" integration details page is displayed, featuring its logo, version 1.3.1, and a prominent blue button labeled "+ Add Nginx" which is circled in pink.

Integrations

Choose an integration to start collecting and analyzing your data.

Browse integrations Installed integrations

Web site crawler

Add search to your website with the App Search web crawler.

All categories 256

AWS 25

Azure 23

Cloud 38

Communications 3

Config management 2

Containers 13

Custom 22

Elast

Monitor, detect a performance issue

nginx

Nginx

Elastic Agent

Version 1.3.1

+ Add Nginx

Overview Settings

Nginx Integration

This integration periodically fetches metrics from Nginx servers. It can parse access and error logs created by the HTTP server.

Compatibility

The Nginx stubstatus metrics was tested with Nginx 1.19.5 and are expected to work with all version ≥ 1.9 . The logs were tested with version 1.19.5. On Windows, the module was tested with Nginx installed from the Chocolatey repository.

Logs

Timezone support

This datasource parses logs that don't contain timezone information. For these logs, the Elastic Agent reads the local timezone and uses it when parsing to convert the timestamp to UTC. The timezone to be used for parsing is included in the event in the event.timezone field.

To disable this conversion, the event.timezone field can be removed with the drop_fields processor.

Screenshots

Details

Version 1.3.1

Category Security, Web

Kibana assets

- Dashboards 3
- ML modules 1
- Saved searches 3
- Visualizations 19

インテグレーションを設定し、ポリシーを選択

The screenshot shows the 'Add Nginx integration' wizard. The top navigation bar includes 'Integrations > Nginx > Add integration'. The main title is 'Add Nginx integration' with a subtitle 'Configure an integration for the selected agent policy.' Below this, the first step is titled 'Configure integration'.

Integration settings
Choose a name and description to help identify how this integration will be used.

Integration name: nginx-1
Description: Optional

[Advanced options](#)

Configuration options:

- Collect logs from Nginx instances
- Collect logs from third-party REST API (experimental)
- Collect metrics from Nginx instances

Where to add this integration?

Create agent policy
Add this integration to a new set of hosts by creating a new agent policy. You can add agent in the next step.

New agent policy name: Agent policy 1
 Collect system logs and metrics

[Advanced options](#)

At the bottom right, there are 'Cancel' and 'Save and continue' buttons. The 'Save and continue' button is highlighted with a pink oval.

ホストに Elastic Agent を追加

Nginx integration added

To complete this integration, add **Elastic Agent** to your hosts to collect data and send it to Elastic Stack.

Add Elastic Agent later

Add Elastic Agent to your hosts

Add agent

Add Elastic Agents to your hosts to collect data and send it to the Elastic Stack.

[Enroll in Fleet](#) [Run standalone](#)

Enroll an Elastic Agent in Fleet to automatically deploy updates and centrally manage the agent.

1 Select enrollment token

Agent policy 1 has been selected. Select which enrollment token to use when enrolling agents.

Authentication settings

Enrollment token Default (dfac5e7e-9695-43a7-8c6e-59ac67b4fb16)

2 Install Elastic Agent on your host

Select the appropriate platform and run commands to install, enroll, and start Elastic Agent. Reuse commands to set up agents on more than one host. For aarch64, see our [downloads page](#). For additional guidance, see our

[Close](#)

ホストに Elastic Agent を追加

Add agent

Add Elastic Agents to your hosts to collect data and send it to the Elastic Stack.

[Enroll in Fleet](#) [Run standalone](#)

2 Install Elastic Agent on your host

Select the appropriate platform and run commands to install, enroll, and start Elastic Agent. Reuse commands to set up agents on more than one host. For aarch64, see our [downloads page](#). For additional guidance, see our [installation docs](#).

[Linux Tar](#) [Mac](#) [Windows](#) [RPM](#) [DEB](#)

```
curl -L -0 https://artifacts.elastic.co/downloads/beats/elastic-tar  
tar xzvf elastic-agent-8.2.0-linux-x86_64.tar.gz  
cd elastic-agent-8.2.0-linux-x86_64  
sudo ./elastic-agent install --url=https://e3dbc640d4094d1f8ff2:12345
```

3 Confirm agent enrollment

If you are having trouble connecting, see our [troubleshooting guide](#).

[Close](#)

Add agent

Add Elastic Agents to your hosts to collect data and send it to the Elastic Stack.

[Enroll in Fleet](#) [Run standalone](#)

✓ Confirm agent enrollment

✓ 1 agent has been enrolled.

[View enrolled agents](#)

✓ Incoming data confirmed

✓ Incoming data received from 1 of 1 recently enrolled agent.

Next, analyze your data using our integration assets such as curated views, dashboards and more.

[View assets](#)

[Close](#)

アセットを確認

≡ D Integrations Nginx View deployment details

◀ Browse all integrations



Nginx

Elastic Agent

Version 1.3.1 | Agent policies 1 | [+ Add Nginx](#)

Overview Integration policies **Assets** Settings

▼ Dashboards 3

- [Metrics Nginx] Overview
Overview dashboard for the Nginx integration in Metrics
- [Logs Nginx] Access and error logs
Dashboard for the Logs Nginx integration
- [Logs Nginx] Overview
Dashboard for the Logs Nginx integration

▶ Saved searches 3

▶ Visualizations 19

Elastic Agent のトラブルシュート

- エージェントのトラブルシュート用コマンドがいくつがある
- Elastic Agent バージョンの取得

```
elastic-agent version
```

Elastic のデプロイとバージョン
が一致すべき

- Elastic Agent ステータスをチェック

```
elastic-agent status
```

ステータスが健全でなければ何
か対応が必要

- 現在の Elastic Agent 設定を確認

```
elastic-agent inspect
```

エージェントの設定が正しいか確
認

- Elastic Agent 診断情報を収集

```
elastic-agent diagnostics collect
```

診断情報の zip ファイルを生成

Summary: **Elastic Agent**

Module 2 Lesson 1



Summary

- Elastic Agent はホストのログ、メトリックなどのデータをモニタリングするための、単一の統合された手法
- Elastic Agent で利用するインテグレーションは外部のサービスやシステムと Elastic を簡単につなげる仕組みを提供
- Agent ポリシーはどのインテグレーションをどのホストで実行するかを指定する
- Fleet を使えば Elastic Agent を集中管理し、より迅速に、より簡単にデータを登録できる

Quiz

1. **True or False:** Elastic Agent を使う場合、デプロイが必要なエージェント数は収集するデータによって変動する
2. **True or False:** Elastic Agent インテグレーションはポピュラーなアプリ、サービスからのデータ収集を行う簡単で、統合された方法を提供
3. **True or False:** Elastic Agent のポリシーは、単一ホスト上で複数のインテグレーションを実行できる

Elastic Agent

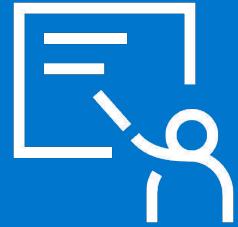
Lab 2.1

ホストに Elastic Agent をインストールしましょ
う



Logs

Module 2 Lesson 2



ビジネスに関する疑問

- 新しいトレーニングのランディングページに何人訪れた?
- なぜこの JavaScript アプリは遅いのか?
- ダウンロードサービスのメンテナンスはいつ予定すべき?
- ヨーロッパからどれだけのウェビナー登録があったか?

ログとは？

- ログはアクティビティの記録
 - システムによる
 - アプリケーションによる
 - デバイスによる
 - 人間による
 - ...
- **Timestamp + data**

ログとは？

- アプリケーションログ

```
66.249.73.185 - - [16/Feb/2014:09:47:54 -0500] "GET / HTTP/1.1" 200 37932 "-"  
"Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)"
```

Web logs

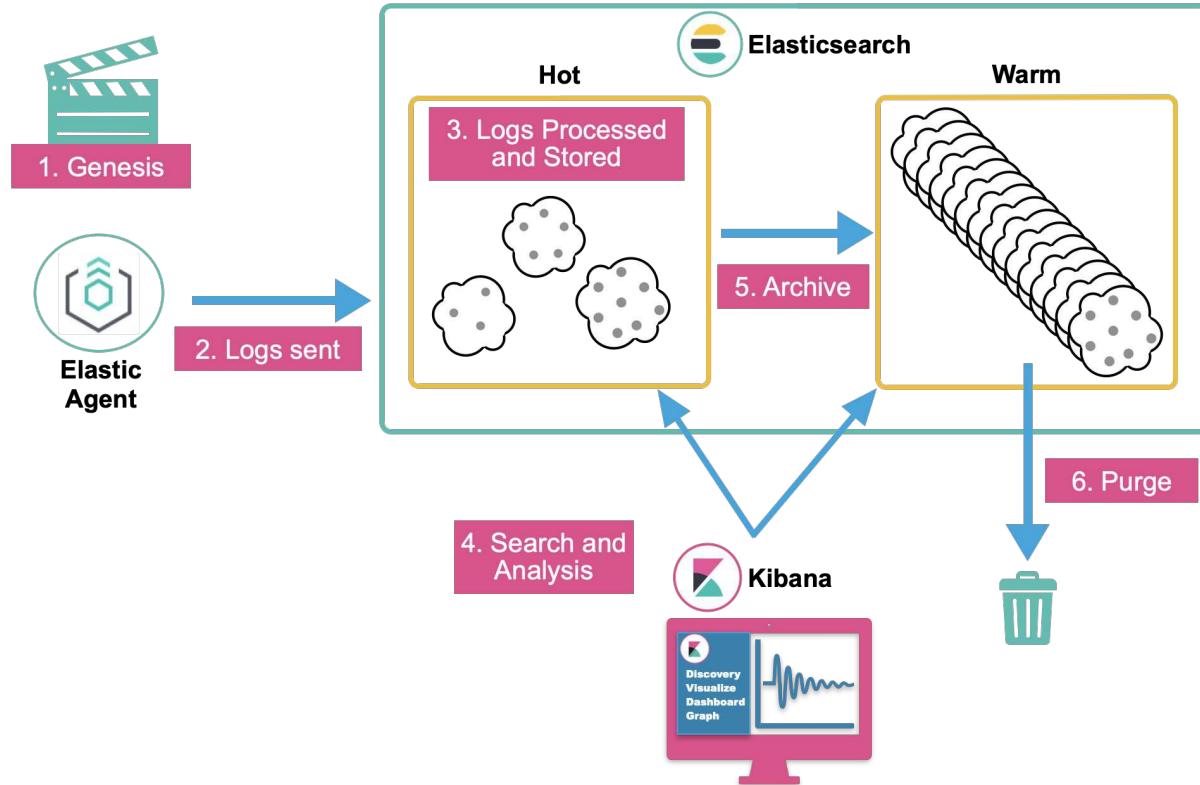
```
[2017-05-18 00:00:05,871][INFO ][cluster.metadata] [esnData-2]  
[.data-es-1-2017.05.18] creating index, cause [auto(bulk api)], templates  
[.data-es-1], shards [1]/[1], mappings [_default_, shards, node, index_stats,  
index_recovery, cluster_state, cluster_stats, node_stats, indices_stats]
```

Elasticsearch logs

```
120707 0:37:09 [Note] Plugin 'FEDERATED' is disabled.  
120707 0:37:09 InnoDB: The InnoDB memory heap is disabled
```

SQL logs

ログのライフサイクル



タイムスタンプについての補足

- タイムスタンプの扱いは難しい
 - 多くのフォーマットがある
 - タイムゾーンはやっかい
- Elasticsearch は **ISO 8601** フォーマットを好む
 - 正しく設定すれば他の時間フォーマットでもインジェスト可能
- **ISO 8601** は誤解を避けるために設計された
 - タイムスタンプを **ISO 8601** 形式で表現するのは良い習慣
- UI アプリケーション (e.g. Kibana) で表示時間を調整できる
 - 保存された時間をユーザーのローカル時間で表示
- 例: "2018-10-05T14:30:00Z"

よくある問題

- 一貫性
 - それぞれのアプリケーション、デバイスのログ形式がばらばら
- フォーマット
 - "Oct 11 20:21:47", "020805 13:51:24"
- 散らばっている
 - いろいろなサーバーにログが分散している
 - SSH + grep はスケールしない
- エキスパートが必要
 - サーバー上のログへのアクセスは制限されている
 - ログフォーマットの限定的な知識

Logs integrations

- Elastic Agent インテグレーションは簡単に一般的な形式のログを収集、パース、可視化
- ログとメトリックの両方をデフォルトで収集するインテグレーションもある

logs

X

ActiveMQ Logs
Collect and parse logs from ActiveMQ instances with Filebeat.

Apache HTTP Server
Collect logs and metrics from Apache servers with Elastic Agent.

Apache Tomcat
Collect and parse logs from Apache Tomcat servers with Elastic Agent.

Arbor Peakflow Logs
Collect and parse logs from Netscout Arbor Peakflow SP with Filebeat.

Atlassian Bitbucket
Collect logs from Atlassian Bitbucket with Elastic Agent.

Atlassian Confluence
Collect logs from Atlassian Confluence with Elastic Agent.

Atlassian Jira
Collect logs from Atlassian Jira with Elastic Agent.

Auditd
Collect logs from Linux audit daemon with Elastic Agent.

Auth0 Log Streams Integration
Collect logs from Auth0 with Elastic Agent.
Technical preview

AWS
Collect logs and metrics from Amazon Web Services with Elastic Agent.

AWS CloudFront
Collect logs from AWS CloudFront with Elastic Agent

AWS Cloudtrail Logs
Collect and parse logs from AWS Cloudtrail with Elastic Agent

Logs integrations の例

- ログファイルのデフォルトのパスが設定されている
- ログの行をパースするパイプラインの定義を含む
- 正しいデータ型を持った各フィールドの定義がある
- ログファイルを可視化するためのサンプルダッシュボードを提供

The screenshot shows the Elasticsearch Logstash configuration interface. It displays two main sections for collecting MySQL logs:

- Collect logs from MySQL hosts**:
 - MySQL error logs**:
 - Collect MySQL error logs
 - Error log paths:
 - /var/log/mysql/error.log*
 - /var/log/mysqld.log*

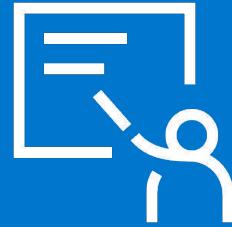
[Add row](#)

 - MySQL slowlog logs (log)**:
 - Collect MySQL slowlog logs using log input
 - Slowlog paths:
 - /var/log/mysql/*-slow.log*
 - /var/lib/mysql/*-slow.log*

[Add row](#)

Summary: Logs

Module 2 Lesson 2



Summary

- ログはデータに関する多くの疑問に対する答えを提供してくれる
- ログのメッセージはタイムスタンプとデータで構成される
- インテグレーションで一般的なログフォーマットの収集、パース、可視化を簡単に実現できる
- Elastic Agent はログディレクトリや特定のログファイルを監視
- Elasticsearch にデータが送信されたら、Elasticsearch をクエリしてデータの探索が可能

Quiz

1. ログメッセージを構成する主要な二つの要素は?
2. **True or False:** マーケティングチームの Scott に全ての Web サーバログファイルを渡して、最後に Scott が実施したウェビナーの登録者数を数えてもらうのは問題ない
3. **True or False:** Elastic Agent でデータの収集と Elastic プラットフォームへのデータ送信を簡単に実現できる

Logs

Lab 2.2

ログを投入し、探索しましょう



Metrics

Module 2 Lesson 3



Monitoring

- システムやサービスは大量のデータを生成する
- And these data should be
 - これらのデータは
 - 保存して
 - 分析して
 - 監視すべき

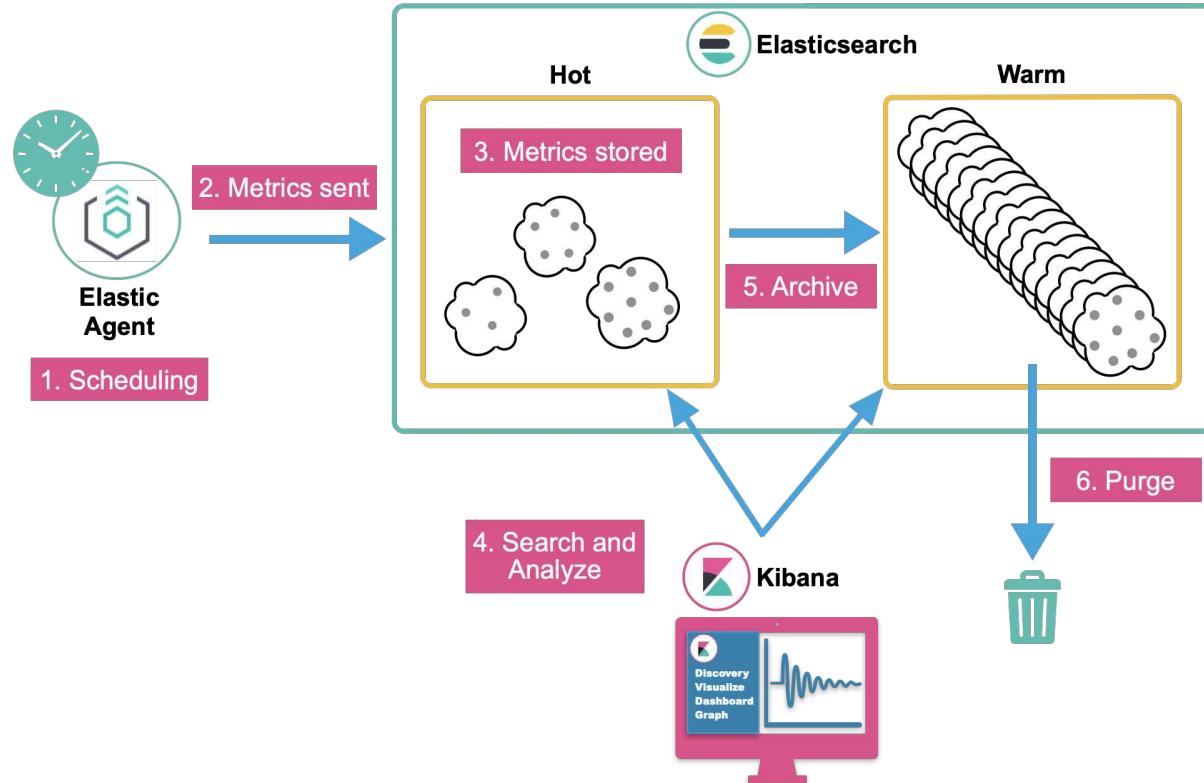
モニタリングの例



ホストの
ログ & メトリック

Elasticsearch の
ログ & メトリック

メトリックのライフサイクル



ログ vs メトリック

- ログとメトリックにはいくつかの共通点がある
 - どちらも時系列データ
 - どちらも複数のキーワードを持つ
- しかしこれらはまた、根本的に異なる
 - メトリックはシステムの定期的な観測にフォーカス
 - メトリックは数値とキーワードにフォーカス
 - ログは発生したイベントの説明（いつ、何が）
 - 通常ログにはパースが必要なテキストが含まれる

ログ vs メトリックの例

- イベントが起こるたびにログが記録される

```
[2018-09-07T07:48:00,127][INFO ][o.e.x.m.MlDailyMaintenanceService]
triggering scheduled [ML] maintenance tasks
[2018-09-07T07:48:00,381][INFO ][o.e.x.m.a.TransportDeleteExpiredDataAction]
[_8LMCWq] Deleting expired data
[2018-09-07T07:48:00,648][INFO ][o.e.x.m.MlDailyMaintenanceService]
Successfully completed [ML] maintenance tasks
```

- メトリックは定期的に記録される schedule

```
[2018-09-07T06:00:00,000][filesystem] 50085941248 overlay / 67371577344
[2018-09-07T06:05:00,000][filesystem] 50085917352 overlay / 67371577344
[2018-09-07T06:10:00,000][filesystem] 50075903715 overlay / 67371577344
```

タイムスタンプ

- Elasticsearch は ISO 8601 を好む
- タイムスタンプにはタイムゾーンが含まれる
- Kibana はユーザーのローカルタイムゾーンを利用する

2018-09-07T06:10:00

タイムゾーンなし

2018-09-07 06:10:00 -0400

New York
タイムゾーン

Metrics integrations

- オペレーティングシステムやサーバー上で稼働しているサービスからのメトリック収集に役立つ
- デフォルトでログとメトリックの両方を収集するインテグレーションもある

The screenshot shows a search bar at the top with the word "metrics" highlighted by a pink oval. Below the search bar is a grid of 12 cards, each representing a different metric integration:

- ActiveMQ Metrics**: Collect metrics from ActiveMQ instances with Metricbeat.
- Aerospike Metrics**: Collect metrics from Aerospike servers with Metricbeat.
- Apache HTTP Server**: Collect logs and metrics from Apache servers with Elastic Agent.
- APM**: Collect performance metrics from your applications with Elastic APM.
- AWS**: Collect logs and metrics from Amazon Web Services with Elastic Agent.
- AWS Billing Metrics**: Collect billing metrics from Amazon Web Services with Elastic Agent.
- AWS CloudWatch**: Collect logs and metrics from Amazon CloudWatch with Elastic Agent.
- AWS DynamoDB Metrics**: Collect metrics from Amazon DynamoDB service with Elastic Agent.
- AWS EBS Metrics**: Collect metrics from Amazon Elastic Block Storage service with Elastic Agent.
- AWS EC2**: Collect logs and metrics from Amazon Elastic Compute Cloud service with Elastic Agent.
- AWS ELB**: Collect logs and metrics from Amazon Elastic Load Balancing service with Elastic Agent.
- AWS Fargate**: Collects metrics from containers and tasks running on Amazon ECS clusters with Elastic Agent. (Beta)

Metrics integrations の例

- 特定のサービスからデータを収集する
基本的な仕組みを持つ
- サービスへの接続方法を指定
- メトリックの収集頻度を指定
- 複数のメトリックセットを指定可能

The screenshot shows the Elasticsearch Metrics integrations configuration interface. It displays three separate sections, each with a toggle switch labeled "Collect [metric] metrics".

- Docker metrics:** Hosts: unix:///var/run/docker.sock, Period: 10s, De-Dot labels: checked (If True, remove dot notation on container labels).
- Docker cpu metrics:** Hosts: unix:///var/run/docker.sock, Period: 10s, De-Dot labels: checked (If True, remove dot notation on container labels).
- Docker diskio metrics:** Hosts: unix:///var/run/docker.sock, Period: 10s, De-Dot labels: checked (If True, remove dot notation on container labels).

Summary: Metrics

Module 2 Lesson 3



Summary

- メトリックとログは重要なオブザーバビリティデータを提供
- ログは何が、いつ起きたかに関する情報
- メトリックは定期的に収集された特定の情報
- Elastic Agent は複数のメトリックをシステムやサービスから収集できる
- データが Elasticsearch に送信されたら、Elasticsearch をクエリしてデータを探索することができる

Quiz

1. **True or False:** メトリックは数値しか保持しない
2. **True or False:** ログとメトリックの主な違いは、ログは発生した事象を説明するのに対し、メトリックは定期的なシステムの計測にフォーカスしている点である
3. **True or False:** Elastic オブザーバビリティを導入する際、ISO 8601 形式に従いタイムゾーン情報を持ったタイムスタンプを使うのは良い慣習である

Metrics

Lab 2.3

メトリックを投入し、探索しましょう



Agenda

- **Module 1: Getting started**
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理

APM データの収集

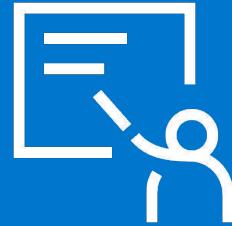
Module 3

Topics

- Elastic APM
- Java agent
- Node.js agent
- RUM agent

Elastic APM

Module 3 Lesson 1



Elastic APM

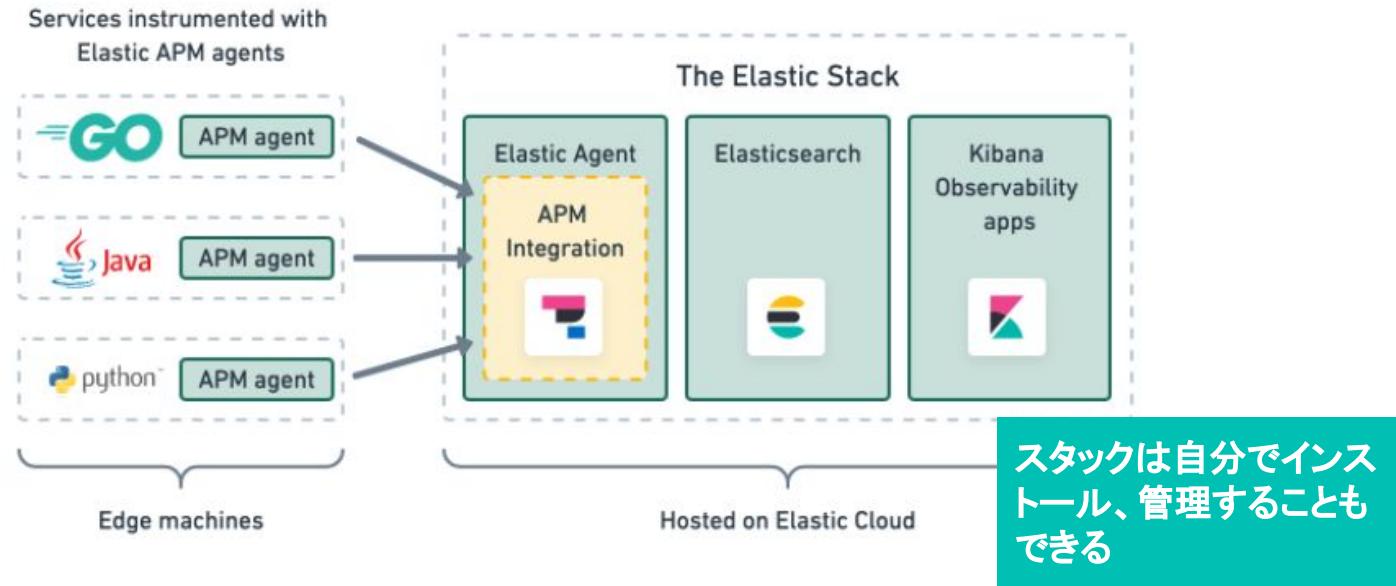
- Elastic プラットフォーム上に構築された APM システム
- リアルタイムにソフトウェアサービスを監視することができる
 - 応答時間に関する詳細な性能情報を収集
 - 例、データベースクエリ、キャッシュ呼び出し、HTTP リクエストなど
- 自動的に処理されなかったエラーや例外を収集
 - エラーはスタックトレースに基づいてグループ化される
 - 新しく発生したエラー、発生回数を簡単に特定できる
- ホストレベル、およびエージェント固有のメトリックの監視に役立つ

Elastic APM のコンポーネント

- Elastic APM は 4つの要素で構成される
 - APM agents
 - Elastic APM integration
 - Elasticsearch
 - Kibana
- これら 4つのコンポーネントが連携動作するには二つの方法がある

Elastic APM アーキテクチャ

- ひとつは、エッジマシン上で APM エージェントを稼働させる方法
- そして中央の APM インテグレーションへとデータを送信する



Elastic APM もうひとつのアーキテクチャ

- APM をスケールさせる必要がある場合、APM agent と APM インテグレーションをエッジマシン上で稼働させる方法もある
- そして中央の Elastic Agent 経由で参加する

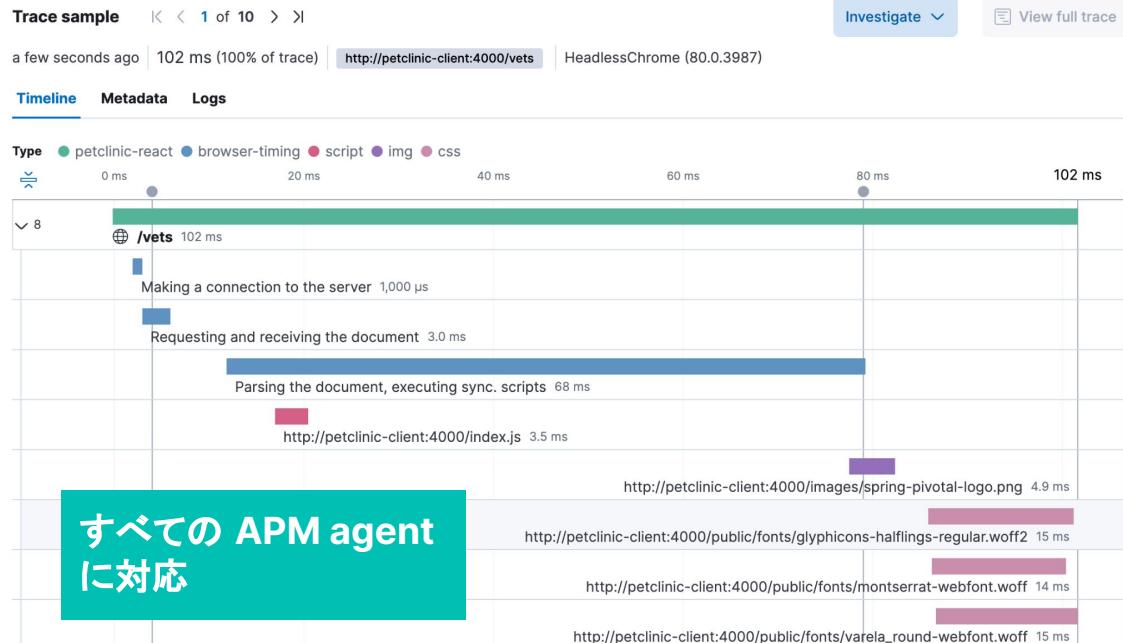


データモデル

- Elastic APM agent はいろいろな種別の **event** を捕獲
- Event には以下の種別がある
 - **Spans**: 実行された特定のコードパスに関する情報
 - **Transactions**: サービスやアプリケーションを計測している Elastic APM agent によって捕獲されたイベントの説明
 - **Errors**: マッチする例外やログメッセージで例外のグループを定義
 - **Metrics**: 基本的なホストレベル、およびエージェント固有のメトリックを提供

分散トレーシング

- トレースは共通の root を持つトランザクションとスパンのグループ
- 分散トレーシングにより 単一のビューでマイクロ サービスアーキテクチャを通した性能分析が可 能となる



分散トレーシングの仕組み

- Elastic APM はすべてのリクエストをトレース
 - フロントエンドサービスへの最初の web リクエストから
 - バックエンドサービスへ実行されたクエリまで
- リクエストヘッダがトレースのコンテキストで修飾される

```
traceparent:                      // header name
00-                                // version
0af7651916cd43dd8448eb211c80319c- // trace id
b7ad6b7169203331-                  // parent span id
01                                // flags
(sampling)
```

- Trace id がリクエスト間で伝搬
- 同じ trace id を持つそれぞれのサービスでのリクエストは単一のトレースオーバービューでつながる

Real User Monitoring (RUM)

- Web ブラウザーなど、クライアントとユーザーのやりとりをキャプチャ
 - JavaScript agent が RUM agent
 - APM インテグレーションで RUM エンドポイントの有効化が必要
- アプリケーションの有意義なインサイトを提供
 - クライアントサイドアプリの性能問題
 - サーバーサイドアプリのレイテンシ
 - 最も使われているブラウザ、デバイス、プラットフォームの特定
 - 位置情報を元とした Web サイトの地理的な性能、各ユーザーの性能

Backend agents vs. Frontend agents

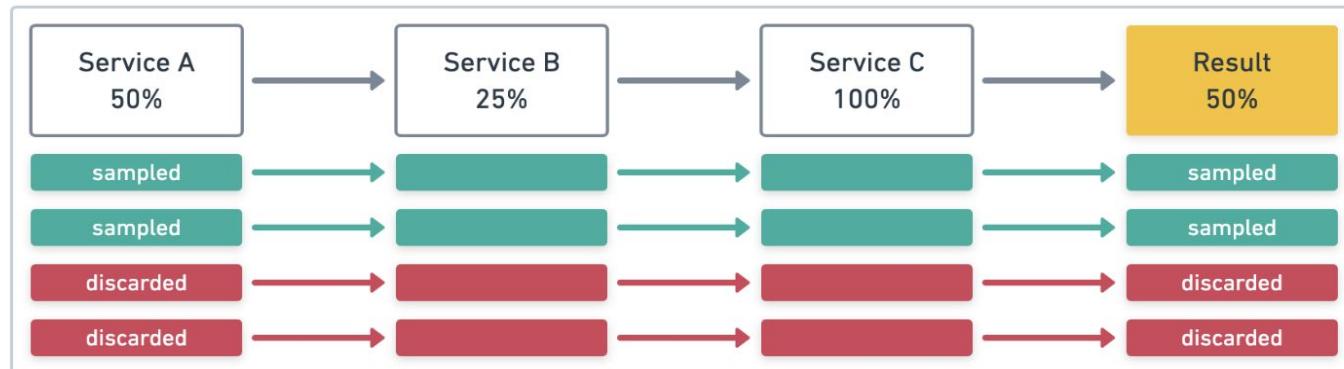
- Elastic APM backend agent はリクエストとレスポンスを監視
- RUM JavaScript agent はリアルなユーザー体験とクライアントサイドアプリのやりとりを監視
 - 例: **timeToFirstByte, domInteractive, domComplete**
- RUM JavaScript agent はフレームワークに依存しない
 - あらゆるフロントエンド JavaScript アプリで利用可能

サンプリング戦略

- 分散トレーシングは非常に大量なデータを生成するため、高コストでノイズも多
くなりがち
- 投入されるデータ量と、そのデータ分析に必要な労力を削減するためにサンプ
リングを行う
- Elastic APM では二種類のサンプリングに対応
 - **head-based** サンプリング
 - **tail-based** サンプリング

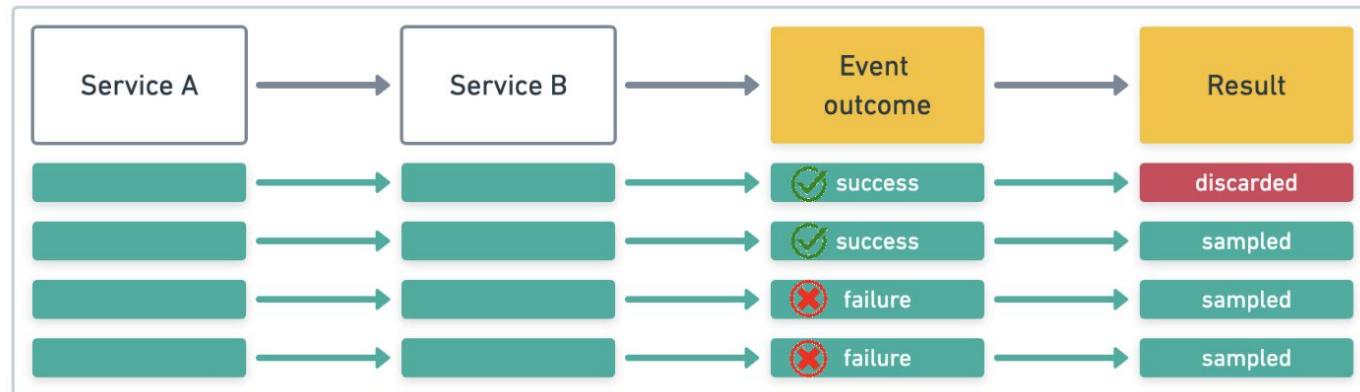
Head-based サンプリング

- デフォルトでは Elastic APM は **head-based** サンプリングを使う
 - トレースが開始される際にランダムサンプリングを決定、後続のサービスでは最初の決定に従う
 - 固定的なサンプルレート手法は小規模なアプリで効果的
 - 例: トランザクションの 50% を記録



Tail-based サンプリング

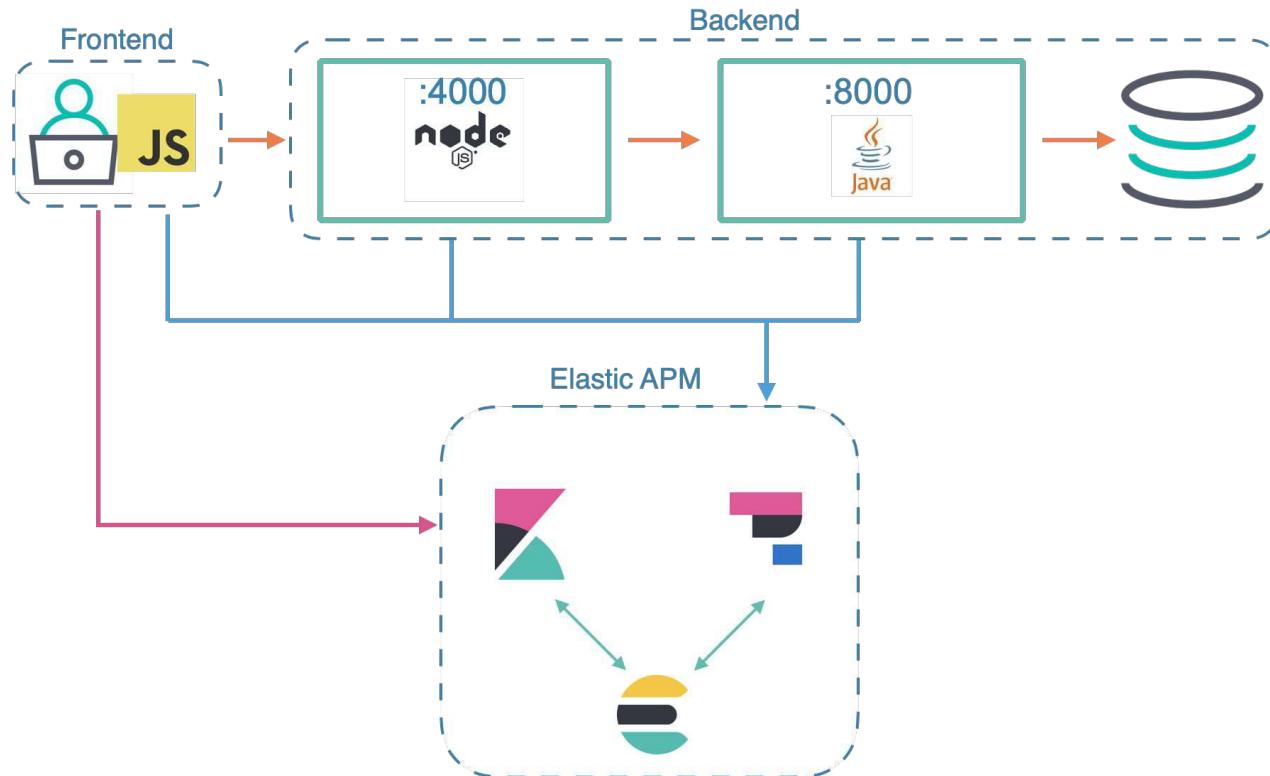
- **tail-based** サンプリングは APM integration の設定で有効化できる
 - サンプリングはトレースが完了した時点でいくつかのルールやポリシーに基づき決定する
 - 大規模なアプリできめ細やかなサンプリング制御が可能
 - 例: 正常リクエストの 50% と 100% の失敗リクエスト



サンプルレート

- 最適なサンプルレートは?
 - 残念ながら存在しない
- サンプリングはデータ、アプリケーションのスループット、データ保持ポリシーその他の要因に依存する
- しかし、トランザクションの 0.1 (すなわち 10%) を記録するようエージェントを設定するのは通常良い始め方

ラボアーキテクチャ



Summary: **Elastic APM**

Module 3 Lesson 1



Summary

- Elastic APM でソフトウェアサービスおよびアプリケーションをリアルタイムで監視できる
- Elastic APM は処理されないエラーや例外を自動的に収集する
- Elastic APM は 4つの部品で構成される: APM agents, APM integration, Elasticsearch, Kibana
- Elastic APM は分散トレーシングに対応、マイクロサービスアーキテクチャを通して単一のビューで性能の分析が可能
- **RUM RUM** は web ブラウザとユーザのやりとりをキャプチャし、web アプリの性能に関する詳細なリアルユーザー体験のビューを提供

Quiz

1. **True or False:** Real User Monitoring でマイクロサービスアーキテクチャ全体の性能を単一のビューで分析できる
2. Elastic APM を構成する 4つのコンポーネントは?
3. **True or False:** 運用コストを切り詰めるにはサンプリングレートを下げるといい

Elastic APM

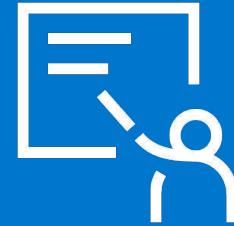
Lab 3.1

APM インテグレーションを触ってみましょう



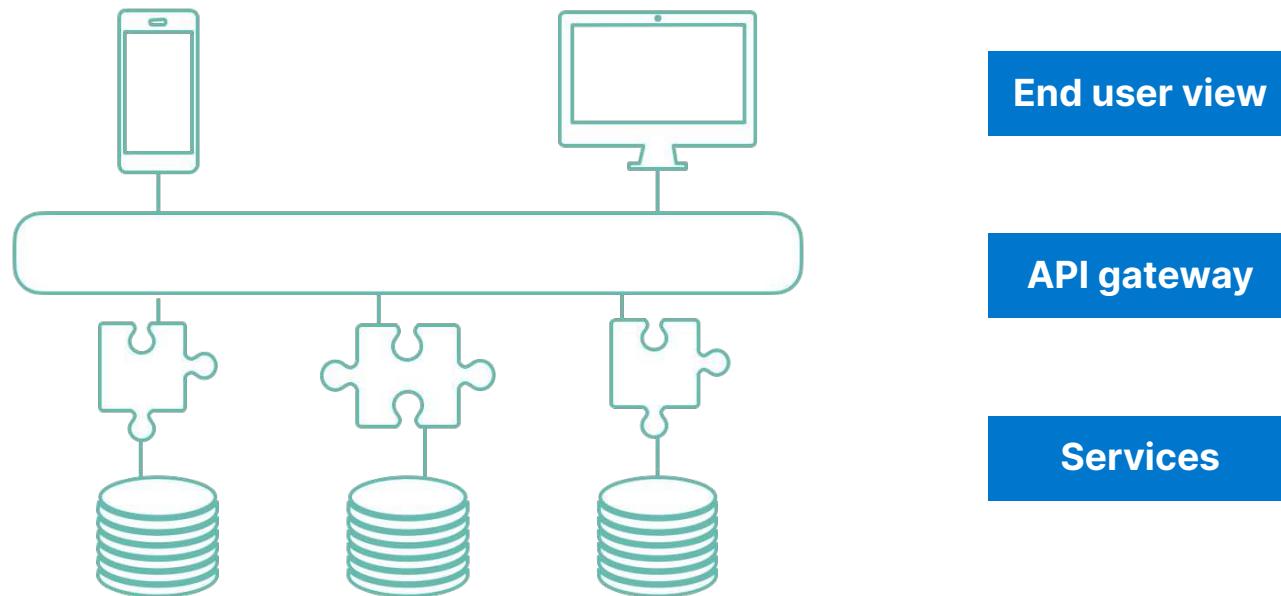
Java agent

Module 3 Lesson 2



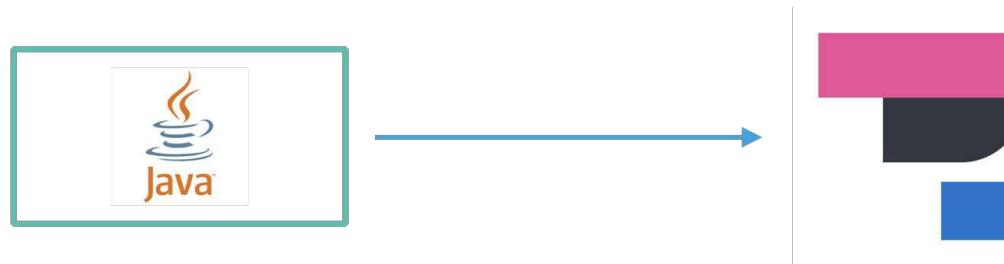
サービスレイヤー

- ミドルウェアレイヤーは共通のサービスを複数のアプリケーションに提供する



Java agent

- 自動的にアプリケーションの性能を計測
- 加えて、例外やエラーも捕獲
- ポピュラーなフレームワークや技術をビルトインサポート
- あらゆるアプリケーションを計測できるシンプルな API も提供



エージェントが動作する仕組み

- サポート対象のテクノロジを自動計測しイベントを記録
 - 例: データベースクエリ、HTTP リクエストなど
- エージェントはバイトコード計測を実施
 - 安全に小さなコードの断片をイベントの前後に注入し実行時間、メタデータや HTTP の情報を計測
- コードのリコンパイルは不要
- イベントを **transaction** や **span** といった形で Elastic APM に送信

サポート対象のテクノロジ

- Java versions
 - Oracle JDK, Open JDK, IBM J9 VM, HP-UX JVM, and SAP JVM
- Web Frameworks
 - Servlet API, Spring Web MVC, JavaServer, Spring Boot, ...
- Application Servers/Servlet Containers
 - Tomcat, WildFly, JBoss EAP, Jetty, WebSphere Liberty, ...
- その他の API も順次拡充中
 - data stores, networking frameworks, and much more...
 - www.elastic.co/guide/en/apm/agent/java/current/supported-technologies-details.html

エージェントのセットアップ

- Elastic APM Java Agent のセットアップ方法は 3種類
- マニュアル セットアップ
 - -javaagent フラグを使ってアプリケーションを起動
- 自動 セットアップ
 - APM agent standalone JAR を特定、もしくは全 JVM にアタッチ
- プログラマティック API セットアップ
 - APM agent アーティファクトへの依存関係を宣言
- 自動、およびプログラマティック API はこのトレーニングでは扱わない
 - インストール方法がまだベータのため (ver 1.3.1 まで)

マニュアルセットアップ

- 最新の Java agent **elastic-apm-agent-<version>.jar** ファイルを Maven Central からダウンロード
 - search.maven.org/search?q=g:co.elastic.apm%20AND%20a:elastic-apm-agent
 - アプリケーションではエージェントへの依存関係を定義する必要はない
- JVM フラグ **-javaagent** を追加してアプリを起動 app

```
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
```

アプリケーションサーバーでのエージェント設定

- アプリケーションサーバーでは **-javaagent** フラグとシステムプロパティを設定する
- **Embedded servers** (e.g., Spring Boot)
 - **-javaagent** JVM フラグで一般的な設定
 - **-D** プリフィックスでエージェントを設定
- **その他のアプリケーションサーバー** (例 Apache Tomcat)
 - 追加の設定が必要
 - www.elastic.co/guide/en/apm/agent/java/current/setup-javaagent.html

エージェントの設定

- Java エージェントの設定方法はいくつかある
- 以下は優先順位の高い順のリスト
 - Kibana での中央集中管理
 - **elasticapm.properties** プロパティファイル
 - Java システムプロパティ
 - 環境変数
 - 実行時のアタッチパラメータ
 - デフォルト値

最小設定

- **service_name**
 - イベントのグループ化に利用
- **server_urls**
 - デフォルトは <http://localhost:8200>
- **application_packages**
 - スタックトレースの折りたたみに利用
- すべての設定オプションは [ドキュメント](#) を参照

プロパティファイル

- デフォルトではエージェントの jar が置かれている場所と同じディレクトリ内の **elasticapm.properties** という名前のファイルを探す
- config_file** オプションでパスを変更することができる
- プロパティファイルを作成する

```
service_name=my-cool-service
application_packages=org.example
server_urls=http://localhost:8200
```

Java システムプロパティ

- 全ての設定キーの頭には **elastic.apm.** がつく
- コマンドラインから設定可能

```
-Delastic.apm.service_name=my-application  
-Delastic.apm.application_packages=org.example  
-Delastic.apm.server_urls=http://localhost:820  
o
```

環境変数

- 全ての設定キーは大文字で先頭に **ELASTIC_APM_** がつく
- 以下の例のように OS の環境変数を設定する

```
export ELASTIC_APM_SERVICE_NAME=my-cool-service
export ELASTIC_APM_APPLICATION_PACKAGES=org.example
export ELASTIC_APM_SERVER_URLS=http://localhost:8200
```

カスタムイベント

- サポートされているフレームワークに加え、独自のコードパスも追跡できる
- **Java agent API** では以下が可能
 - 手動でカスタムの span や transaction を作成
 - 例外の捕獲
 - エラーの追跡
 - Transaction のアクティベート (分散トレーシング向け)
- Transaction は以下の方法で作成
 - Java アノテーションで暗黙的に
 - Static メソッド呼び出しで明示的に
- www.elastic.co/guide/en/apm/agent/java/current/public-api.html

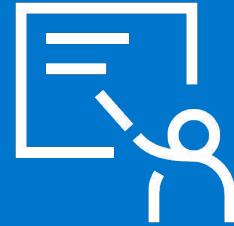
Log の関連付け

Trace に関するすべてのログへナビゲートできる

1. アプリケーションのログを Elasticsearch にインジェスト
 - Java ECS logging を使うか
 - Plain-text のログを出力している場合は独自のパーサーを定義
2. エージェントでログの関連付けを有効化
 - **enable_log_correlation** を **true** に
 - Java エージェントが Mapped Diagnostic Context に **transaction.id** と **trace.id** を注入
3. IDフィールドを 抽出
 - Java ECS logging は自動的にこれを実施
 - Plain-text ログの場合、パターンレイアウトを利用

Summary: **Java agent**

Module 3 Lesson 2



Summary

- Elastic APM Java Agent は自動的にアプリケーションの性能を計測しエラーを記録する
- ポピュラーなフレームワークやテクノロジをビルトインサポート
- 必要ならシンプルな API で手動でアプリケーションの計測も可能
- **service_name** はアプリケーションのイベントのグループ化に利用する重要な設定、指定しない場合 Java agent が推測する
- アプリケーションからエージェントへの依存関係を定義する必要はない

Quiz

1. **True or False:** Java アプリケーションのモニタリングを開始する際、まず行うべきはエージェントへの依存関係を宣言することだ
2. **True or False:** サポート対象外のテクノロジーを利用している場合でも Public API を使えばコードを計測できる
3. **True or False:** `service_name` 設定オプションは同一アプリケーション内で発生したイベントのグループ化に利用される

Java agent

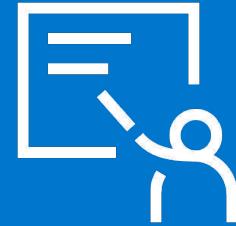
Lab 3.2

Java サーバーから **trace** をインジェストしましょう



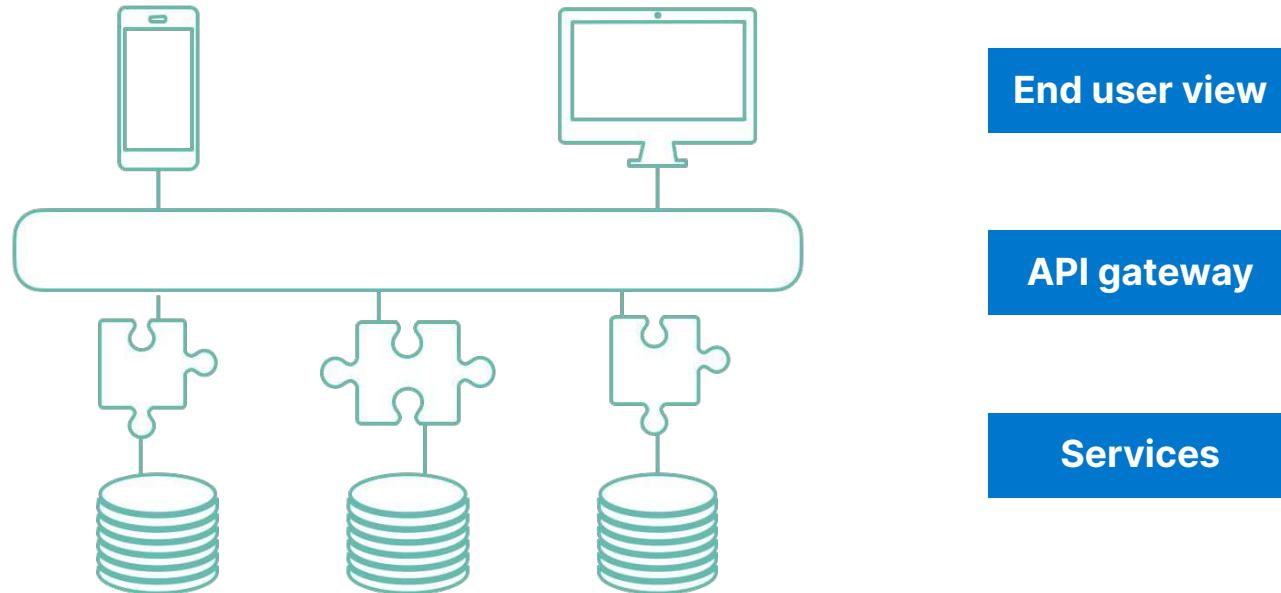
Node.js agent

Module 3 Lesson 3



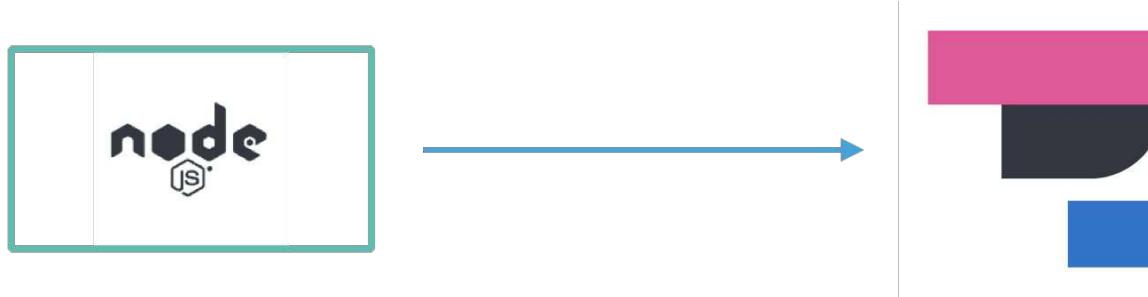
API gateway

- The API gateway はクライアント、バックエンドサーバー間の API 呼び出しに対するリバースプロキシとして振る舞う



Node.js agent

- 自動的に性能メトリックとエラーを収集
- ポピュラーなフレームワークやルーターをビルトインサポート
- あらゆるアプリケーションを計測可能なシンプルな API も提供

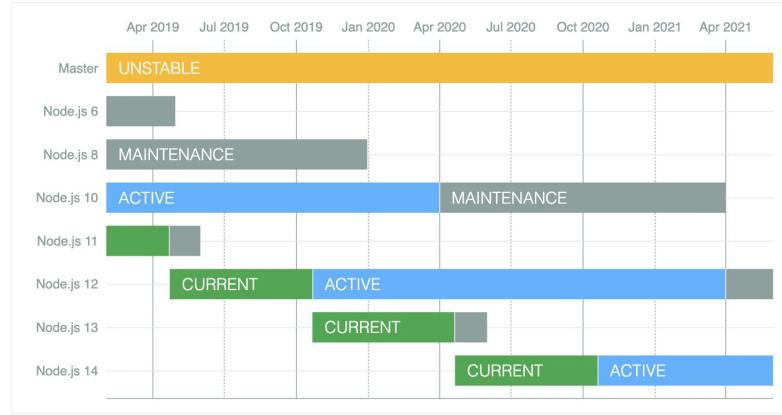


エージェントが動作する仕組み

- サポート対象のフレームワークを自動計測しイベントを記録
 - 例: データベースクエリや HTTP リクエスト
- ロードする際にモジュールにパッチを適用
- トレースのコンテキストを非同期の呼び出し間で伝搬
- 自動的にモジュールの関数呼び出しとコールバック呼び出しを関連付け
 - 実行時間の計測、メタデータと HTTP 情報
- Elastic APM に **transactions** や **spans** 形式でイベントを送信

サポート対象のテクノロジ

- Agent は Node.js 自体のサポートスケジュールに追随



- Frameworks
 - Express, hapi, Koa, Restify, Fastify, and AWS Lambda
- その他のモジュールも順次拡充中
 - www.elastic.co/guide/en/apm/agent/nodejs/current/supported-technologies.html

エージェントの設定

- まず **elastic-apm-node** モジュールをアプリケーションの依存関係に加える

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

Node.js agent を開始

- つづいて、Node.js agent をアプリケーションからその他のモジュールを require する前に開始する
 - すなわち express, http, などの前に
- メイン のファイルから agent を require し start する
 - index.js, server.js, app.js**

```
const apm = require('elastic-apm-node').start({  
    // add configuration options here  
})
```

エージェントの設定

- Node.js agent の設定方法はいくつがある
- 以下は優先度の高い順の設定方法
 - Kibana での中央集中管理
 - 環境変数
 - 設定オブジェクト
 - Agent 設定ファイル

必須の設定

- 必須の設定は **serviceName** だけ
- デフォルトでは、利用可能な場合 **package.json** の **name** フィールド
- 全ての設定オプションは [ドキュメント](#) 参照

環境変数

- エージェントの開始前に環境変数を export

```
ELASTIC_APM_SERVICE_NAME=my-application  
ELASTIC_APM_SERVER_URL=http://localhost:8200  
ELASTIC_APM_ENVIRONMENT=production
```

設定オブジェクト

- Agent 開始時に設定オブジェクトを渡すことができる

```
const apm = require('elastic-apm-node').start({  
  serviceName: 'my-application',  
  serverUrl: 'http://localhost:8200',  
  environment: 'production'  
})
```

Agent 設定ファイル

- デフォルトでは現在の作業ディレクトリ内にある **elastic-apm-node.js** という名前の設定ファイルを探す
- configFile** 設定オプションでカスタムのパスを設定可能
- 以下のように設定オブジェクトを export する設定ファイルを作成する

```
module.exports = {
  serviceName: 'my-application',
  serverUrl: 'http://localhost:8200',
  environment: 'production'
}
```

ES モジュールのサポート

- ES モジュールでは全ての import 文はいかなる関数呼び出しの前に評価される
 - 例: Babel, TypeScript, **--experimental-modules** フラグ
- このため、設定ファイルを start 関数に渡すことができない
- **elastic-apm-node/start** モジュールのインポートが必要

```
import apm from 'elastic-apm-node/start'
```

- そして環境変数が agent 設定ファイルで agent を設定する

TypeScript

- 次の方法でエージェントを import する

```
import * as apm from 'elastic-apm-node/start'
```

- default import を使いたい場合
 - **esModuleInterop** コンパイラーオプションを利用する

カスタムイベント

- Node.js agent でサポートされていない技術を使っている場合は?
 - 例: カスタムコードやバックグラウンドジョブ
- **Node.js agent APIs** を使うと以下が可能
 - カスタマイズして手動で span や transaction を作成できる
 - エラーを記録
- Agent をインストールし、アプリケーションで start する必要がある
- 詳細は [API reference](#) を参照

ログの関連付け

- アプリケーションのログと Elastic APM Node.js agent が捕獲した transaction を関連づけることができる
- 実現するにはログで以下の識別子を含む必要がある
 - **transaction.id, trace.id, span.id**
- Node.js agent の **apm.currentTraceIds** メソッドでこれらの情報を取得できる

ログ関連付けの例

- **apm.currentTraceIds** はオブジェクトを生成
 - **trace.id** と **transaction.id** か **span.id** のいずれかが設定されている
 - 利用できない場合空のオブジェクトが返る

```
{
  "trace.id": "abc123",
  "transaction.id": "abc123"
}
// or ...
{
  "trace.id": "abc123",
  "span.id": "abc123"
}
```

- これを利用し構造化 logger で APM trace とログを関連付けられる
- テキストのみの logger もサポートされている

Summary: **Node.js agent**

Module 3 Lesson 3



Summary

- Elastic APM Node.js Agent は自動的にアプリケーションの性能を計測しエラーを記録する
- ポピュラーなフレームワークやテクノロジーをビルトインサポート
- 必要なら、シンプルな API で手動でアプリケーション計測も可能
- Node.js APM agent をアプリケーションからその他のモジュールを require する前に開始する必要がある
- Node.js agent の開始時に必須の設定は **serviceName** だけ、設定されていない場合、デフォルトで **package.json** の **name** フィールドを利用する

Quiz

1. **True or False:** Node.js agent を利用するにはアプリケーションにインストールする必要がある
2. **True or False:** その他のモジュールを require する前に Node.js agent を開始する必要がある
3. **True or False:** start 関数に設定オブジェクトを渡すのが常に最適な設定方法である

Node.js agent

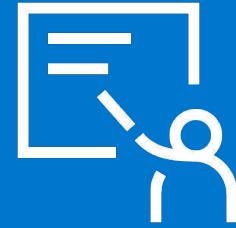
Lab 3.3

Node.js サービスから `trace` をインジェストしましょう



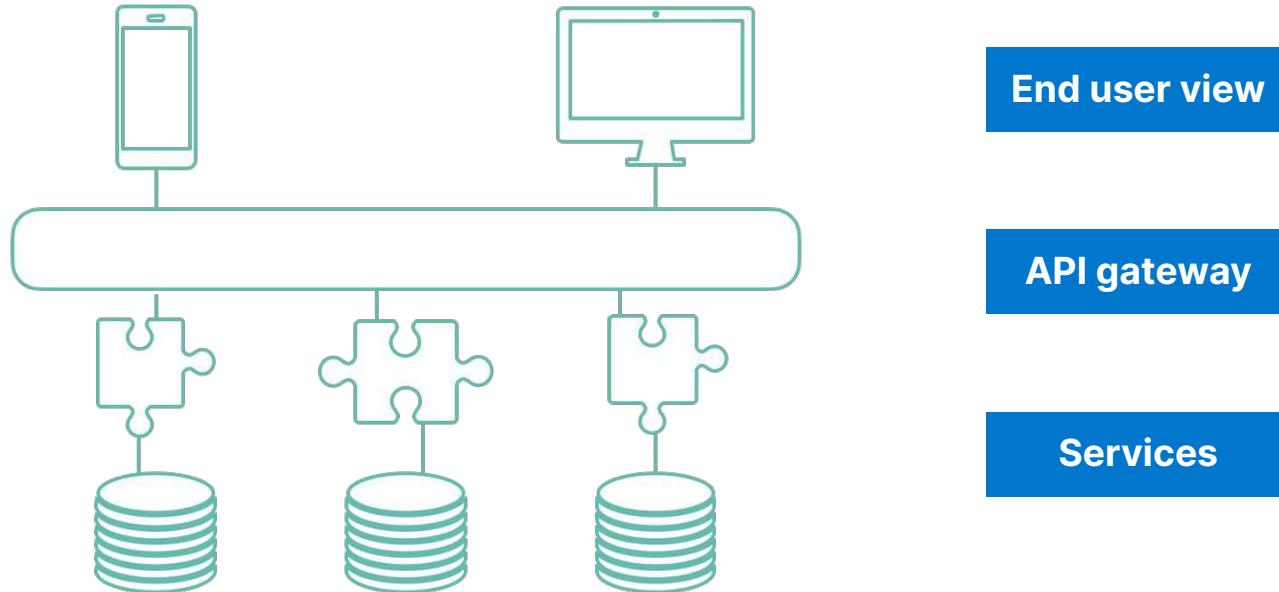
RUM agent

Module 3 Lesson 4



Frontend

- フロントエンドはユーザーがアプリケーションとやりとりする画面
- ユーザーはアプリケーションにアクセスしてどんなことをしている?



RUM JavaScript agent

- ユーザーのブラウザ上で動いている Web アプリの詳細な性能メトリックを取得し、エラーを記録
- ポピュラーなプラットフォームとフレームワークをビルトインサポート
- カスタムの計測向けの API も提供
- すべての送信リクエストで分散トレーシングをサポート



エージェントが動作する仕組み

- 自動的に以下を計測
 - ページロードメトリック
 - 静的アセットのロード時間
 - API リクエスト (XMLHttpRequest と Fetch)
- Navigation Timing と Resource Timing API を使い、ページロード性能と静的アセットのロード時間を取得
- 自動的に全ての HTTP 送信リクエストをキャプチャ
 - XHR と Fetch API リクエストを計測
- 処理されなかった JavaScript エラーを収集
- すべての transaction でエージェントは自動的に breakdown metrics をキャプチャ

既存クライアントへの影響

- 小さなフットプリント
- ユーザーのブラウザへのオーバーヘッドは最小
- ガイドライン
 - 大量のトラフィックの発生は避ける
 - ノイズの発生を避ける
 - 多すぎるデータは使えないデータになる

サポート対象のテクノロジ

- Platforms
 - Android, Chrome, Edge, Firefox, Internet Explorer, Safari, and iPhone
- Frameworks
 - integrations for React, Angular, and Vue
- Single page applications
- アプリケーションから登録したクリックイベントリスナー経由のユーザー操作
- その他多くの機能のサポートを順次拡充
 - www.elastic.co/guide/en/apm/agent/rum-js/current/supported-technologies.html

エージェントのセットアップ

- Elastic APM integration で RUM エンドポイントを有効に
- RUM agent をクライアントアプリにインストール
- RUM agent を設定

RUM endpoint を有効化



Integrations

Elastic APM

Elastic APM

[View deployment details](#)

Real User Monitoring

Manage the configuration of the RUM JS agent.

Enable RUM

Enable Real User Monitoring (RUM)

Enabled

Origin Headers

Optional

 X

Allowed Origin headers to be sent by User Agents.

Custom headers

Configure authentication for the agent

Access-Control-Allow-Headers

Optional

Supported Access-Control-Allow-Headers in addition to
"Content-Type", "Content-Encoding" and "Accept".

RUM agent をインストール

- RUN agent の利用方法はいくつかある
 - script タグの利用
 - bundlers の利用

Script タグの利用

- <script> タグを HTML ページに追加
- そして elasticApm グローバルオブジェクトを利用し agent を初期化

```
<script
  src="https://mydomain.com/path/to/elastic-apm-rum.umd.min.js"
  crossorigin
</script>
<script>
  elasticApm.init({
    serviceName: 'my-application',
    serverUrl: 'http://172.31.38.42:8200',
    serviceVersion: '1.0'
  })
</script>
```

- [GitHub](#) か [UNPKG](#) から最新バージョンのエージェントをダウンロード
 - お使いのサーバー/CDN に事前にファイルをデプロイ (~16KB)

bundler の利用

- アプリケーションに RUN agent を依存関係としてインストール
 - npm で agent バージョンを管理

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

- Agent を設定
 - RUN agent ライブラリをロードしサービスを初期化

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

Production build

- RUM agent は全てのデバッグメッセージをブラウザのコンソールに出力
 - 開発時にはとても便利なログだが
 - プロダクション向けには最適化されていない
- 最適化されたプロダクションバージョンを利用するよう注意
 - RUM agent バンドルのサイズを削減
 - 全てのデバッグメッセージを出力しない

Production build の例

- Webpack

```
const { EnvironmentPlugin } = require('webpack')

plugins: [
  new EnvironmentPlugin({
    NODE_ENV: 'production'
  })
]
```

Webpack 設定から
Environment プラグ
インを読み込み

- Rollup

```
const replace = require('rollup-plugin-replace')

plugins: [
  replace({
    'process.env.NODE_ENV': 'production'
  })
]
```

Replace プラグインを
読み込み正しい build
環境を使う

Agent の設定

- **serviceName**
 - 設定必須 (例 my-application-rum)
- **serverUrl**
 - RUM endpoint
- **serviceVersion**
 - アプリのバージョン (例: package.json, git commit ref)
 - Elastic APM integration で正しい sourcemap を探すために利用
- **active**
 - Agent を有効にするか無効にするかの boolean 設定
- 全ての設定オプションは [ドキュメント](#) 参照

分散トレーシングの設定

- RUN agent ではデフォルトで有効になっている
- 同一オリジンからのリクエストのみ対象
- クロスオリジンのリクエストは含まれない
- 利用するには **distributedTracingOrigins** 設定オプションを有効化
- 例を見てみましょう

分散トレーシングの設定例

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

https://my-application.com
への全てのリクエストは trace
の対象となる

クロスオリジンのリクエストも含
むようにする

カスタムイベント

- RUM agent でサポートされていないものを使っていた場合は?
 - 例: アプリケーションでは一切ページをロードしない
- RUM agent API を利用できる
 - カスタマイズして手動で span や transaction を作成
 - エラーを記録
- アプリケーションへ agent をインストールして開始する必要がある
- 詳細は [API reference](#) を参照

Summary: RUM agent

Module 3 Lesson 4



Summary

- Elastic APM integration で RUM endpoint を有効にする必要がある
- アプリケーションデプロイ時には production build を使う
- Agent は自動的にページロードメトリック、静的アセット、API リクエストのロード時間を計測
- Agent は必要な場合、クライアントコードを手動で計測するためのシンプルな API も提供
- 分散トレーシングで異なるオリジンからのリクエストも含めたい場合 cross-origin リクエストを有効化する必要がある

Quiz

1. **True or False:** Elastic APM integration で Real User Monitoring を有効化する設定が必要
2. **True or False:** RUM agent はユーザーのブラウザから **serviceName** で設定された Elastic APM integration にデータを送信する
3. **True or False:** RUM agent ではデフォルトで分散トレーシングが有効になっている

RUM agent

Lab 3.4

ユーザー体験のデータを JavaScript アプリケーションからインジェストしましょう



Agenda

- Module 1: Getting started
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- **Module 4: オブザーバビリティデータの活用**
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理

オブザーバビリティデータの活用

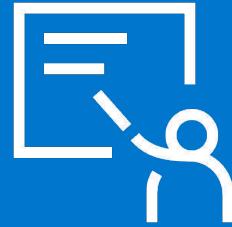
Module 4

Topics

- Logs アプリ
- Metrics アプリ
- APM アプリ
- User Experience アプリ

Logs アプリ

Module 4 Lesson 1



Logs アプリ

- サーバー、VM、コンテナなどから流れてきたすべてのログイベントを、ひとつの画面で集中管理することができる

Stream

Search for log entries... (e.g. host.name:host-1)

Customize Highlights Last 1 day Stream live

Apr 27, 2022 event.dataset Message

Showing entries from Apr 27, 11:23:56

Time	Source	Message
11:23:56.000	system.syslog	run-docker-runtime\x2drunc-moby-44c0660270116e4a714b5b165f41fad0e01c3558 fd55a2c4d8a56cd15cde3d94-runc.Vwza98.mount: Succeeded.
11:23:56.000	system.syslog	run-docker-runtime\x2drunc-moby-44c0660270116e4a714b5b165f41fad0e01c3558 fd55a2c4d8a56cd15cde3d94-runc.Vwza98.mount: Succeeded.
11:23:56.000	system.syslog	run-docker-runtime\x2drunc-moby-bb67ca8b4d00a9e8761661ed8e4fd08da71636d 766c1c3ba4fb125952156ee3-runc.y5dIzH.mount: Succeeded.
11:23:56.000	system.syslog	run-docker-runtime\x2drunc-moby-bb67ca8b4d00a9e8761661ed8e4fd08da71636d 766c1c3ba4fb125952156ee3-runc.y5dIzH.mount: Succeeded.
11:24:01.000	nginx.access	[nginx][access] 127.0.0.1 "GET /nginx_status? HTTP/1.1" 200 104
11:24:02.000	system.syslog	run-docker-runtime\x2drunc-moby-3af964e31261972a5331c00f84e60693cfea9559 495031524698a4a07ab5ae8c-runc.taXPJ4.mount: Succeeded.
11:24:02.000	system.syslog	run-docker-runtime\x2drunc-moby-3af964e31261972a5331c00f84e60693cfea9559 495031524698a4a07ab5ae8c-runc.taXPJ4.mount: Succeeded.
11:24:06.000	system.syslog	run-docker-runtime\x2drunc-moby-44c0660270116e4a714b5b165f41fad0e01c3558



ログイベントのストリーミング

- **ストリーミング**を使用してログイベントをリアルタイムに監視する
- またはストリーミングを**停止**して特定の時間帯の履歴を参照する

The screenshot shows a log stream interface. At the top, there are buttons for 'Customize' and 'Highlights', and a date range selector set to 'Last 1 day'. To the right of the date range is a button labeled 'Stop streaming' with a red oval around it. The main area displays log entries from April 27, 2022. Each entry includes a timestamp, a source dataset, and a message. The messages show various system logs like syslog and metricbeat logs. On the right side, a vertical timeline shows the time of each log entry, ranging from 11:29:55.000 to 11:30:05.000. A circular icon at the bottom indicates that new entries are being streamed.

Date	Source	Message
Apr 27, 2022	event.dataset	Message
11:29:55.000	system.syslog	495031524698a4a07ab5ae8c-runc.HjUVfk.mount: Succeeded. run-docker-runtime\x2drunc-moby-3af964e31261972a5331c00f84e60693cfea9559
11:29:57.000	nginx.access	495031524698a4a07ab5ae8c-runc.HjUVfk.mount: Succeeded. [nginx][access] 127.0.0.1 "GET /nginx_status? HTTP/1.1" 200 104
11:29:57.000	system.syslog	run-docker-runtime\x2drunc-moby-bb67ca8b4d00a9e08761661ed8e4fd08da71636d 766c1c3ba4fb125952156ee3-runc.LaCci0.mount: Succeeded.
11:29:57.000	system.syslog	run-docker-runtime\x2drunc-moby-bb67ca8b4d00a9e08761661ed8e4fd08da71636d 766c1c3ba4fb125952156ee3-runc.LaCci0.mount: Succeeded.
11:29:57.426	elastic_agent.metricbeat	[elastic_agent.metricbeat][info] Non-zero metrics in the last 30s
11:29:57.981	elastic_agent.metricbeat	[elastic_agent.metricbeat][error] Error fetching data for metricset mysql.status: Error 1045: Access denied for user 'root'@'172.18.0.1' (using password: YES)
11:30:05.000	system.syslog	run-docker-runtime\x2drunc-moby-3af964e31261972a5331c00f84e60693cfea9559 495031524698a4a07ab5ae8c-runc.3gb5yB.mount: Succeeded.

Last update 4 seconds ago

Streaming new entries

表示するフィールドのカスタマイズ

- 一覧にどのフィールドを表示するかを選択することができる

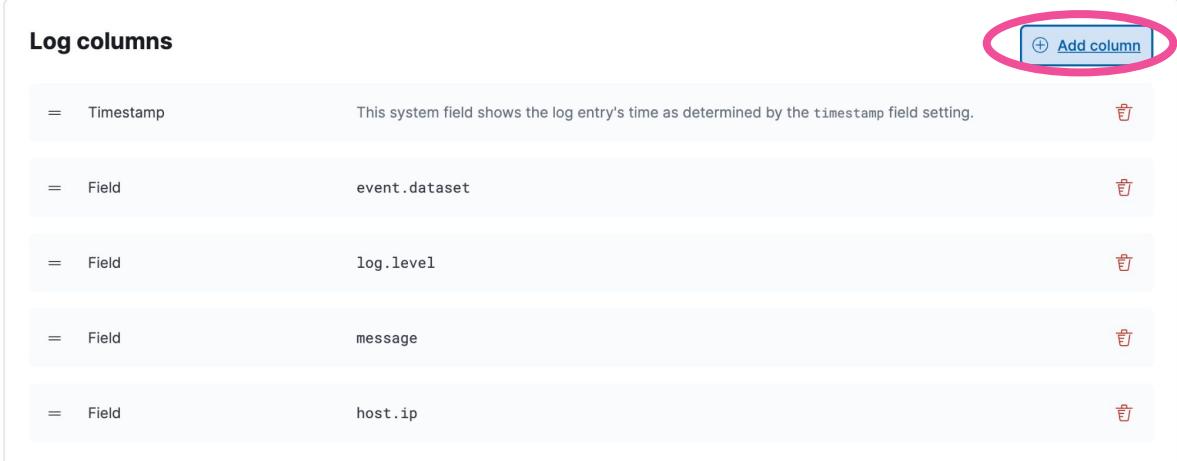
Settings Alerts and rules Add data

Log columns

=	Field	Description	Edit
=	Timestamp	This system field shows the log entry's time as determined by the timestamp field setting.	edit
=	Field	event.dataset	edit
=	Field	log.level	edit
=	Field	message	edit
=	Field	host.ip	edit

+ Add column

Discard Apply



ログのフィルタリング

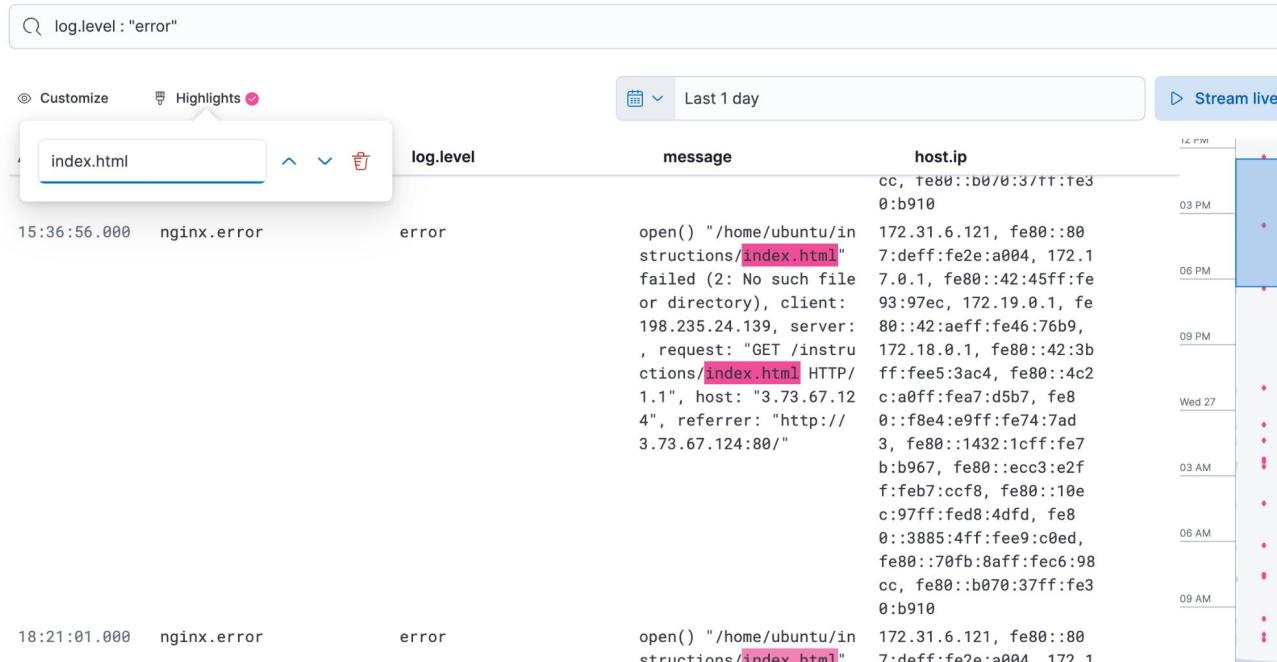
- Log App はブラウザ上で実行できる検索機能付きの `tail -f` のようなもの

The screenshot shows the Log App interface. At the top, there is a search bar with the query "log.level : \"error\"". Below the search bar are buttons for "Customize" and "Highlights", and a date range selector set to "Last 1 day". To the right of the date range is a button for "Stream live". The main area displays a table of log entries for April 26, 2022. The columns are "Apr 26, 2022", "event.dataset", "log.level", "message", and "host.ip". The table shows a single entry with the timestamp "12:30:30.000", dataset "nginx.error", log level "error", message detailing an error opening a file, and host IP "172.31.6.121". A vertical timeline on the right side shows the log entry's position over time, with markers for 03 PM, 06 PM, 09 PM, Wed 27, 03 AM, 06 AM, and 09 AM.

Apr 26, 2022	event.dataset	log.level	message	host.ip
Showing entries from Apr 26, 12:30:30	12:30:30.000	nginx.error	error open() "/home/ubuntu/in structures/index.html" 7:deff:fe2e:a004, 172.1 failed (2: No such file 7.0.1, fe80::42:45ff:fe or directory), client: 93:97ec, 172.19.0.1, fe 104.140.188.10, server: 80::42:aeff:fe46:76b9, , request: "GET /instru ctions/index.html HTTP/ 1.1", host: "3.73.67.12 1.1", host: "3.73.67.12 4", referrer: "http:// 3.73.67.124:80/"	172.31.6.121, fe80::80 7.0.1, fe80::42:45ff:fe 93:97ec, 172.19.0.1, fe 104.140.188.10, server: 80::42:aeff:fe46:76b9, 172.18.0.1, fe80::42:3b ff:fee5:3ac4, fe80::4c2 1.1", host: "3.73.67.12 c:a0ff:fea7:d5b7, fe8 0::f8e4:e9ff:fe74:7ad 3, fe80::1432:1cff:fe7 b:b967, fe80::ecc3:e2f f:feb7:ccf8, fe80::10e c:97ff:fed8:4dfd, fe8 0::3885:4ff:fee9:c0ed, fe80::70fb:8aff:fec6:98 cc, fe80::b070:37ff:fe3 0:b910

単語のハイライト

- 検索したログイベント内の単語を表内でハイライトすることができる



ログの詳細を精査

- 検索・filtratedしたイベントのメタデータや構造化されたフィールドの詳細を精査できる

```
open() "/home/ubuntu/in  
structions/index.html" 172.31.6.121, fe80::80  
7:deff:fe2e:a004, 172.1  
failed (2: No such file  
or directory), client:  
93:97ec, 172.19.0.1,  
198.235.24.139, server:  
80::42:aeff:fe46:76b! View in context  
, request: "GET /instru  
ctions/index.html HTTP/  
1.1", host: "3.73.67.12  
4", referrer: "http://  
3.73.67.124:80/"  
3, fe80::1432:1cff:fe7  
ELOG7 - 2022-04-26T18:36:56.000Z
```

Details for log entry ptMpZ4ABo04ycomZv3kM	
From index .ds-logs-nginx.error-default-2022.04.26-00001	
<input type="text"/> Search...	<button>Investigate ▾</button>
Field	Value
@timestamp	2022-04-26T18:36:56.000Z
agent.ephemeral_id	48b0217b-918b-42b3-a096-a2d9ca3335fe
agent.id	09bf446-2d1f-4dc3-9e48-3fa55750e23b
agent.name	ip-172-31-6-121
agent.type	filebeat
agent.version	8.2.0
cloud.account.id	836370109380
cloud.availability_zone	eu-central-1c
cloud.image.id	ami-01809964379e7a4
cloud.instance.id	i-0f427b389eb2e2c40

ログの文脈(Context) の理解

- 調査対象のイベントの詳細に加えて、そのイベントの前後でどのようなイベントが起こっていたかを同時にチェックできる

Displayed logs are from file /var/log/nginx/error.log and host ip-172-31-6-121

Apr 26, 2022	event.dataset	log.level	message	host.ip
15:36:56.000	nginx.error	error	open() "/home/ubuntu/instructions/index.html" failed (2: No such file or directory), client: 198.235.2.4.139, server: , request: "GET /instructions/index.html HTTP/1.1", host: "3.73.67.124", referrer: "http://3.73.67.124:80/"	172.31.6.121, fe80::807:deff:fe2:a004, 172.17.0.1, fe80::42:45ff:fe93:97ec, 172.19.0.1, fe80::42:aef4:fe46:76b9, 172.18.0.1, fe80::42:3bff:fee5:3ac4, fe80::4c2c:a0ff:fe
18:21:01.000	nginx.error	error	open() "/home/ubuntu/instructions/index.html" failed (2: No such file or directory), client: 128.14.134.170, server: , request: "GET /instructions/index.html HTTP/1.1", host: "3.73.67.124", referrer: "http://3.73.67.124:80/"	172.31.6.121, fe80::807:deff:fe2:a004, 172.17.0.1, fe80::42:45ff:fe93:97ec, 172.19.0.1, fe80::42:aef4:fe46:76b9, 172.18.0.1, fe80::42:3bff:fee5:3ac4, fe80::4c2c:a0ff:fe

Machine Learning とアラート

- Logs アプリ内でアラートを作成
- ログ内の異常の検出と調査
- ログイベントのカテゴリライズによるパターンの検出

Settings

Alerts and rules ▾

 Add data

Logs

Stream

Anomalies

Categories

Summary: Logs アプリ

Module 4 Lesson 1



Summary

- Logs アプリを利用してログイベントを監視できる
- ログのストリーミングにより、継続的に流入するログのリアルタイムな参照が可能
- Logs アプリはブラウザ上で実行できる検索機能付きの `tail -f` のようなもの

Quiz

1. **True or False:** Logs アプリではログを監視することができる
2. **True or False:** Logs アプリではログの履歴しか見ることができない
3. **True or False:** Logs アプリは `tail -f` の進化版と考えることができる

Logs アプリ

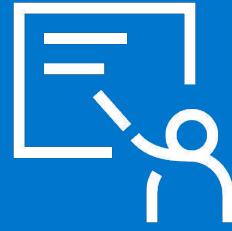
Lab 4.1

Logs アプリを探索してみましょう



Metrics アプリ

Module 4 Lesson 2



Metrics アプリ

- Elasticsearch 内のログや APM データを用いて、インフラのメトリックを可視化し、問題のあるスパイクの診断、高負荷状態のリソースの特定、Pod の自動的な検出と追跡、メトリックの統合を補助する

Observability

- Overview
- Alerts
- Cases
- Logs
- Metrics**
- APM
- Uptime
- User Experience

Inventory

Search for infrastructure data... (e.g. host.name:host-1)

04/27/2022 12:44:16 PM Auto-refresh

All 1

ip-172-31-6-121 4.6%

4.6%
0%

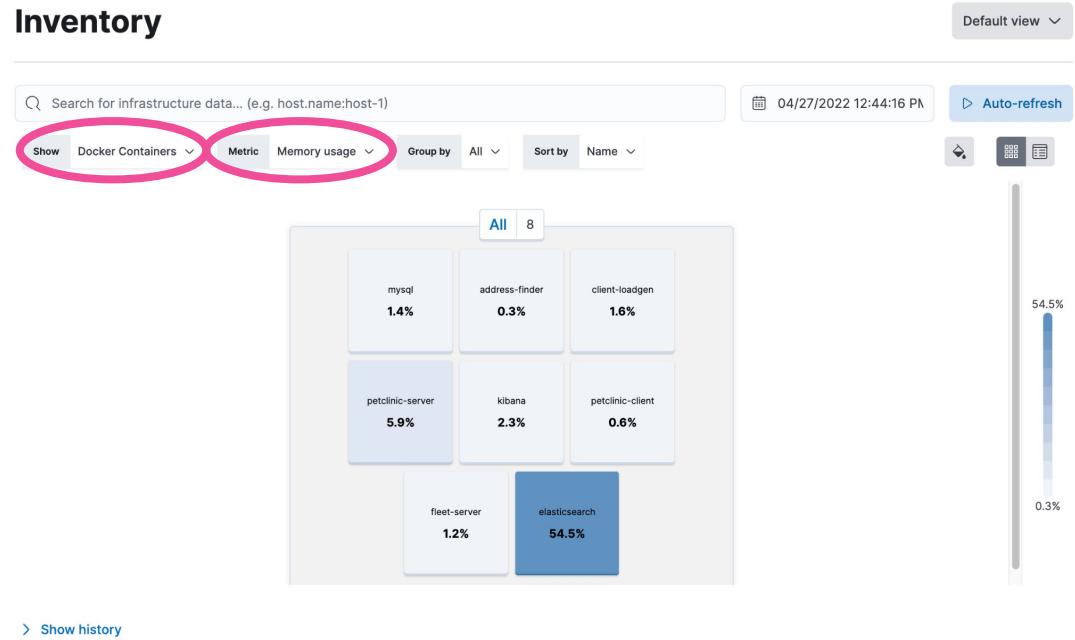


Copyright Elasticsearch BV 2015-2022 Copying, publishing and/or distributing without written permission is strictly prohibited

 elastic

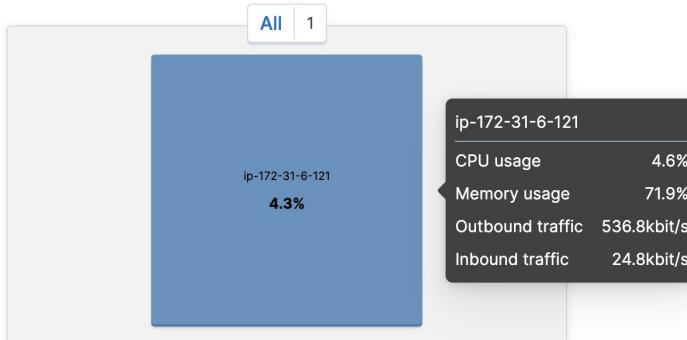
インフラストラクチャのメトリックの概観

- **Inventory** ページでは、監視対象のホストやコンテナ、Kubernetes Pod の概要を確認できる
- 表示可能なメトリックの例:
 - CPU
 - Memory
 - Log rate



ホストのメトリック

- System のインテグレーションを追加すると、Inventory からホストの各種メトリックを分析できる

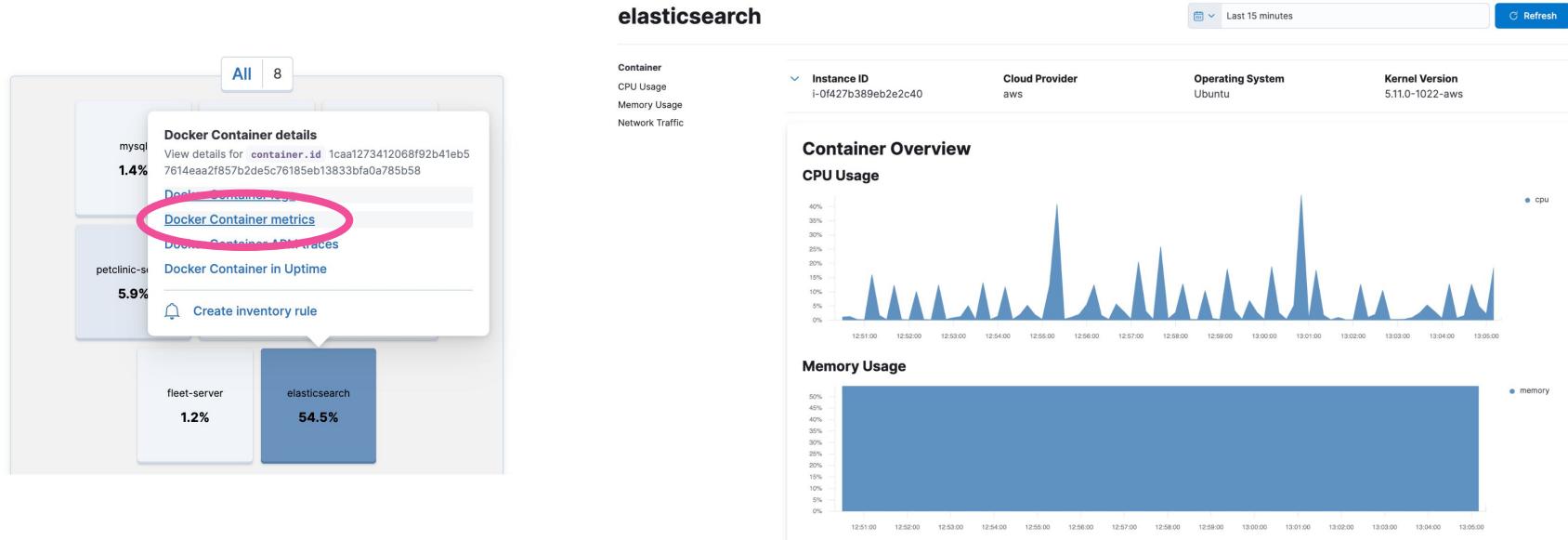


特定のリソースのログ、トレース、
Uptime 情報も参照可能



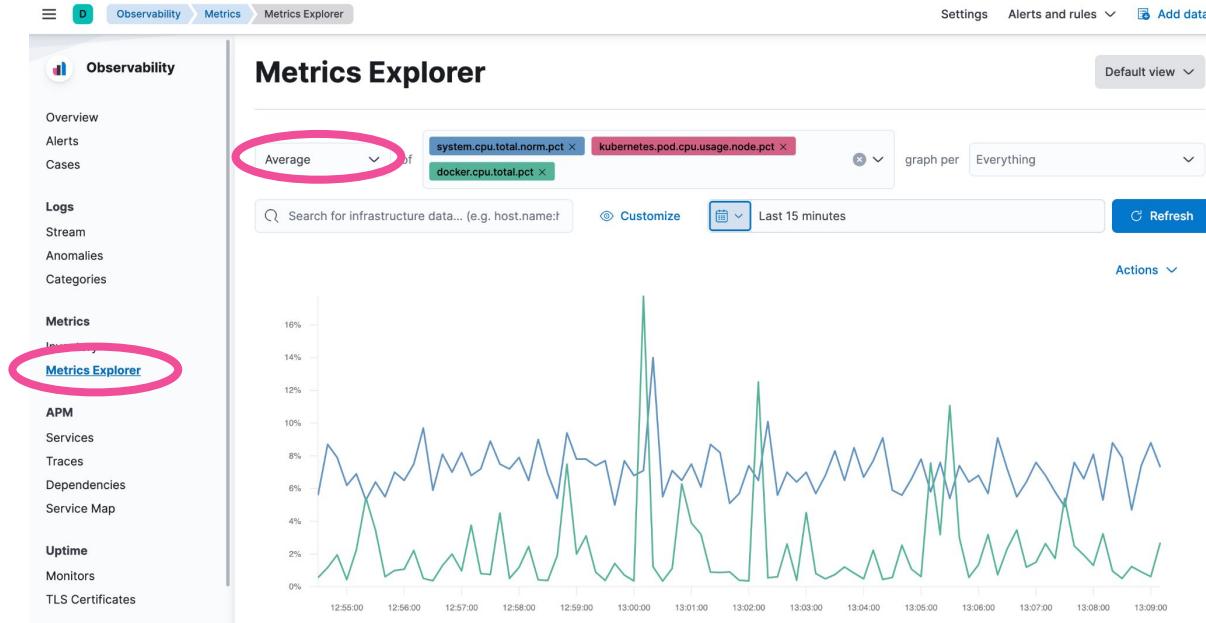
Docker コンテナのメトリック

- Docker のインテグレーションを追加すると、Inventory から Docker コンテナの各種メトリックを分析できる



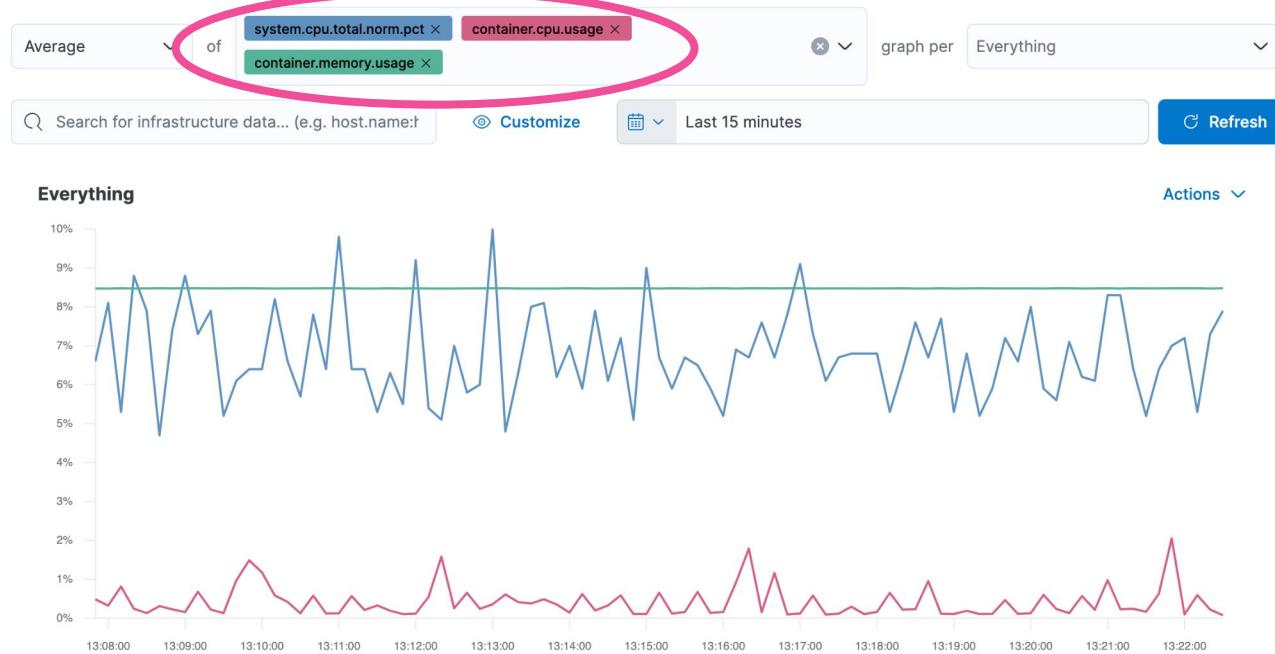
各種メトリックの集計と解析

- Metrics Explorer ページでは、収集した各種メトリックを集計して時系列のビジュアライゼーションを作成することができる



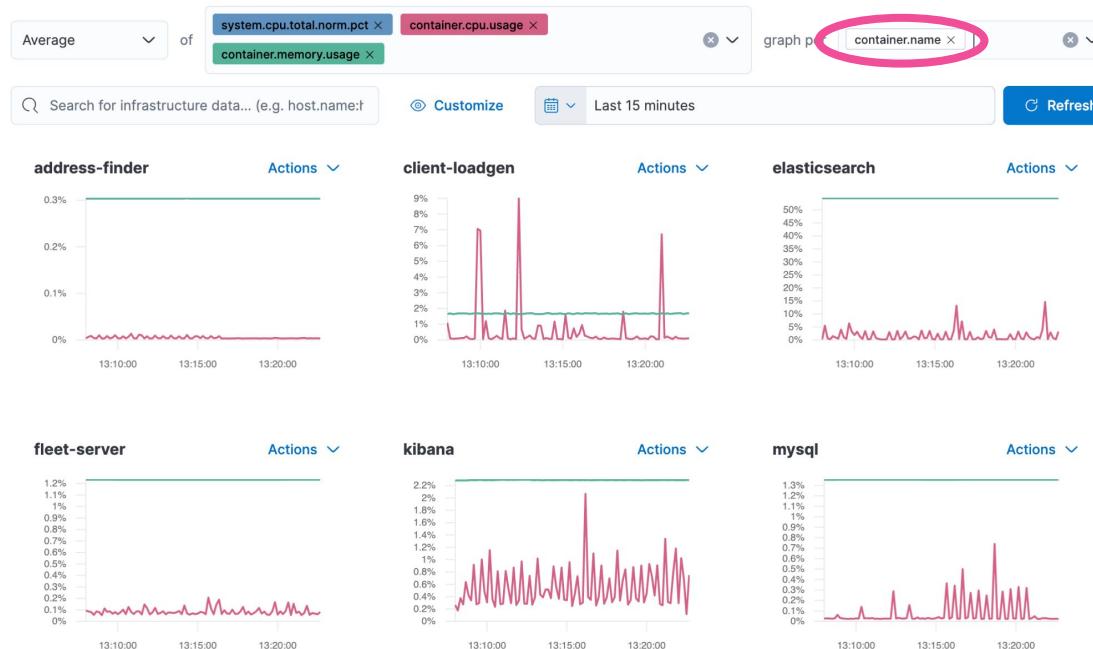
関連メトリックでチャート作成

- 関連メトリックを比較するチャートを作成



メトリックの分割

- 任意のフィールドでチャートを分割して表示することができる



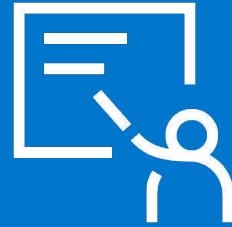
Machine Learning とアラート

- Machine learning ジョブを作成して、ホストや Kubernetes Pod のメモリの利用率やネットワークの異常状態を検知したり調査したりすることができる
- Metrics アプリから直接アラートを作成できる

Settings  Anomaly detection Alerts and rules ▾  Add data

Summary: Metrics アプリ

Module 4 Lesson 2



Summary

- **Metrics** アプリでは、インフラの問題を診断を補助してくれる各種メトリックを可視化することができる
- **Inventory** ページでは、監視対象のホスト、Docker コンテナ、Kubernetes Pod 等でインフラの各種メトリックをフィルタすることができる
- **Metrics Explorer** では、各種メトリックを集計して時系列のビジュアライゼーションを作成することができる

Quiz

1. **True or False:** Metrics アプリを利用して、インフラの各種の問題を診断することができる
2. **True or False:** Inventory ページではホストに関するメトリックだけしか参照できない
3. **True or False:** Metrics Explorer を利用してメトリックのビジュアライゼーションを作成することができる

Metrics アプリ

Lab 4.2

Metrics アプリを探索してみましょう



APM アプリ

Module 4 Lesson 3



APM アプリ

- 監視対象のアプリケーションの健康状態やパフォーマンスに関する高次元のオーバービューを提供する

The screenshot shows a navigation sidebar on the left with the following items:

- Observability
- Overview
- Alerts
- Cases
- Logs
- Metrics
- APM** (highlighted with a pink arrow pointing to the Services section)
- Uptime
- User Experience

The main area is titled "Services" and displays the following information:

Name	Environment	Latency (avg.)	Throughput	Failed transaction rate
petclinic-node	development	5.3 ms	42.5 tpm	0%
petclinic-react		156 ms	4.5 tpm	N/A
petclinic-java		9.8 ms	1.9 tpm	0%
petclinic-address-finder		37 ms	0.1 tpm	0%

Below the table, there is a search bar, a time range selector (Comparison, Day before, Last 15 minutes), and a refresh button.

Page footer:

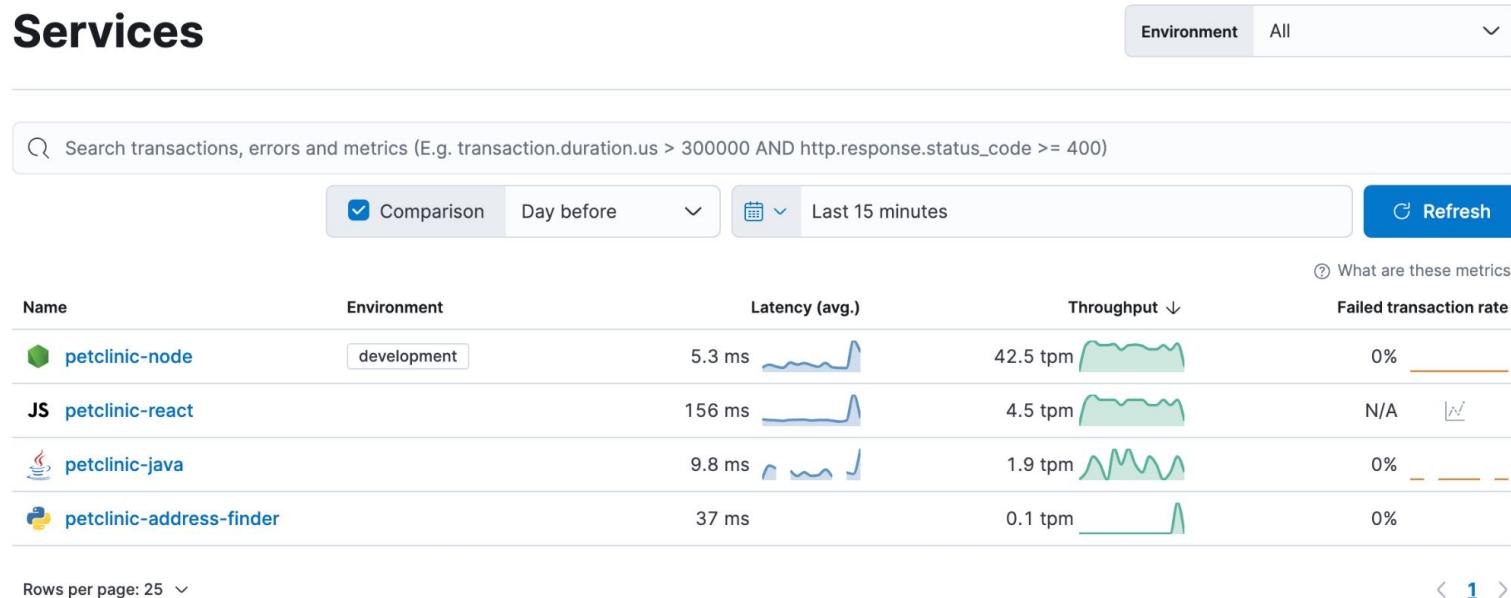
Copyright Elasticsearch BV 2015-2022 Copying, publishing and/or distributing without written permission is strictly prohibited

elastic

Services インベントリ

- 監視対象のサービスの健康状態や一般的なパフォーマンスについての簡潔なオーバービューを提供する

Services



Traces

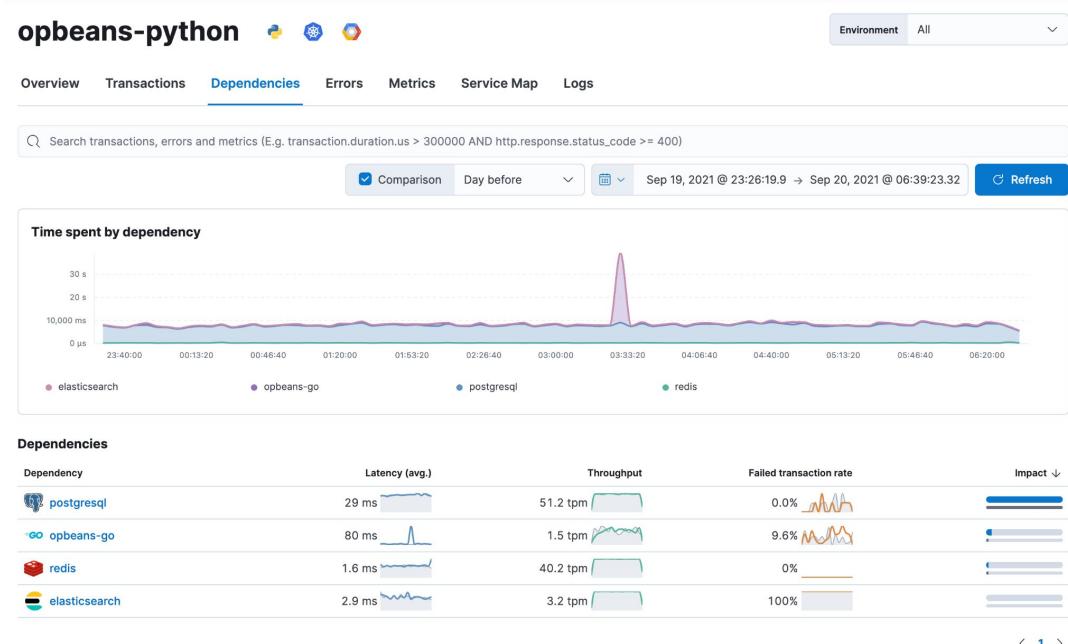
- Traces は関連するトランザクションを統合して、リクエストがどのサービスによってどのように処理されたかという、end-to-end のパフォーマンス情報を提供する
- Traces は各トランザクションの開始点を表示する

Traces

Name	Originating service	Latency (avg.)	Traces per minute	Impact
/	JS petclinic-react	259 ms	2.2 tpm	<div style="width: 100%;"></div>
GET static file	petclinic-node	3.7 ms	33.5 tpm	<div style="width: 100%;"></div>
/error	JS petclinic-react	109 ms	0.8 tpm	<div style="width: 100%;"></div>
/owners/list	JS petclinic-react	124 ms	0.7 tpm	<div style="width: 100%;"></div>
/vets	JS petclinic-react	143 ms	0.5 tpm	<div style="width: 100%;"></div>
OwnerEditor:StateChange	JS petclinic-react	441 ms	0.1 tpm	<div style="width: 100%;"></div>
FindOwnersPage	JS petclinic-react	294 ms	0.2 tpm	<div style="width: 100%;"></div>
/api/owners	petclinic-node	54 ms	0.9 tpm	<div style="width: 100%;"></div>
OwnerEditor:ZipChange	JS petclinic-react	265 ms	0.1 tpm	<div style="width: 100%;"></div>
/api/vets	petclinic-node	38 ms	0.5 tpm	<div style="width: 100%;"></div>
ErrorPage	JS petclinic-react	281 ms	< 0.1 tpm	<div style="width: 100%;"></div>
VetsPage	JS petclinic-react	274 ms	< 0.1 tpm	<div style="width: 100%;"></div>

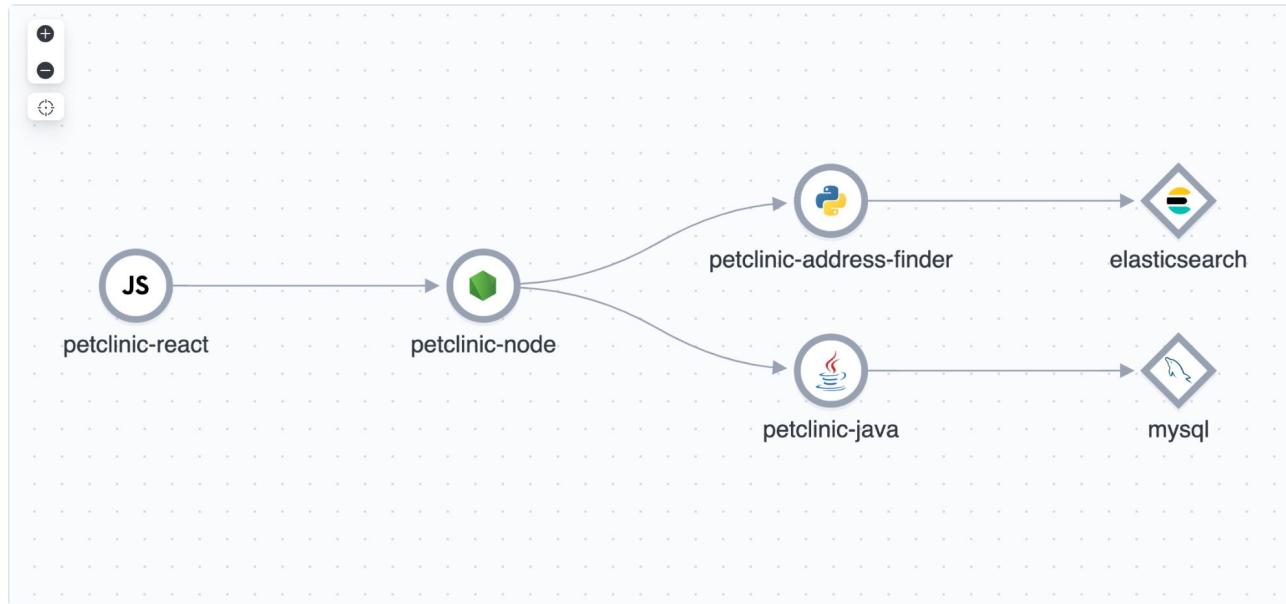
Dependencies

- データベースやサードパーティのサービスなど、APMに計装化されていないアプリケーションの依存関係に関するオーバービューを提供する



Service Map

- Service Map は、監視対象のアプリケーションのアーキテクチャに計装化されたサービスの関連性をグラフ形式でリアルタイムに表示する



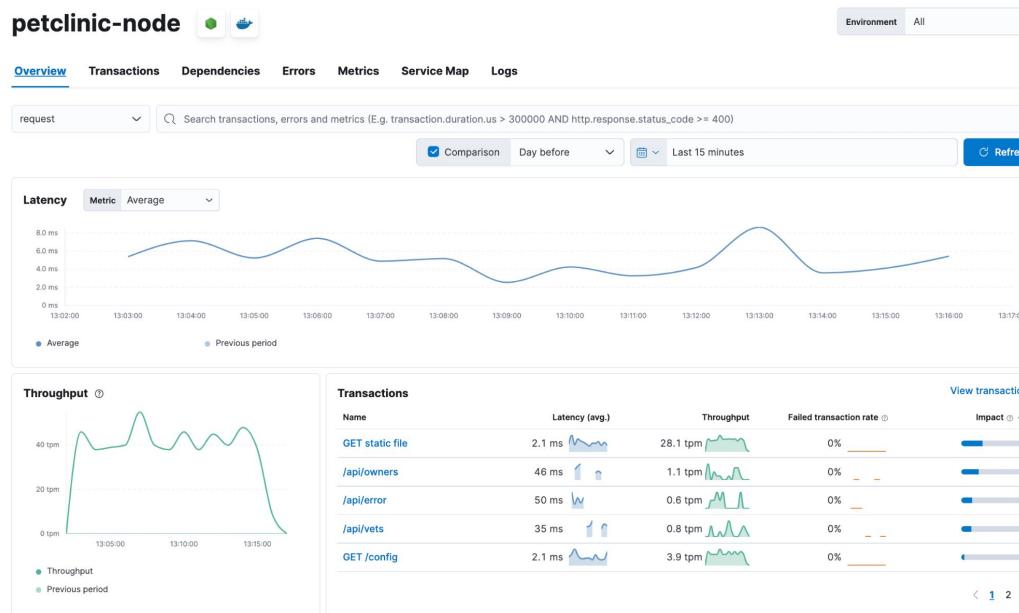
Service Map の種類

- 全体
 - すべての計測化されたサービスとそのコネクションを表示する
- サービス固有
 - 選択したサービスのコネクションをハイライトする



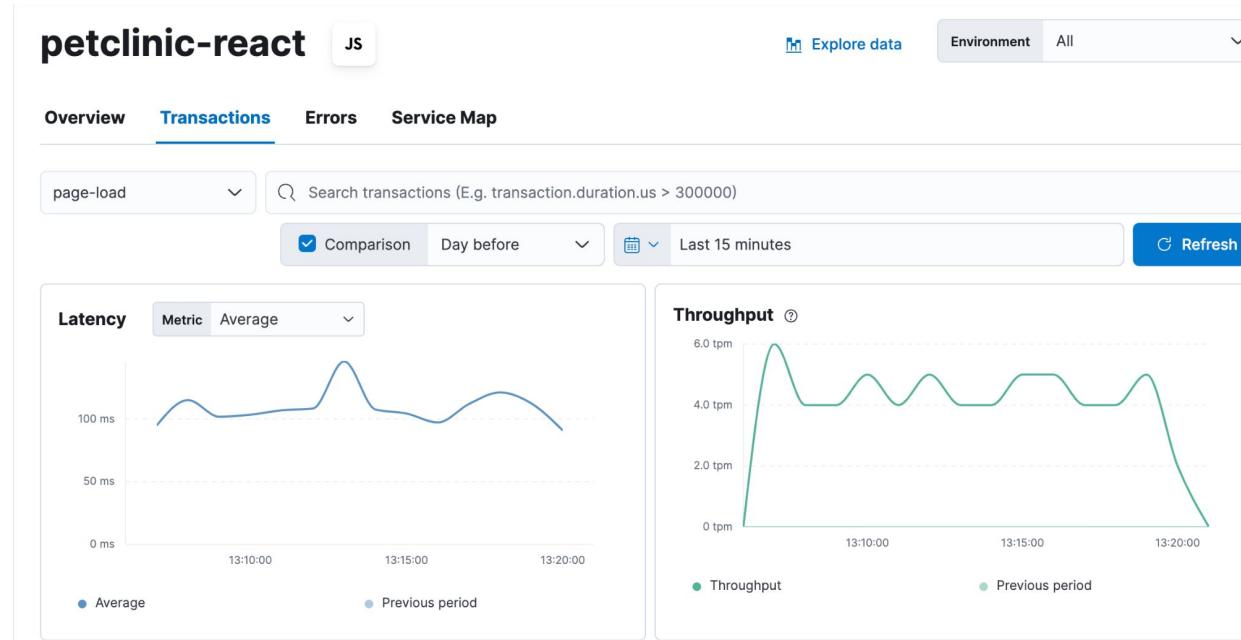
Overview

- サービスの稼働状況についての高次元な可視性を提供する様々なチャートやテーブルをチェックできる



Transactions 概要

- HTTP リクエストやデータベースクエリ等のパフォーマンスマトリックを可視化する



Transactions テーブル

- 選択したサービスに関するトランザクショングループのリストを表示

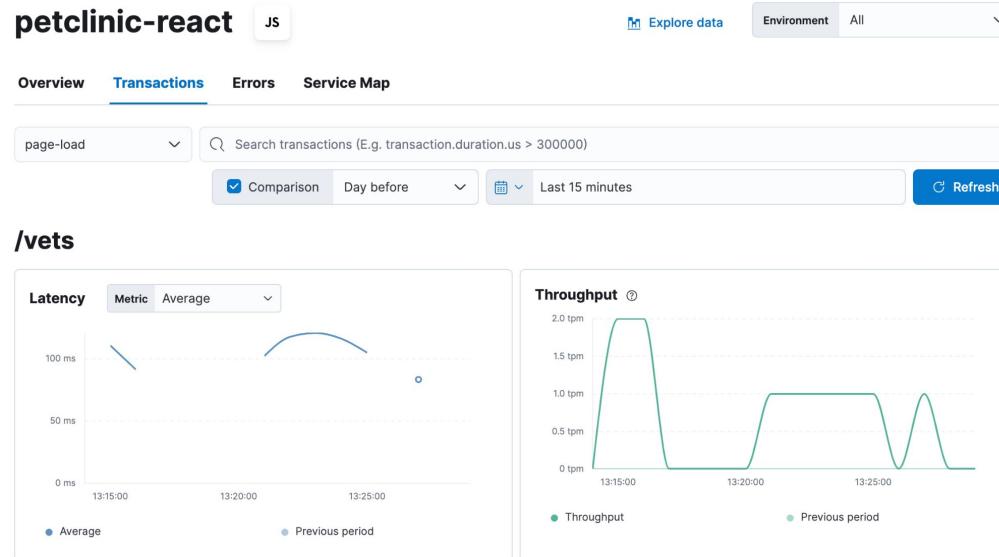
Name	Latency (avg.)	Throughput	Failed transaction rate	Impact
/	105 ms	1.5 tpm	N/A	Medium
/owners/list	112 ms	0.9 tpm	N/A	Medium
/vets	105 ms	0.7 tpm	N/A	Medium
/error	105 ms	0.6 tpm	N/A	Medium
/owners/new	118 ms	0.1 tpm	N/A	Low
/owners/1/pets/1/edit	171 ms	< 0.1 tpm	N/A	Low
/owners/9	154 ms	< 0.1 tpm	N/A	Low
/owners/1	92 ms	< 0.1 tpm	N/A	Low

Rows per page: 25 ▾

◀ 1 ▶

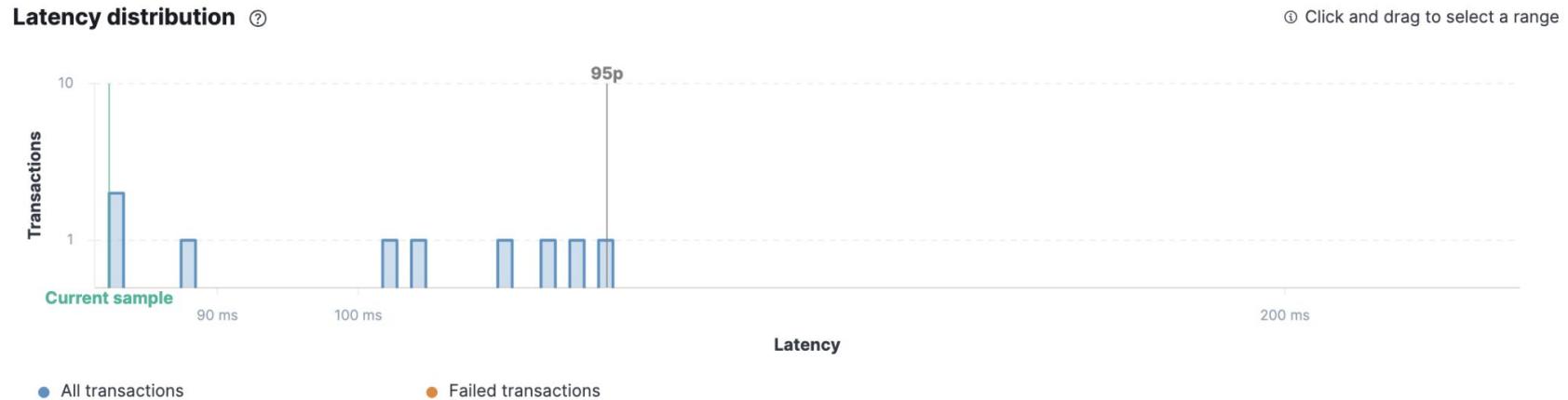
Transaction 詳細

- トランザクションの概要部と類似
- 選択されたトランザクション・グループに関わるすべてのデータを表示する



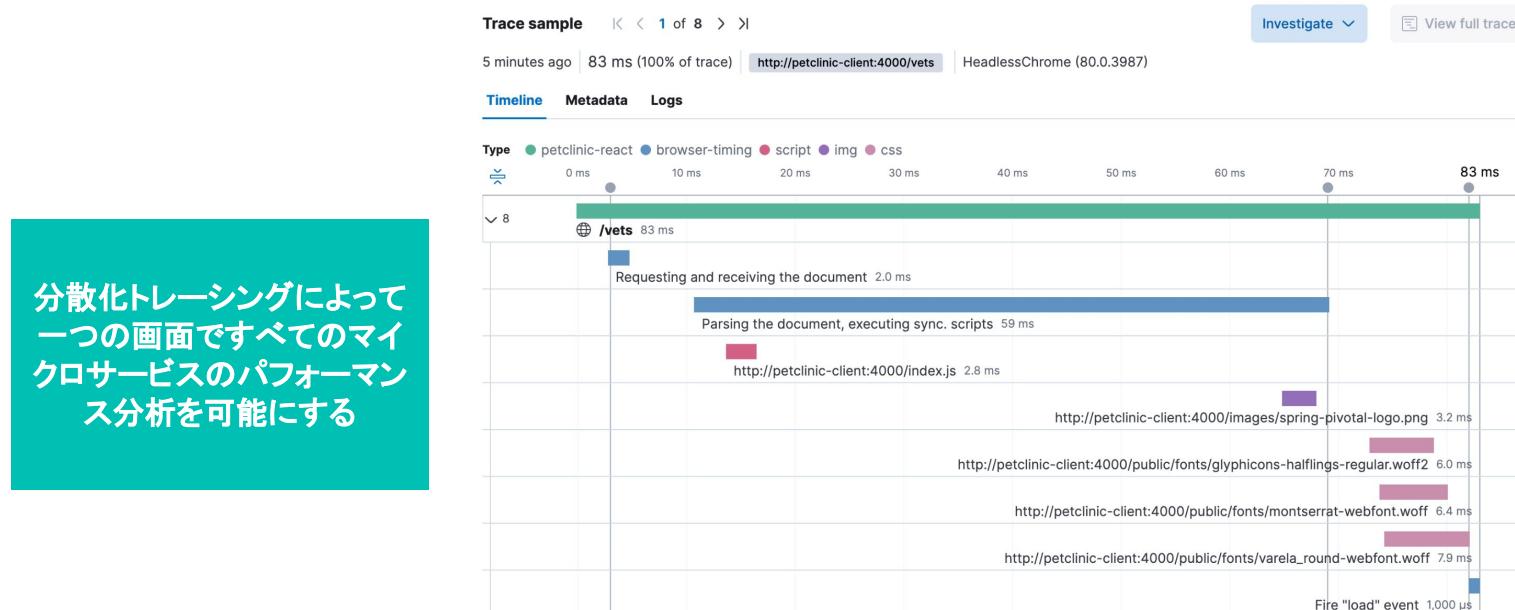
Latency distribution

- 特定の期間内のトランザクションの実行時間をチャートにプロット



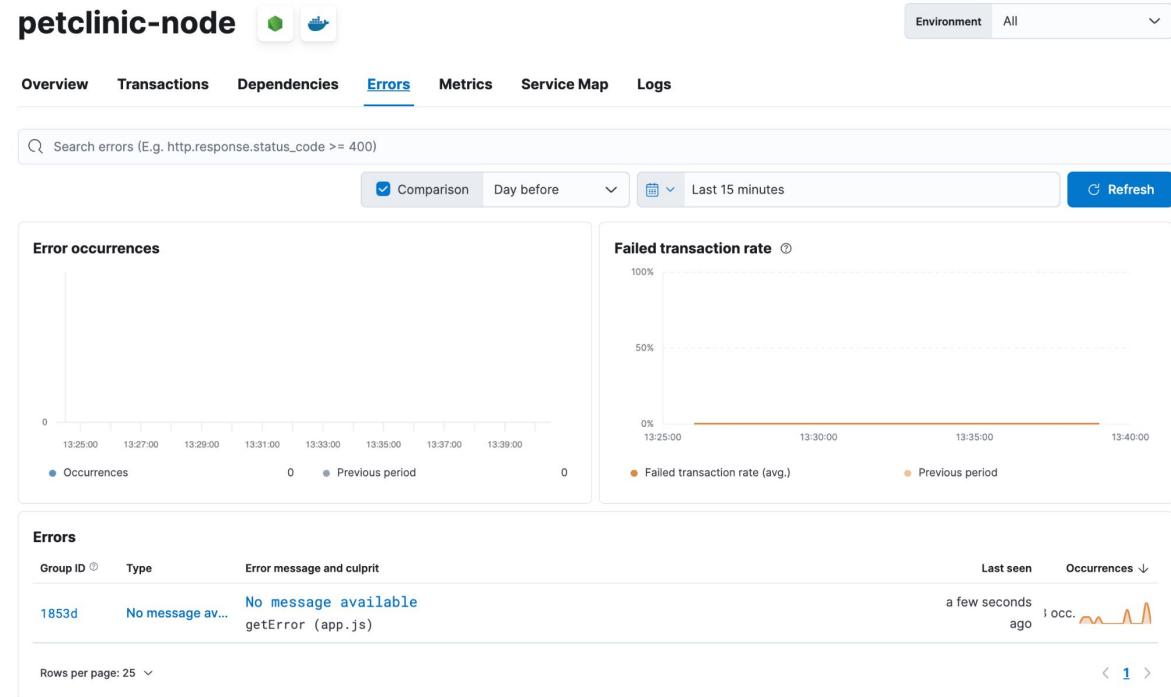
Trace sample timeline

- 実行時間のウォーターフォールチャート。トランザクションの親子関係を理解して、あるリクエストがなぜ遅いのかを最終的に特定するのに役立つ



Errors 概要

- APM のエージェント、または APM Agent API でユーザーがマニュアルでレポートした例外の高次元なビューを提供する



Error occurrence

- 厳密にいつどこでエラーが起きたかをそのスタックトレースとともにチェックできる

Error occurrence [View 9 occurrences in Discover.](#)

a few seconds ago | GET http://petclinic-client:4000/api/error | 404 Not Found | HeadlessChrome (80.0.3987) | [/api/error](#)

[Log stack trace](#) [Exception stack trace](#) [Metadata](#)

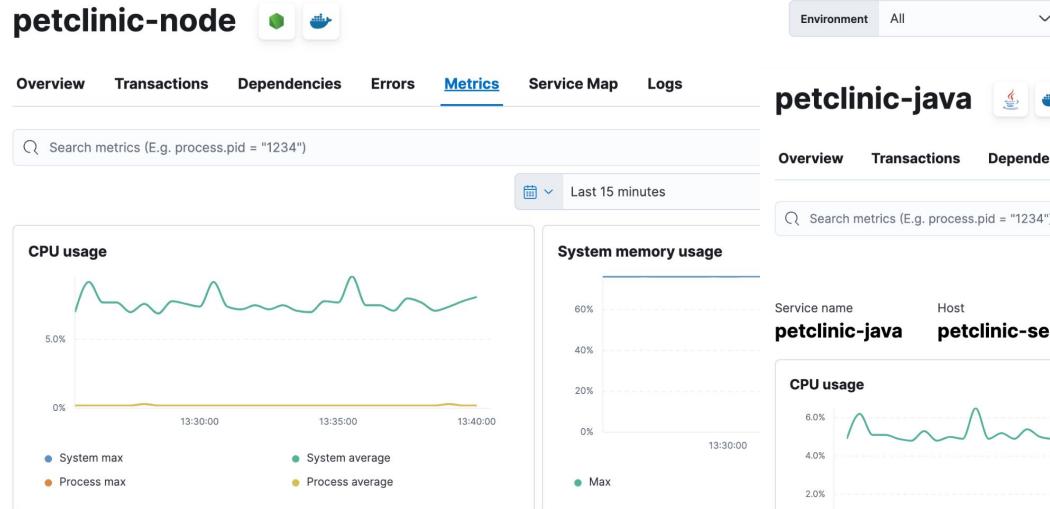
```
✓ at userResDecorator (app.js:192)
187.     return settings.api_prefix+req.url
188.   },
189.   userResDecorator: function(proxyRes, proxyResData, userReq, userRes) {
190.     if (proxyRes.statusCode >= 400) {
191.       let err = getError(proxyRes, proxyResData);
192.       apm.captureError(err, {
193.         request: userReq,
194.         response: proxyRes,
195.         custom: captureErrorResponse(proxyResData)
196.       });
    
```

› [3 library frames](#)

Metrics 概要

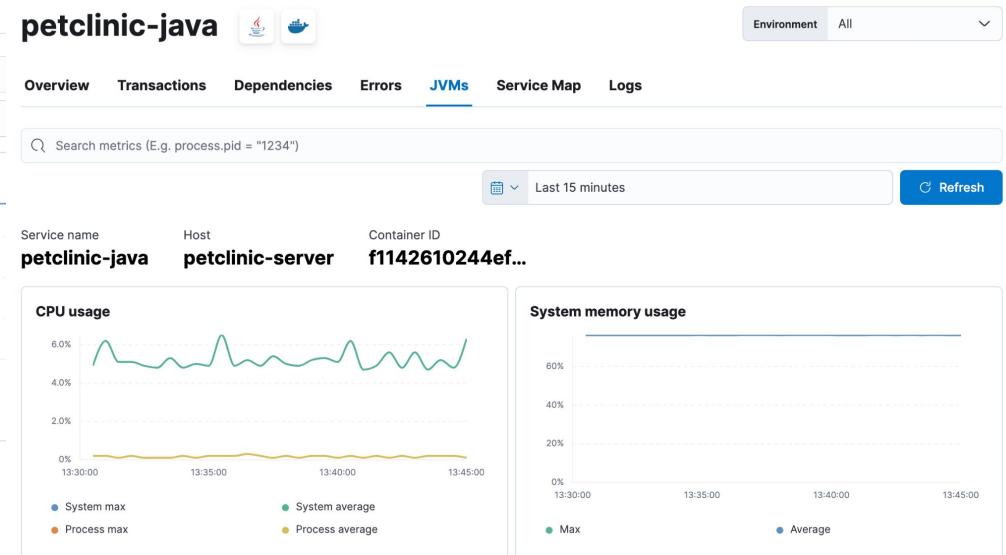
- 各エージェント固有のメトリックを表示する

petclinic-node



View metrics for each JVM when using the Java agent

petclinic-java



Logs 概要

- アプリケーションに関連するログの表示

petclinic-node

Environment All

Overview Transactions Dependencies Errors Metrics Service Map Logs

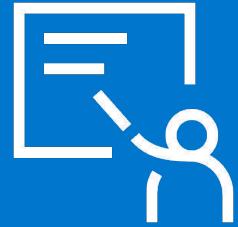
Last 15 minutes Refresh

Timestamp	Message
Showing entries from May 4, 13:26:48	
13:26:48.080	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:27:23.746	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:29:25.943	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:35:07.013	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:35:57.289	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:39:02.623	[[access] 138.204.24.129 N/A "GET /api/error? HTTP/1.1" 404
13:39:44.750	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:39:59.538	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
13:40:15.135	[[access] 172.18.0.3 N/A "GET /api/error? HTTP/1.1" 404
Showing entries until May 4, 13:40:15	

Summary:

APM アプリ

Module 4 Lesson 3



Summary

- APM アプリはアプリケーションの健康状態やパフォーマンスの高次元のオーバービューを提供する
- APM の分散化トレーシングは一つの画面ですべてのマイクロサービスのパフォーマンス分析を可能にする
- Service Map は、監視対象のアプリケーションのアーキテクチャに計装化されたサービスの関連性をグラフ形式でリアルタイムに表示する

Quiz

1. **True or False:** Service Map を使うとアプリケーションアーキテクチャを可視化することができる
2. **True or False:** APM アプリは system インテグレーションで供給されるメトリックスを表示する
3. **True or False:** APM アプリの Traces を分析することで応答の遅いリクエストを発見できる

APM app

Lab 4.3

APM アプリを探索してみましょう



User Experience アプリ

Module 4 Lesson 4



User Experience アプリ

- User Experience アプリはウェブアプリケーションの知覚的なパフォーマンスを数値化して分析する方法を提供する

The screenshot shows the Elasticsearch Observability interface. On the left, a sidebar lists navigation options: Overview, Alerts, Cases, Logs, Metrics, APM, Uptime, and User Experience. A large pink arrow points from the 'User Experience' link towards the main dashboard area. The main dashboard is titled 'Dashboard' and includes filters for 'Web application' (set to 'petclinic-react'), 'Percentile' (set to '50th (Median)'), and 'Environment' (set to 'All'). It also features a 'Last 15 minutes' time range and a 'Refresh' button. Below these filters is a search bar labeled 'Filter by URL' and dropdown menus for 'Location', 'Device', 'OS', and 'Browser'. The dashboard displays several performance metrics:

Page load (median)			
Total	92 ms	Backend	3 ms
Frontend	89 ms	Total page views: 70	

Metrics (median)			
First contentful paint	99 ms	Total blocking time	0 ms
No. of long tasks	0	Longest long task duration	0 ms
Total long tasks duration: 0 ms			

なぜ **User Experience** が重要なのか？

- サーチエンジンプロバイダーは、ウェブサイトのランキングを進めるに連れ、ユーザー体験に重点を置くようになってきた
- Elastic を使えば、Google Core Web Vitals に準拠したウェブサイトデータの評価を簡単に行うことができる
 - ページロードのスピード・視覚的な安定性・インタラクティビティ
- これらの Core Web Vitals は Google のランキングを決定する主要素になりつつある
- 自分たちのウェブサイトが Google の検索結果の上位に表示されるためには、適切な Core Web Vitals スコアを獲得する必要がある

User Experience はどのように動いている？

- RUM エージェントによって実現されている
- RUM エージェントはブラウザの timing API を利用する
 - 例: Navigation Timing, Resource Timing, Paint Timing, and User Timing
- User Experience メトリックは、ユーザーがページの一つを取得する度にデータを取得する
- メトリックデータは Elasticsearch に保存され、Kibana 上で参照できる

ページロードの所要時間

- 対象のウェブサイトの概要を提供する
 - サーバーはリクエストの応答にどの程度時間を要しているか？
 - フロントエンドでのパースと描画にどの程度時間を要しているか？
 - どのぐらいのページビューを達成しているか？

Page load (median)

Total ⓘ

92 ms

Backend ⓘ

3 ms

Frontend ⓘ

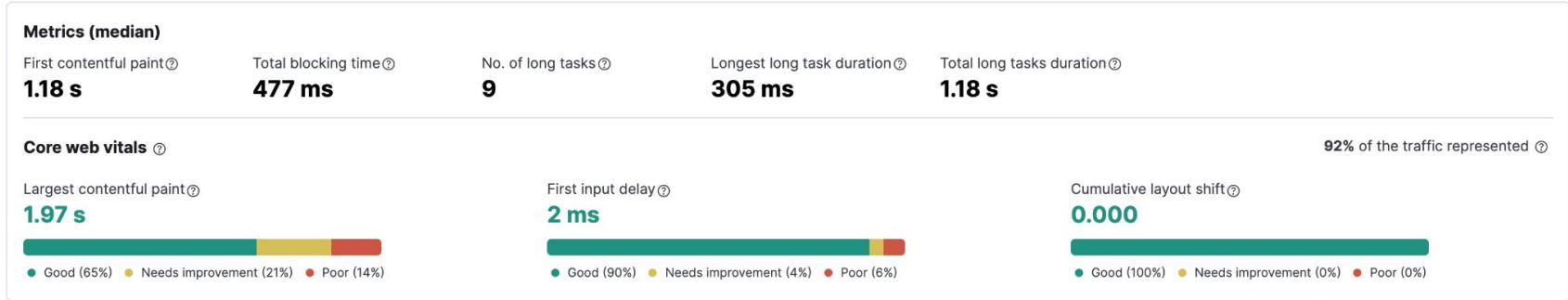
89 ms

Total page views

70

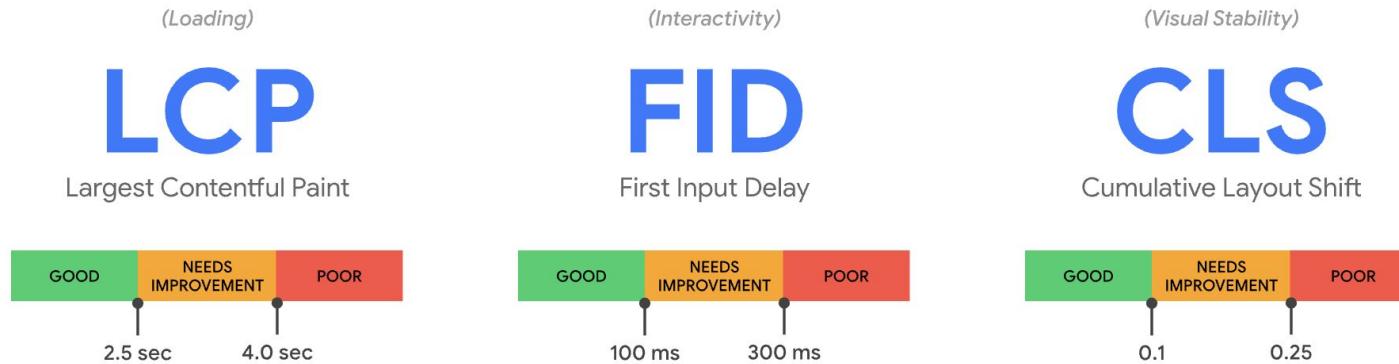
User Experience のメトリック

- 知覚的なパフォーマンスの理解を補助する



Core Web Vitals

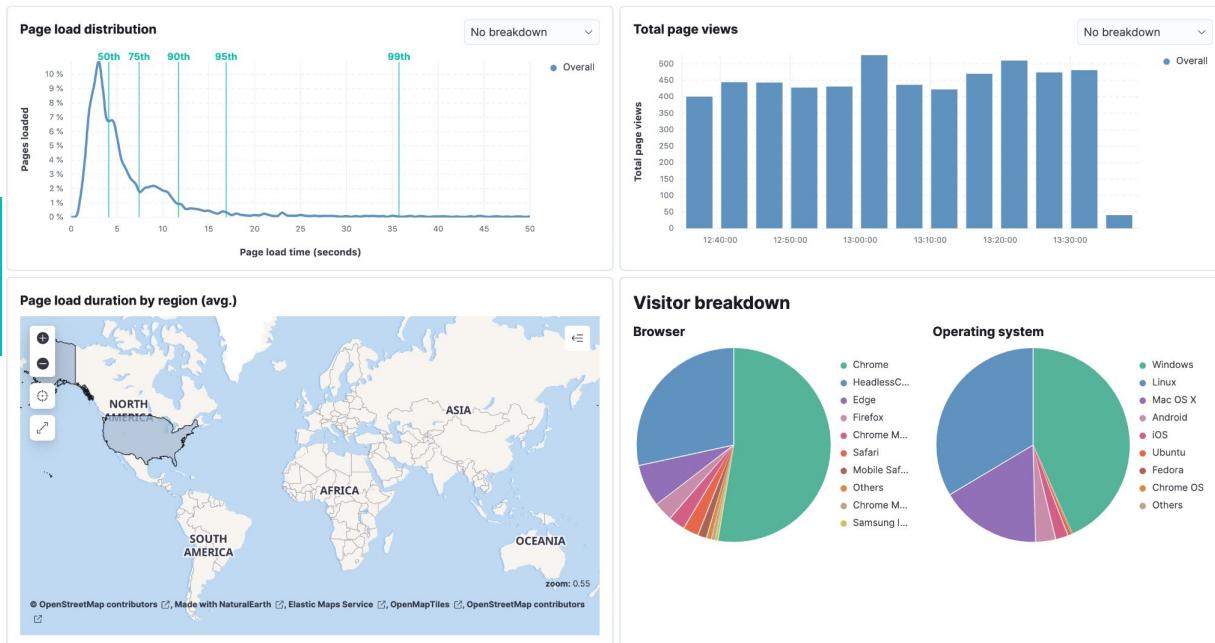
- Google は近年、現実世界でのユーザー体験を数値化し、サイトの優劣をカテゴライズする指標のセットを定義した



Load/View の分布

- オペレーティングシステム(OS)やブラウザのファミリー、地理的な場所はそれぞれウェブサイトのユーザー体験に多大な影響を与える

改善の優先度を検討するため、ユーザーがいつどこからアクセスしているかを理解する



エラーのブレークダウン

- JavaScript のエラーはウェブサイトの良質なユーザーエクスペリエンスの障害になり得る
- エラーモニタリング機能は本番環境のウェブサイトで発生した JavaScript のエラーを捕捉するのに役立つ

JavaScript errors	
Error message	Impacted page loads
Script error.	73.0 %
Uncaught NotFoundError: Failed to execute 'removeChild' on 'Node': The node to be removed is not a child of this node.	12.9 %
Uncaught TypeError: Cannot read property 'push' of undefined	0.2 %
ReferenceError: Can't find variable: _swipe	0.0 %
Unhandled promise rejection: TypeError: null is not an object (evaluating 'c.tagName')	0.7 %
Rows per page: 5	< 1 2 3 4 5 6 7 >

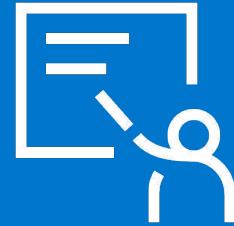
Explore data

- パフォーマンスの内訳、Key Performance Indicator(KPI)、Core Web Vitals 等の分析のための詳細なビジュアライゼーションを作成できる



Summary: User Experience app

Module 4 Lesson 4



Summary

- User Experience アプリは対象の知覚的に捉えられたウェブアプリケーションのパフォーマンスを実際に数値化し分析する方法を提供する
- サーチエンジンプロバイダーは、ウェブサイトのランキングを進めるに連れ、ユーザー エクスペリエンスに重点を置くようになった
- ユーザーがいつ、どこに訪れているかを理解することはウェブサイトの最適化作業の優先順位づけに役立つ

Quiz

1. **True or False:** User Experience アプリでは、ウェブサイトの知覚的なパフォーマンスを分析することができる
2. **True or False:** Core Web Vitals は、知覚的なパフォーマンスの分析には寄与しない
3. **True or False:** ビジターのオペレーティングシステム(OS)やブラウザのファミリー、地理的な位置情報に基づいて、ウェブサイトの最適化の優先順位を決めることができる

User Experience app

Lab 4.4



User Experience アプリを探索してみましょう

Agenda

- **Module 1: Getting started**
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理

データ処理と構造化

Module 5

Topics

- パイプライン
- イベントの抽出
- イベントの変換
- イベントのロード

パイプライン

Module 5 Lesson 1



データ処理と構造化

- データは通常非構造の "message" フィールドとしてインジェストされる
- パースしてフィールド群に変換するのがゴール
 - データの集約、可視化を容易にする

```
{  
  "message": "2019-09-29T00:39:02.912Z [Debug] MyApp stopped"  
}
```



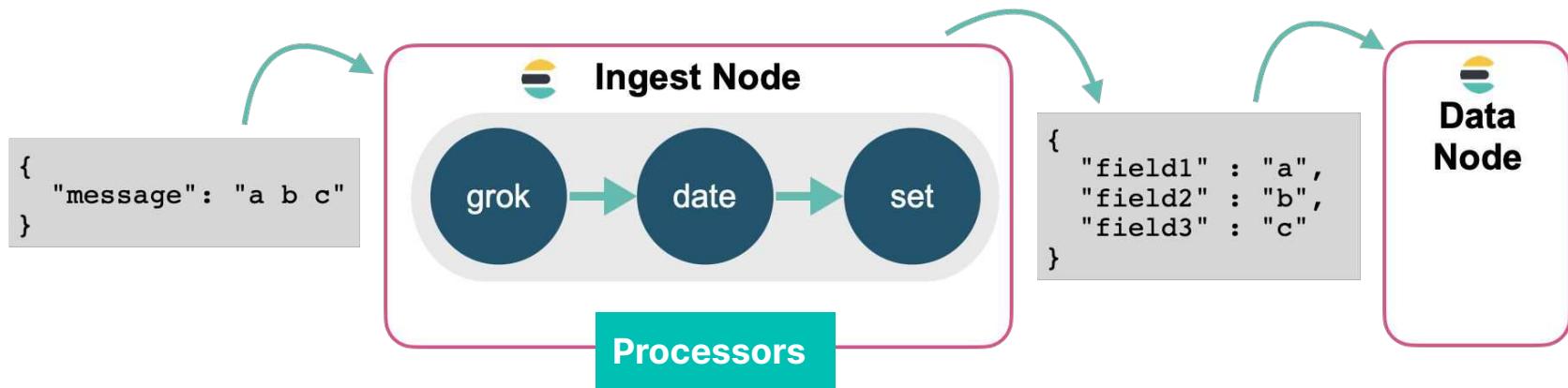
```
{  
  "@timestamp" : "2019-09-29T00:39:02.912Z",  
  "loglevel" : "Debug",  
  "status" : "MyApp stopped"  
}
```

どこでデータを処理、構造化できる？

- Elastic integration は対応するデータソースからのデータを処理し構造化できる
- カスタム、一般的でないアプリケーションでは自分で実装が必要
- 次のいずれかを使う：
 - Elasticsearch ingest pipelines
 - Logstash pipelines

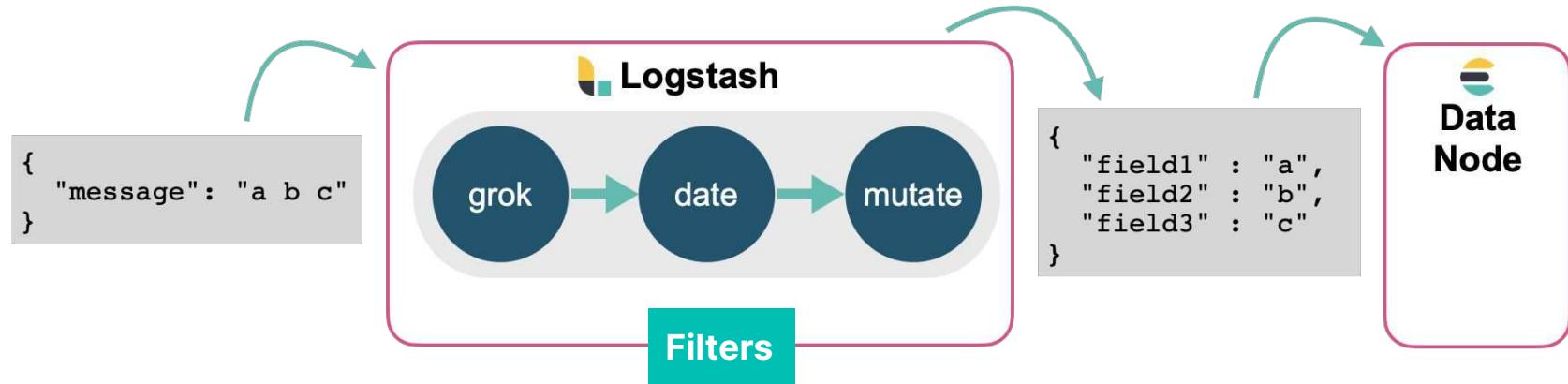
Ingest pipelines

- Elasticsearch **ingest pipeline** はプロセッサのリスト
 - ドキュメントがインデックスされる前に前処理できる
 - 各プロセッサでは渡ってきたドキュメントを処理する
 - Elasticsearch ingest node 上で処理される



Logstash pipelines

- Logstash パイプラインは input, filter, output の組み合わせ
 - Elasticsearch にインデックスするため送信される前にドキュメントを前処理できる
 - 各フィルタで渡ってきたドキュメントを加工できる
 - Logstash 内で処理を実行する



Elasticsearch or Logstash?

- データ処理、構造化などのコンポーネントを使うべき?
- Elasticsearch ingest pipeline は Elasticsearch だけでデータ処理が可能
 - 独自のサーバーを必要とするコンポーネントは追加不要
 - ノードを追加するだけでスケールする
- Logstash は柔軟な ETL ツール
 - あらゆるデータソースから読み取りあらゆるシステムに書き込める
 - 外部システムを使ってデータをエンリッチ (例: RDBMS)

ingest pipeline の作成と管理

- Kibana の **Ingest Pipelines** UI でパイプラインを作成、管理
 - パイプラインの一覧表示、詳細ヘドリルダウン
 - 既存パイプラインを編集、クローン
 - パイプラインを削除

The screenshot shows the Kibana interface with the following details:

- Header:** Stack Management > Ingest Pipelines
- Left Sidebar (Management):**
 - Ingest:**
 - [Ingest Pipelines](#) (highlighted)
 - Logstash Pipelines
 - Data:**
 - Index Management
 - Index Lifecycle Policies
 - Snapshot and Restore
 - Rollup Jobs
 - Transforms
 - Cross-Cluster Replication
 - Remote Clusters
 - Alerts and Insights:**
- Main Content Area (Ingest Pipelines):**
 - Title:** Ingest Pipelines
 - Description:** Use pipelines to remove or transform fields, extract values from text, and enrich your data before indexing.
 - Search Bar:** Search...
 - Buttons:** Reload (green), Create pipeline (blue)
 - Pipeline List:** A table showing five existing pipelines:

Name	Actions
.fleet_final_pipeline-1	...
logs-apm.app-8.2.0	...
logs-apm.error-8.2.0	...
logs-mysql.error-1.2.1	...

新しいパイプラインの作成

- パイプラインの名前と説明を設定

Create pipeline

 [Create pipeline docs](#)

Name

A unique identifier for this pipeline.

X Add version number

Description

A description of what this pipeline does.

Name

Description (optional)

プロセッサを追加

- 必要なだけプロセッサを追加
- オプションで on-failure プロセッサを追加
- サンプルドキュメントでパイプラインをテスト

Processors

Add your first processor

Use processors to transform data before indexing. [Learn more.](#)

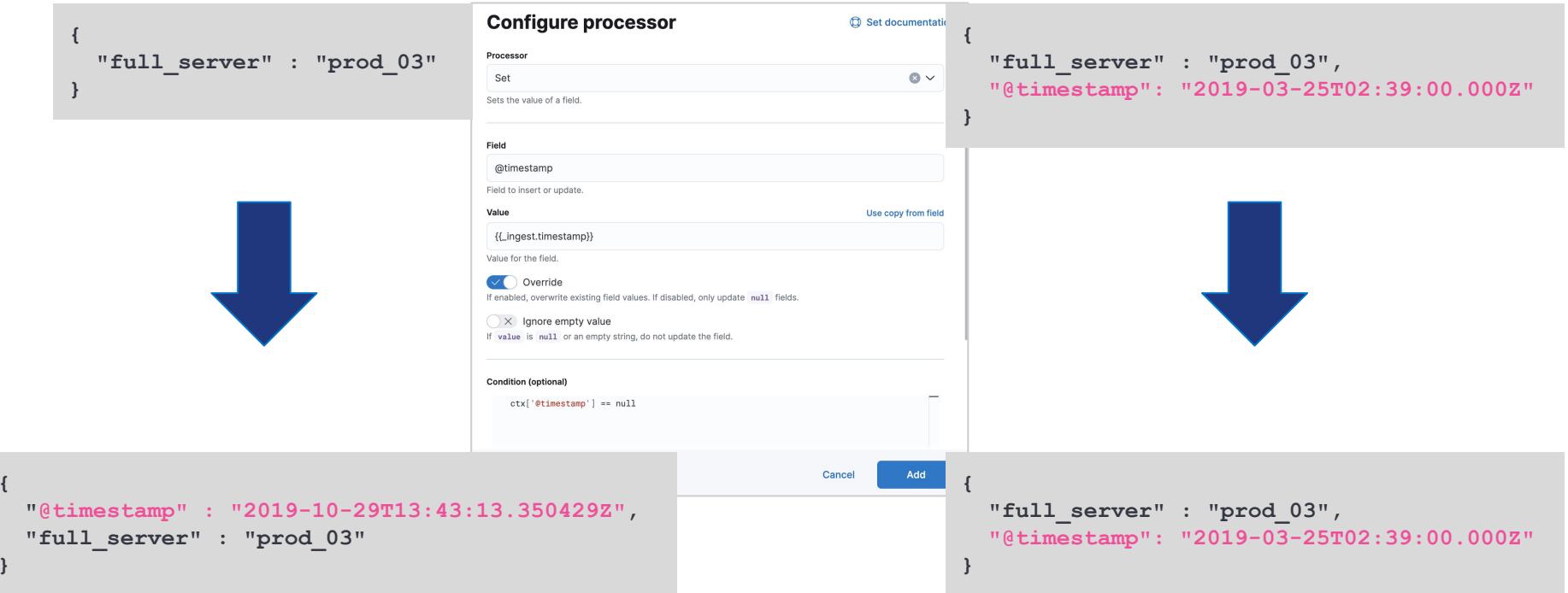
[Add a processor](#)

[!\[\]\(b0a160aca7a8a5859210595467e5caa3_img.jpg\) Import processors](#)

[Create pipeline](#) [Cancel](#) [Show request](#)

条件付きの処理

- 各プロセッサで実行条件を設定できる



Ingest pipeline の失敗

- Ingest pipeline 内でエラーが発生すると、処理が停止し、そのドキュメントはリジェクトされる
- メッセージがパースできない場合に発生する可能性がある、なぜ?
 - 不正なメッセージが渡ってきた?
 - ログ行の構造が変わった?
 - 予期せぬ、未知のログが渡ってきた?
- これらもうまく処理したいでしょう

パイプラインのエラーハンドリング

- on-failure プロセッサを追加し、パイプラインエラー時に異なるプロセッサでドキュメントを処理

パスできなかったドキュメントは "parsefailures" インデックスに送信され、後で確認できる

Configure on-failure processor x

[Set documentation](#)

Processor x v

Set

Sets the value of a field.

Field _index

Field to insert or update.

Value parsefailures [Use copy from field](#)

Value for the field.

Override

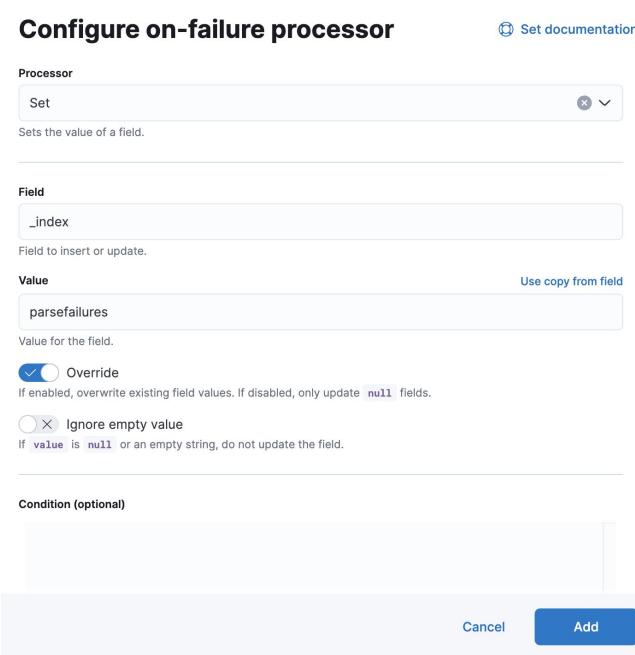
If enabled, overwrite existing field values. If disabled, only update `null` fields.

Ignore empty value

If `value` is `null` or an empty string, do not update the field.

Condition (optional)

Cancel Add



プロセッサのエラーハンドリング

- On-failure ハンドラはプロセッサ単位でも設定できる
- 三つのドットをクリックして **Add on failure handler** を選択

Processors [Import processors](#)

Use processors to transform data before indexing. [Learn more.](#)

⊕ • **Set** Sets value of "@timestamp" to "{{_ingest.timestamp}}"

[+ Add a processor](#)

Failure processors

The processors used to handle exceptions in this pipeline. [Learn more.](#)

⊕ • **Set** Sets value of "_index" to "parsefailures"

[+ Add a processor](#)

Test pipeline: [Add documents](#)

[Duplicate this processor](#)

[Add on failure handler](#)

[Delete](#)

パイプラインをテスト

- パイプラインを保存する前に動作を確認

Test pipeline: [Add documents](#)

- 任意のドキュメントを `_source` に記述することも、インデックスからドキュメントを選択することもできる

Test pipeline

Documents Output

Provide documents for the pipeline to ingest. [Learn more.](#)

> Add a test document from an index

Documents

```
[  
  {  
    "_index": "index",  
    "_id": "id",  
    "_source": {  
      "mysql": {  
        "status": {  
          "max_used_connections": 15  
        }  
      }  
    }  
  },  
  {  
    "_index": "index",  
    "_id": "id",  
    "_source": {  
      "mysql": {  
        "status": {  
          "max used connections": 4  
        }  
      }  
    }  
  }]
```

Clear all

Use JSON format: [{"_index": "index", "_id": "id", "_source": {"foo": "bar"}}]

▷ Run the pipeline

パイプラインを利用する

- デフォルトパイプラインを設定する

```
PUT my_index
{
  "settings": {
    "index": {
      "default_pipeline": "my_pipeline"
    }
  }
}
```

- リクエストで指定されたパイプラインの後にパイプラインを適用したい場合は **final_pipeline** を使う

Fleet と Elastic Agent 用パイプライン

- Fleet は自動的に integration 用の ingest pipeline を追加する
 - インデックステンプレートを使う
 - Elasticsearch は Fleet の data stream にストリームの名前にマッチするテンプレートを適用する
- Fleet integration では Fleet の ingest pipeline を変更したり、カスタムのパイプラインを使わないこと
 - Fleet の data stream を壊してしまうことがある
- Fleet は **Custom Logs** integration 用のパイプラインは提供しない
 - インデックステンプレートや、カスタム設定経由で、安全にカスタムログインテグレーション向けのパイプラインを指定できる

Custom Logs の設定

- **Dataset name** は対象のデータセットの名前を指定
- **Custom Configurations** の **pipeline** ポリシー設定でカスタムのパイプラインを指定

Custom log file

Collect log files

Collect your custom log files.

Fleet は logs-<dataset>-default data stream に新しいデータを追加する

Log file path
/var/log/mysql/mysql.log

Add row Path to log files to be collected

Advanced options

Dataset name
mysql.general

Set the name for your dataset. Changing the dataset will send the data to a different index. You can't use - in the name of a dataset and only valid characters for Elasticsearch index names.

Custom configurations
pipeline: my_mysql_pipeline

Optional

Summary:

パイプライン

Module 5 Lesson 1



Summary

- パイプラインで Elasticsearch にインデックスされる前にドキュメントを加工できる
- データ処理、構造化は Logstash か Elasticsearch で実行できる
- ドキュメントの内容に応じて条件付きでドキュメントを加工できる
- On-failure プロセッサはパイプラインが失敗したときに別のプロセッサセットへとドキュメントを送信する
- On-failure ハンドラをプロセッサ単位に設定することもできる

Quiz

1. Elastic Stack 内のどこでインデックス前にドキュメントを加工できる?
2. **True or False:** Ingest pipeline 内でエラーが発生したドキュメントは、永久に失われる
3. **True or False:** 条件を使えばドキュメントを処理すべきかチェックできる

パイプライン

Lab 5.1

Ingest pipeline を作成しましょう



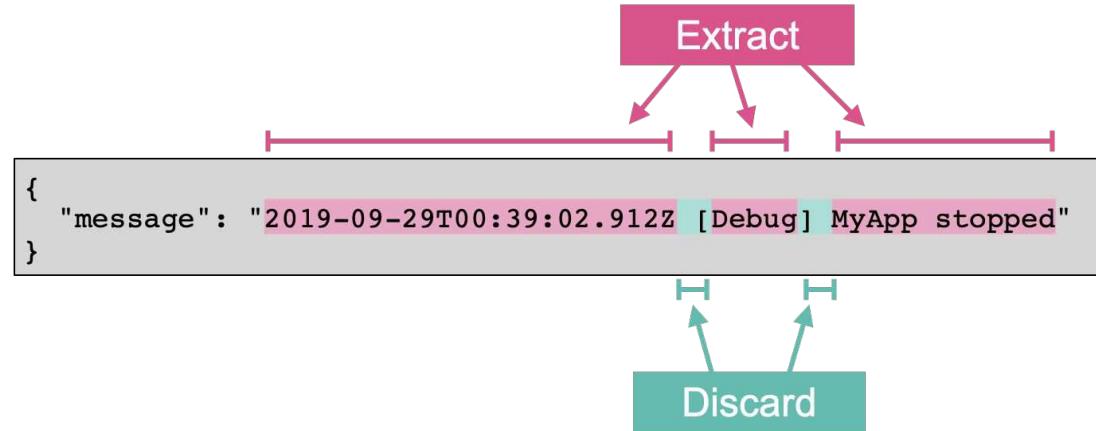
イベントの抽出

Module 5 Lesson 2



非構造データを構造化する

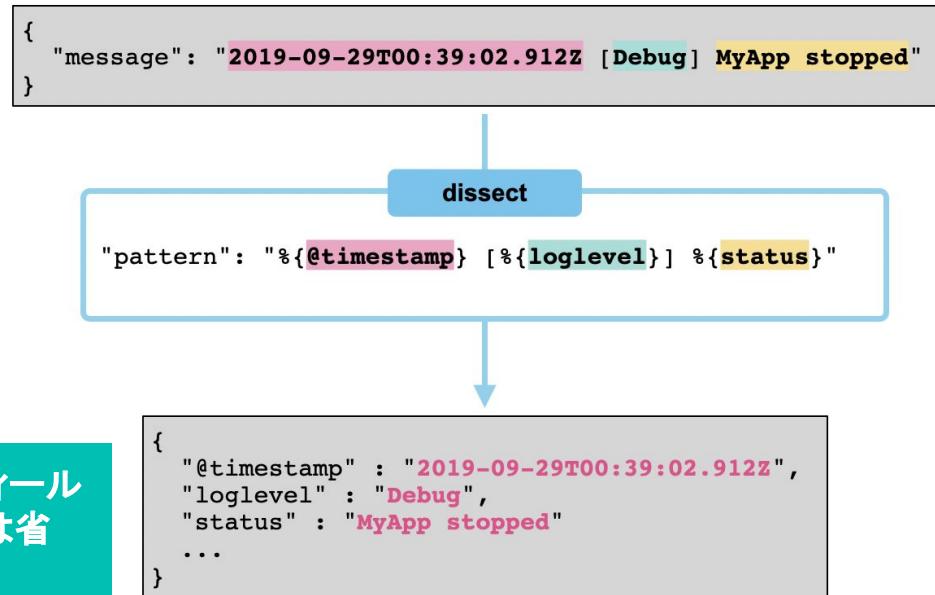
- パイプラインの一般的なユースケースはメッセージの構造化:



- "**MyApp stopped**" はスペースを含んでいて単純に空白で分割できない
- メッセージには【 や 】のように無駄な文字もある

Dissect プロセッサ

- パターンでメッセージの構造を伝える
 - そして %{field} シンタックスでフィールドを抽出

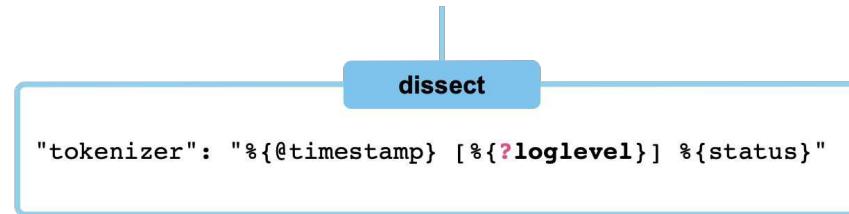


もともとの *message* フィールドは残っているがここでは省略

フィールドをスキップ

- `%{}` でフィールドをスキップ
- もしくは `%{?field}` で dissect のパターンを読みやすくしつつ、フィールドをスキップする
-

```
{  
  "message": "2019-09-29T00:39:02.912Z [Debug] MyApp stopped"  
}
```

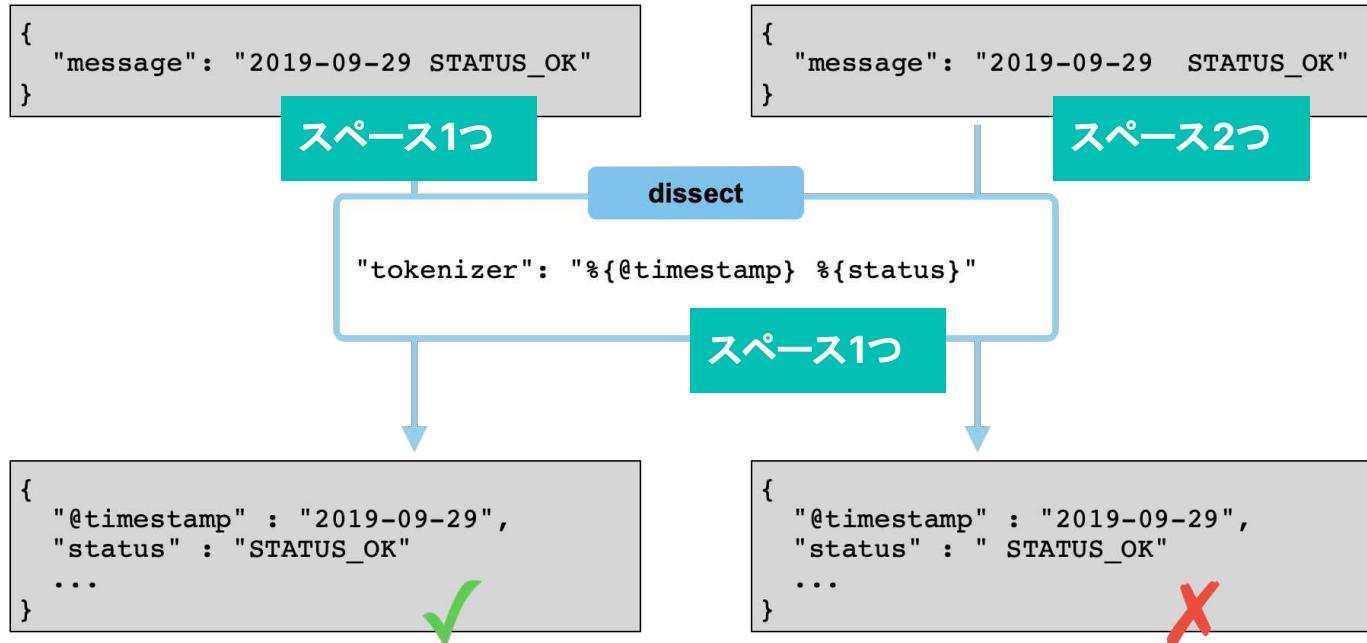


`loglevel` はスキップされ、出力されない

```
{  
  "@timestamp" : "2019-09-29T00:39:02.912Z",  
  "status" : "MyApp stopped",  
  ...  
}
```

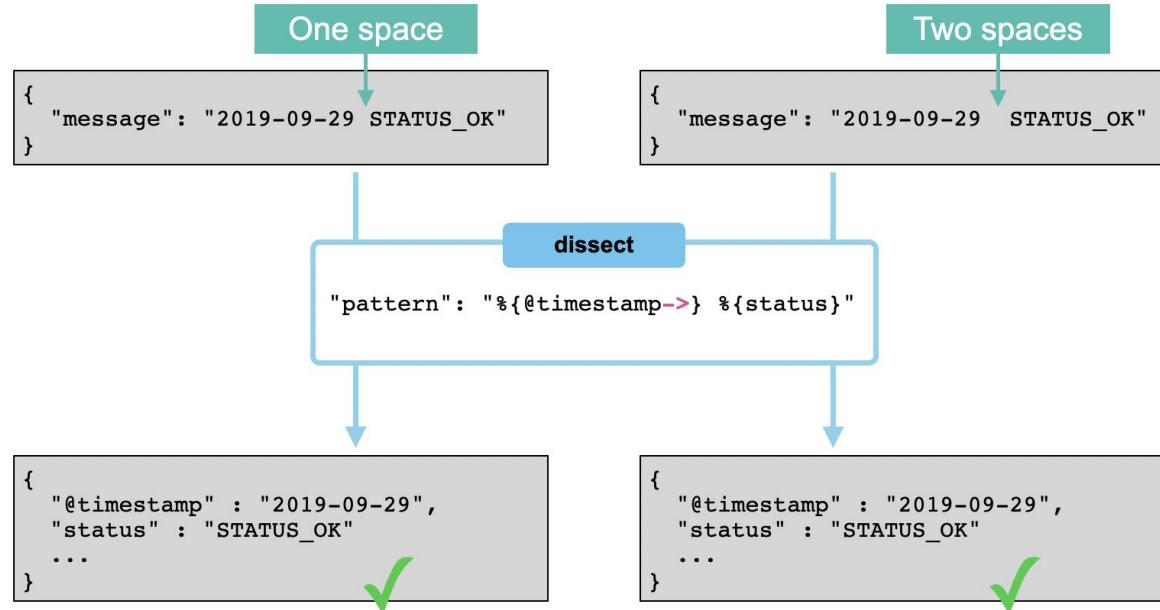
空白の扱い

- Dissect はとても厳格: パターンの文字列は全てマッチする必要がある



空白の扱い

- 右側パディング修飾子 -> で空白を無視できる



フィールドの結合

- `%{+field}` で複数の値をひとつのフィールドに結合できる
- スペースを使い結合された値を分割

```
{  
  "message": "Oct 29 00:39:02 Debug MyApp stopped"  
}
```

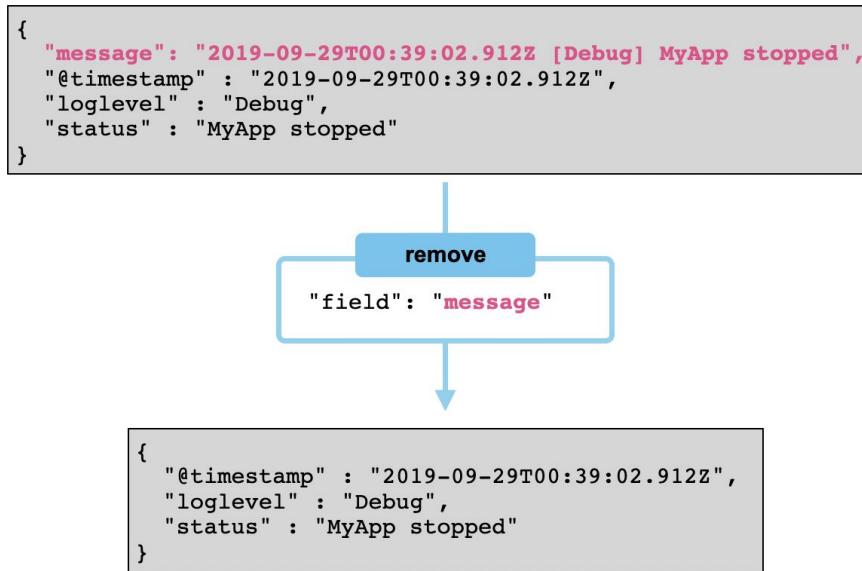
dissect

```
"tokenizer": "%{@timestamp} %{+@timestamp} %{+@timestamp} %{loglevel} %{status}"
```

```
{  
  "@timestamp" : "Oct 29 00:39:02",  
  "loglevel" : "Debug",  
  "status" : "MyApp stopped"  
  ...  
}
```

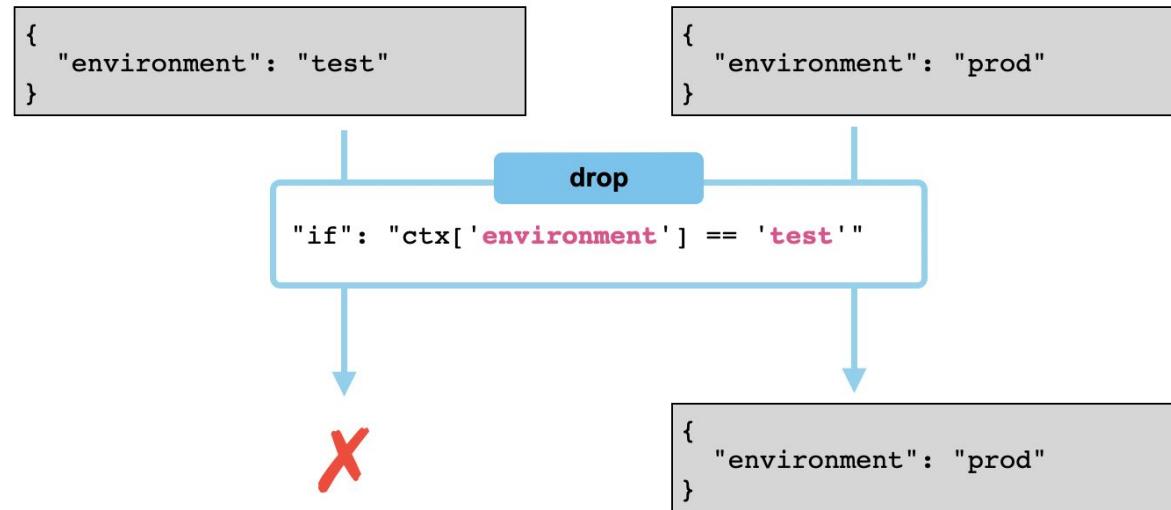
remove プロセッサ

- 不要なフィールドを除去できる
 - 例えば、すでにパース済みのもとの **message** フィールド



ドキュメントを捨てる

- **drop** プロセッサでドキュメントを完全に捨てて Elasticsearch に保存しないようにする



Summary: **Extracting events**

Module 5 Lesson 2



Summary

- **Dissect** プロセッサはフォーマットを指定して対象のフィールドを抽出し構造化ことができる
- Dissect は空白文字を厳格に扱う
- 複数の値を单一のフィールド値として結合し抽出することができる
- **Remove** プロセッサは不要なフィールドをドキュメントから除去する
- **Drop** プロセッサは条件判定を設定してドキュメントを無視することができる

Quiz

1. **True or False:** Dissect は空白文字を意識しない
2. **True or False:** Drop プロセッサは不要なフィールドをドキュメントから除去する
3. 以下のデータに対する dissect パターンを考えましょう:

```
{  
  "message": "2019-09-29 STATUS_OK Server started normally."  
}
```

イベントの抽出

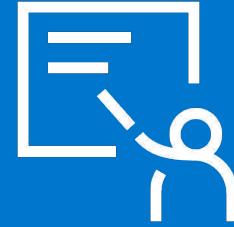
Lab 5.2

Extract events



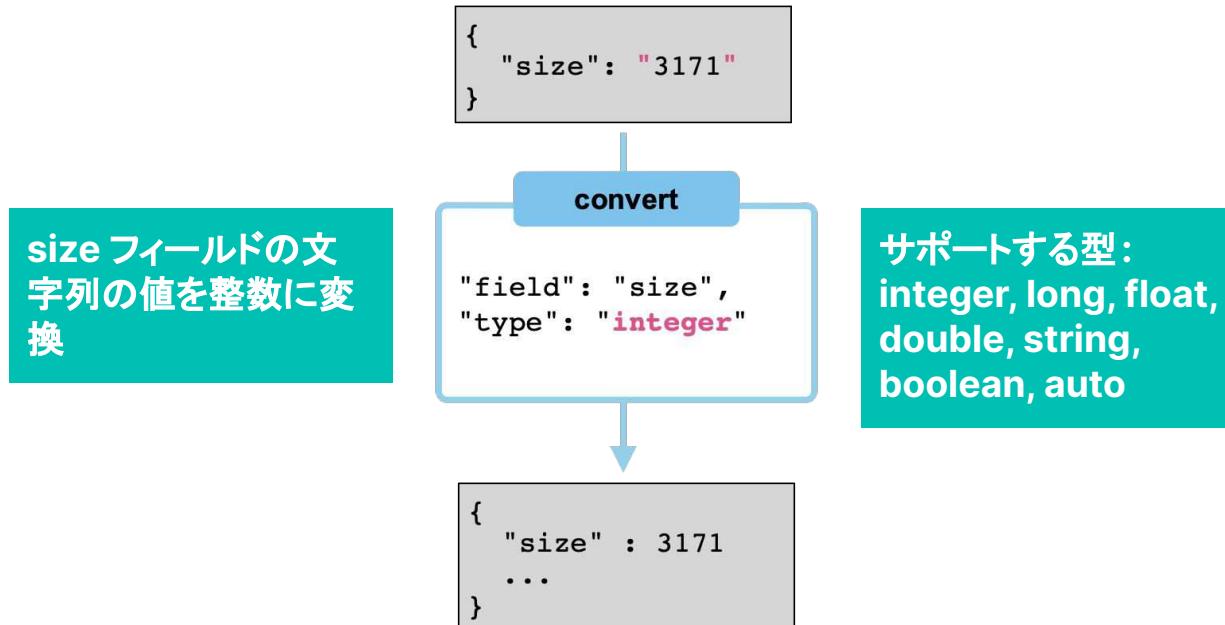
イベントの変換

Module 5 Lesson 3



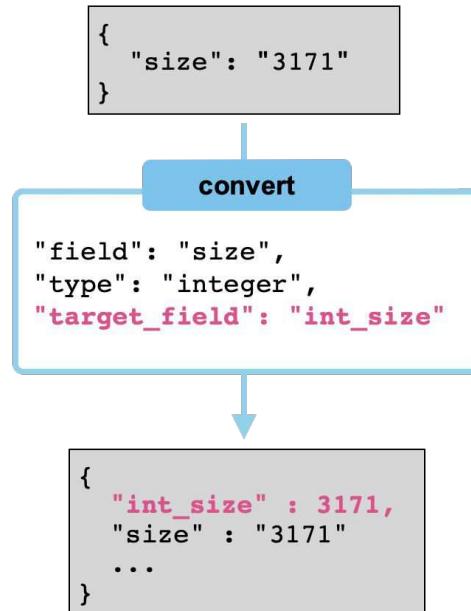
文字列の変換

- **dissect** のようなプロセッサの出力は文字列
- 文字列を他の型に変換するには **convert** プロセッサを使う



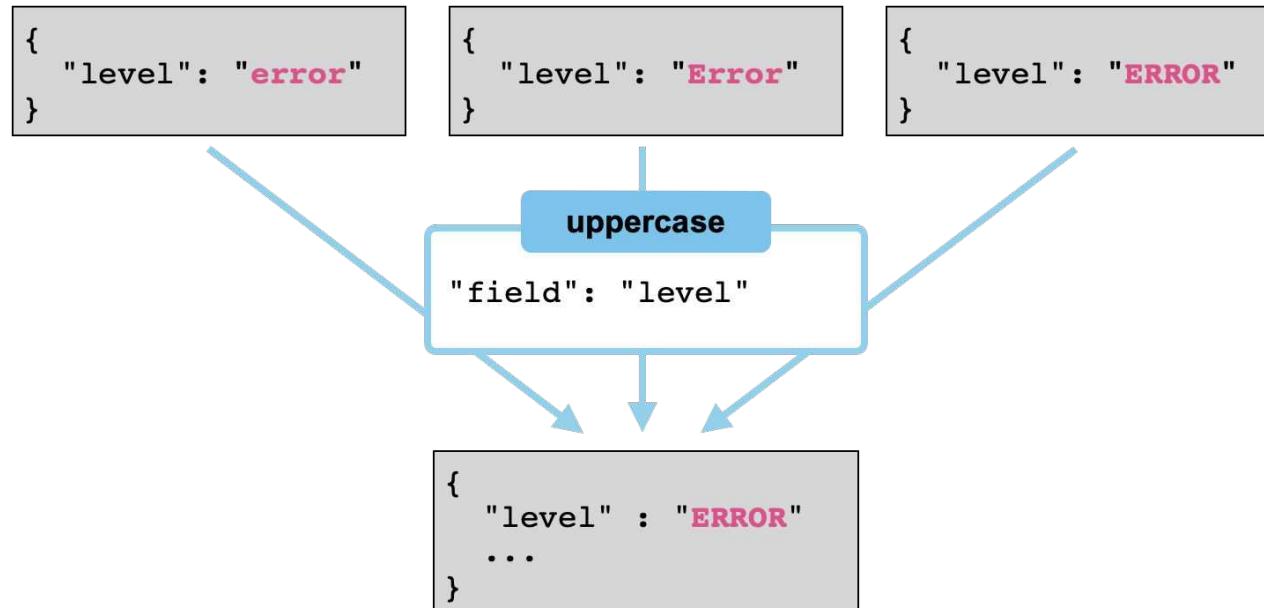
Target フィールド

- **convert** プロセッサは値を同じフィールド上で変換する
- 変換後の値を指定したフィールドに設定することもできる



大文字小文字を揃える

- 文字列を大文字小文字に変換できる
- Elasticsearch では **uppercase** と **lowercase** プロセッサを使う



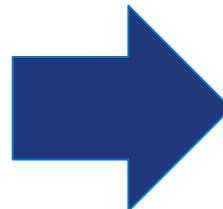
日付の扱い

- 日付はさまざまなフォーマットでやってくる
- これらは共通のフォーマットに正規化すると扱いやすい
 - 簡単に Elasticsearch にインデックスできる
 - Elasticsearch からデータを取得するアプリケーションでも簡単に処理できる

25-03-2019 03:39:00+01:00

03/25/2019 03:39

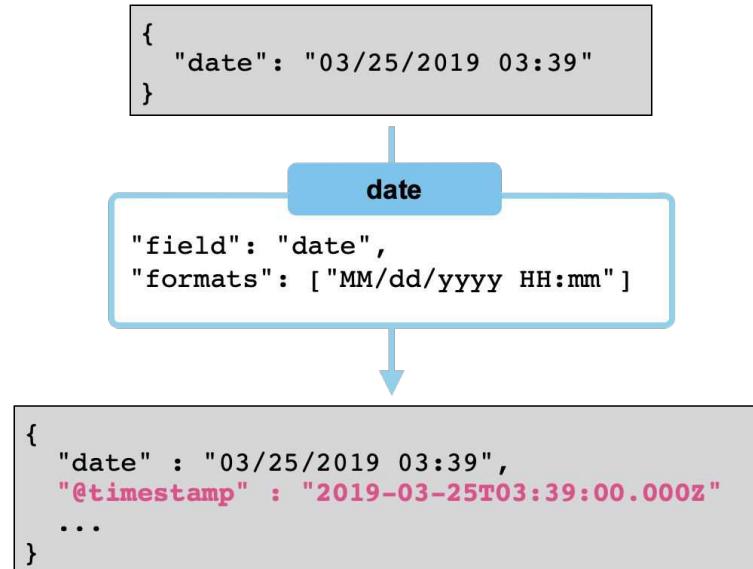
lundi 25 mars 2019 03 h 39 CET



2019-03-25T02:39:00.000Z

date プロセッサ

- **date** プロセッサで日付文字列をパースする
 - **@timestamp** フィールドの値にパースした日付を設定する
 - 値は *ISO-8601* 標準日付フォーマットとなる



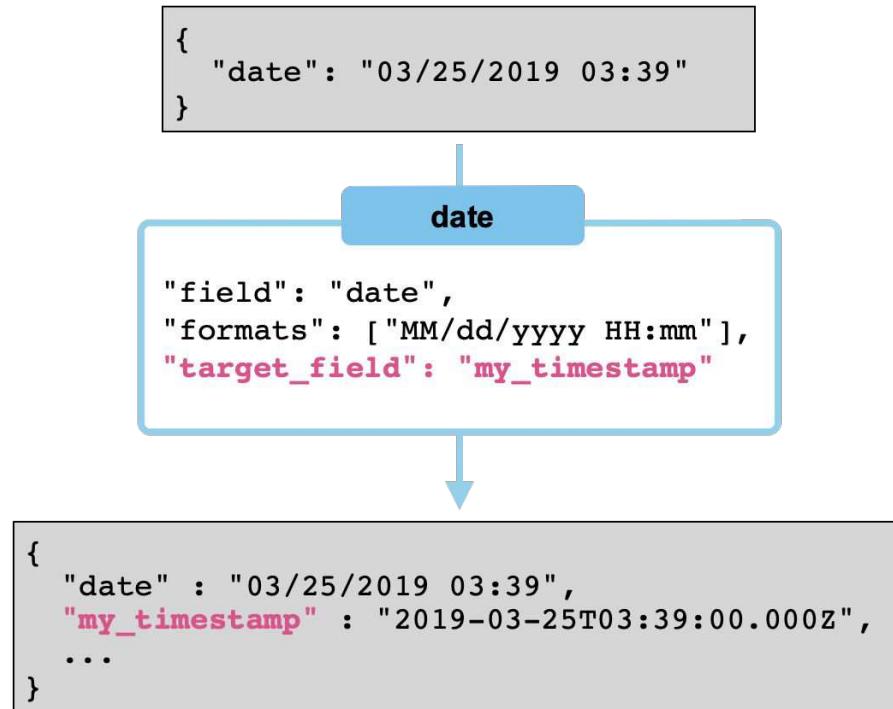
日付フォーマット

- Java の [DateTimeFormatter](#) シンタックスで入力と出力のフォーマットを指定

Symbol	Meaning	Example
y	year	2019
M	month of year	July; Jul; 07
d	day of month	10
a	half-day of day	AM; PM
H	hour of day (0-23)	0
m	minute of hour	30
s	second of minute	15
S	fraction of second	978

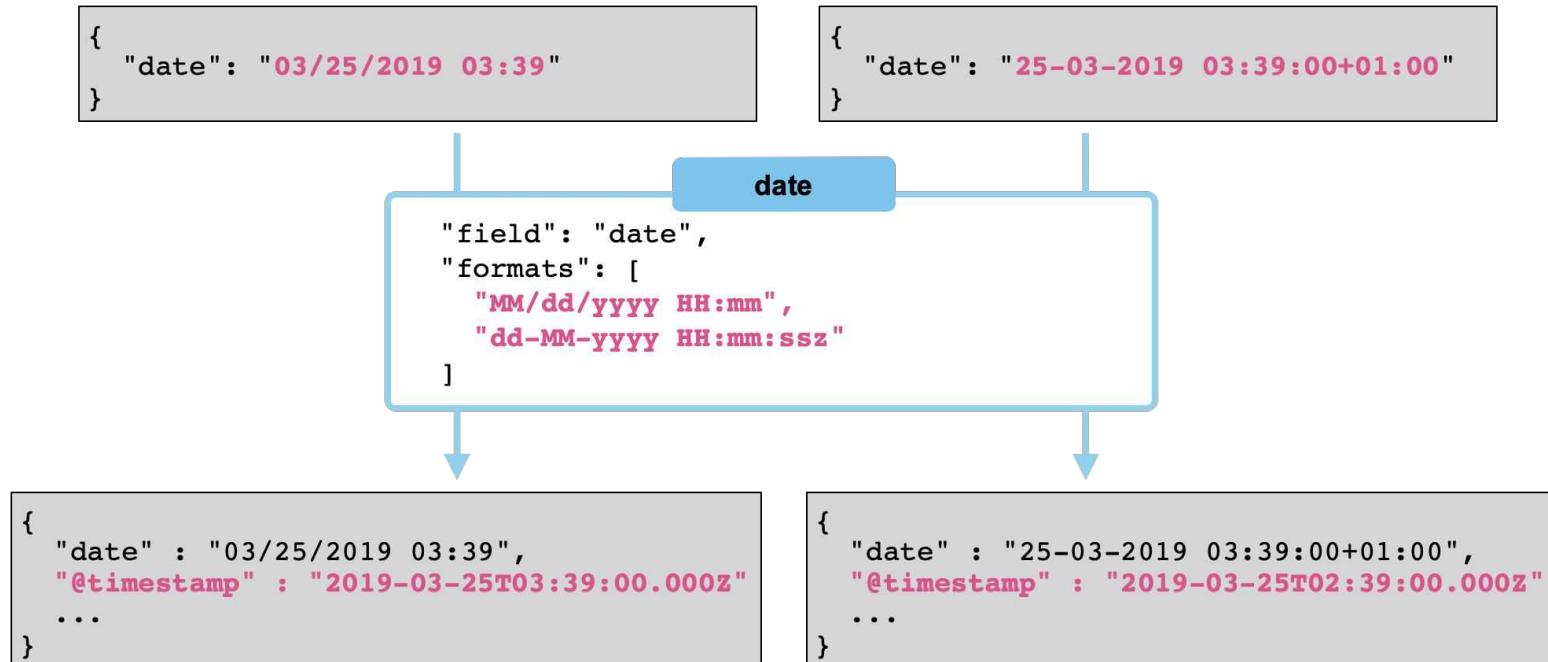
Target フィールド

- 別の出力先フィールドを指定することもできる



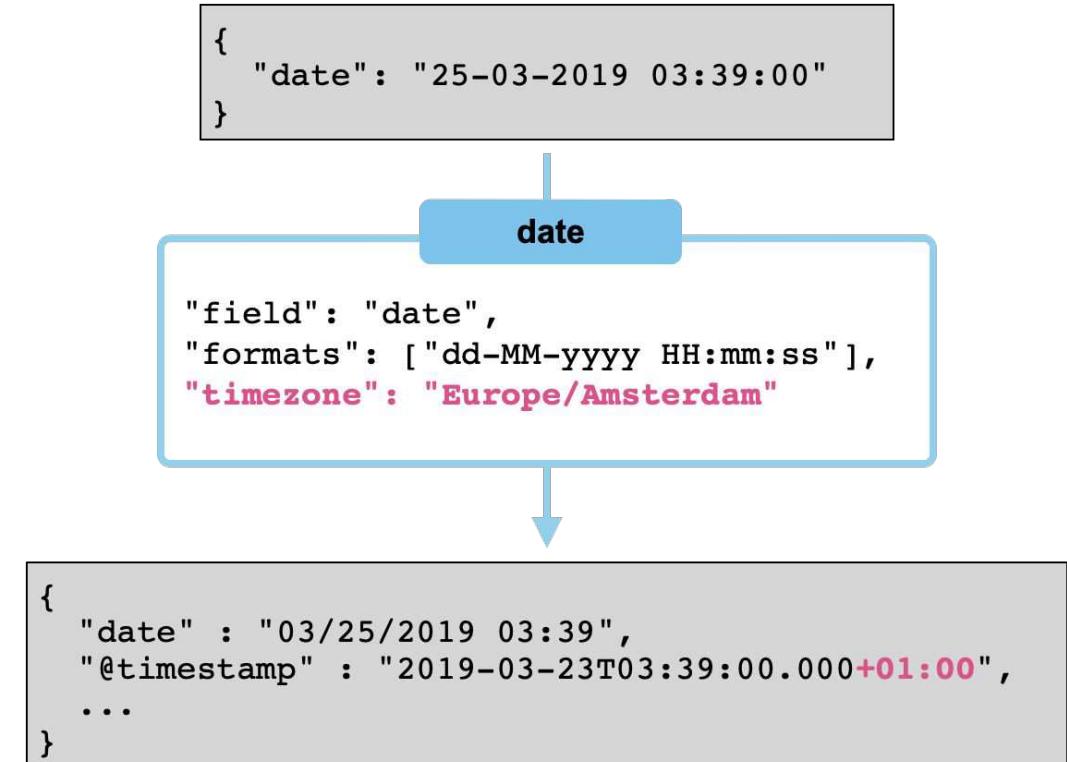
複数の日付フォーマットにマッチ

- 複数のフォーマットを配列で指定可能



タイムゾーン

- タイムゾーンを指定しない場合、タイムスタンプは UTC となる
- タイムゾーンを明示するともできる



ロケール

- locale 設定で日付の言語を指定できる

A date in French

```
{  
  "date": "lundi 25 mars 2019 03 h 39 CET"  
}
```

date

```
"field": "date",  
"formats": ["EEEE d MMMM yyyy HH' h 'mm z"],  
"locale": "fr-fr"
```

```
{  
  "date": "lundi 25 mars 2019 03 h 39 CET"  
  "@timestamp": "2019-03-25T02:39:00.000Z",  
  ...  
}
```

Summary: イベントの変換

Module 5 Lesson 3



Summary

- 文字列は **convert** プロセッサで別の型に変換できる
- **Uppercase** と **lowercase** プロセッサで文字列の大文字小文字を変更可能
- **Date** プロセッサは異なる日付フォーマットを統一できる
- タイムゾーンが指定されない場合、**date** プロセッサで扱うタイムスタンプは UTC となるが、タイムゾーンを明示することもできる
- **Date** プロセッサの **locale** 設定で日付の言語を指定する

Quiz

1. **True or False:** Elasticsearch の date プロセッサはタイムゾーンが明示されていない場合、サーバーのローカルタイムゾーンとして処理する
2. 次の日付を date プロセッサでパースする際どんなフォーマットを指定すればよいか: "12/31/2019"
3. 日付の言語を date プロセッサに伝えるための設定名は?

Transforming events

Lab 5.3

イベントの変換



イベントのロード

Module 5 Lesson 4



ロード時のタスク

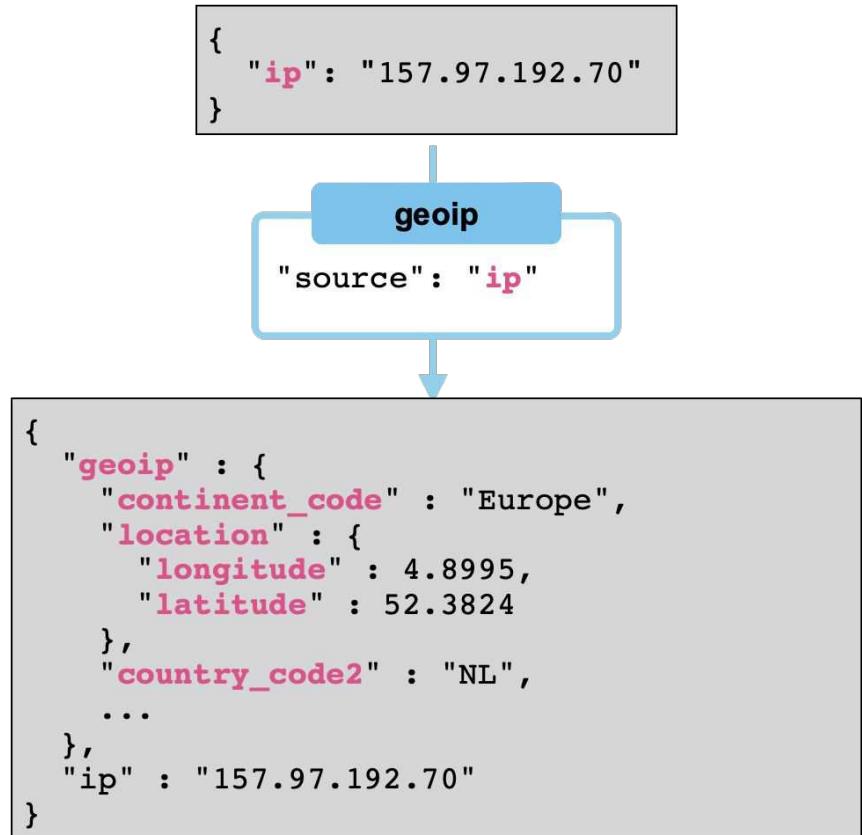
- 位置情報の詳細
- User agent の詳細
- Lookups

位置情報

- ユーザーの位置情報を扱いたい場合がある
 - 顧客はどこに住んでいる?
 - どの国からセキュリティインシデントが引き起こされた?
 - マーケティングキャンペーンが最もうまくいった都市は?
- これらの位置情報をログにエンリッチしたい
 - ログ内の IP アドレスを元に

geoip プロセッサ

- geoip プロセッサは IP アドレスを元に位置情報をルックアップする
- geoip フィールドで geo 情報を付与する

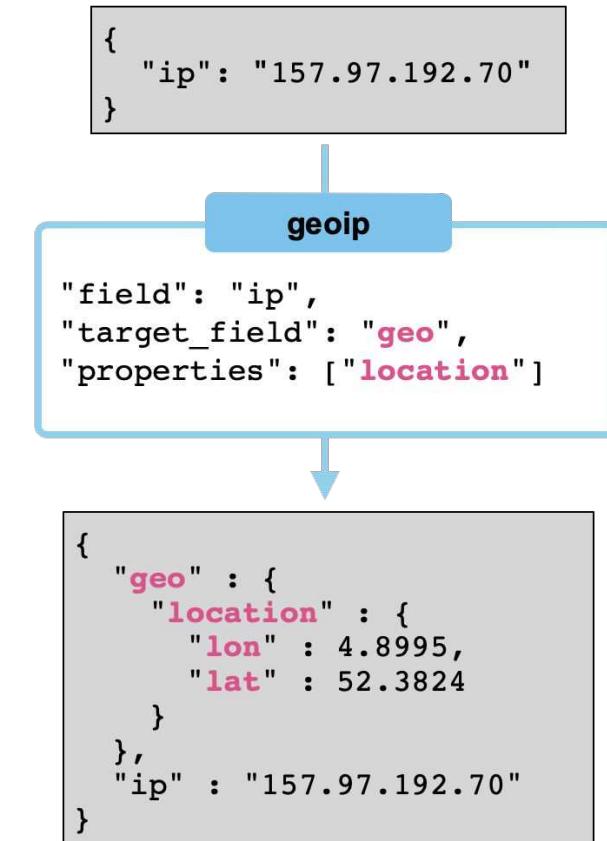


geoip プロセッサが動作する仕組み

- GeoLite2 データベースをルックアップする
 - IP アドレスで参照
 - and populates fields with the results
 - 結果をフィールドに設定
- Elasticsearch には三つの [Maxmind 社の GeoLite2 データベース](#) が付属、ライセンスは CCA-ShareAlike 4.0
 - GeoLite2 **City**, GeoLite2 **Country**, GeoLite2 **ASN**
- データベースは Elasticsearch のローカルにデプロイされている
 - ネットワーク呼び出しが不要なので高速
 - DB の更新は自動的に Elastic からダウンロードされる
- 独自の IP データベースを生成することもできる
 - 参照: github.com/mteodoro/mmutils

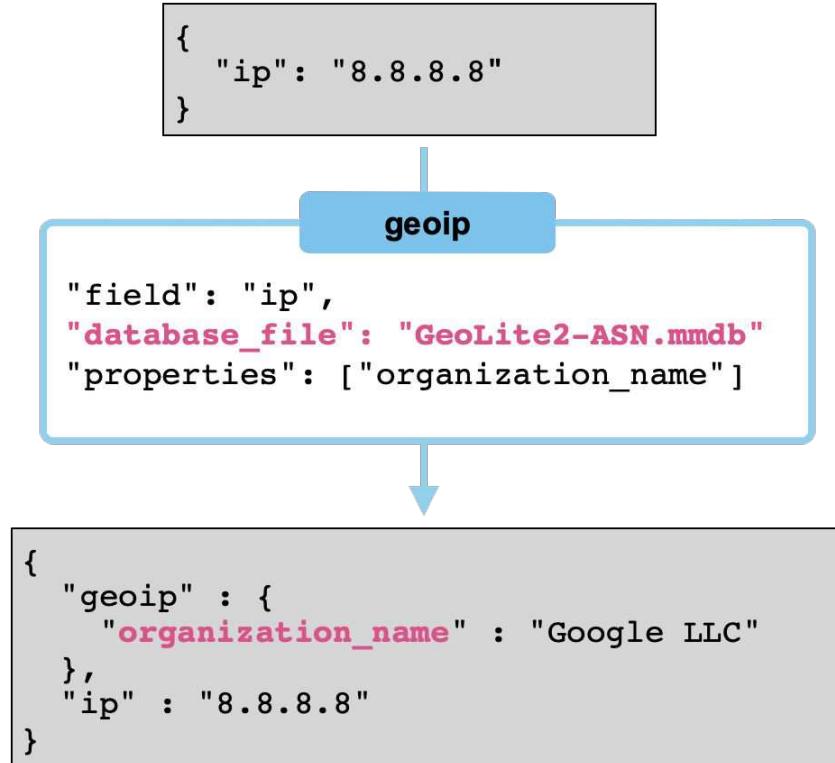
設定オプション

- **target_field** 設定で保存先のフィールドを指定
- **properties** で追加される属性を絞り込むことができる



別のデータベースを使う

- **database_file** 設定で別のデータベースを利用できる
- このモジュールには
GeoLite2-City.mmdb,
GeoLite2-Country.mmdb,
GeoLite2-ASN.mmdb ファイルが同梱されている



User agent

- Web リクエストを送信するブラウザは *user agent* 文字列で自身を表現
- User agent 文字列には多くの情報がある: ブラウザ、OS、バージョン ...などなど
- しかし、user agent 文字列は難解

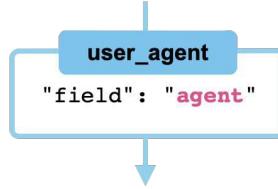
```
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6)  
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/78.0.3904.70  
Safari/537.36
```

- いったいこれは Mozilla, Chrome, Safari?

user_agent プロセッサ

- user_agent プロセッサで user agent 文字列をパースできる

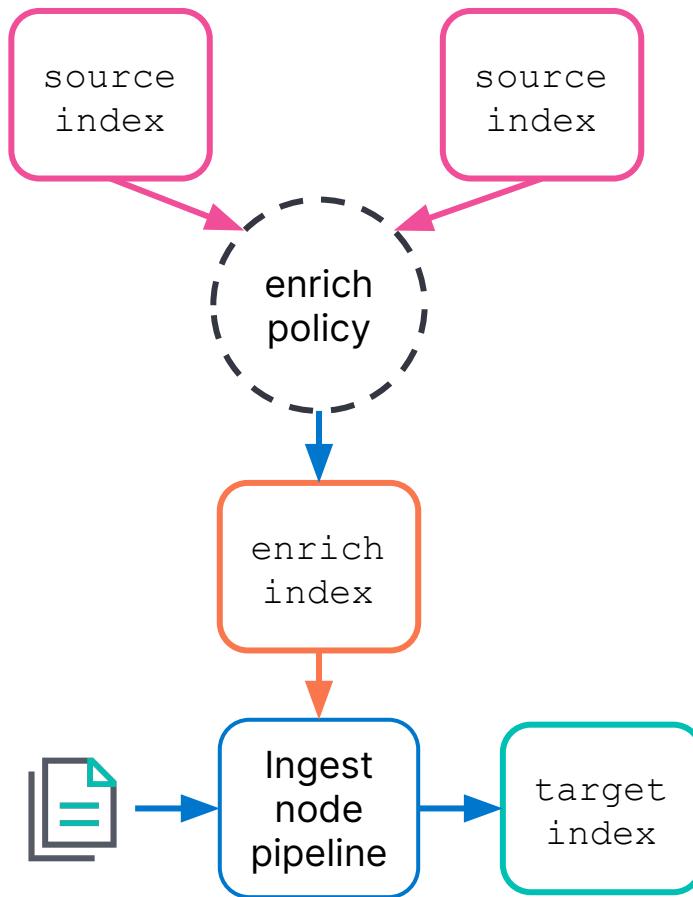
```
{  
    "agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/  
537.36 (KHTML, like Gecko) Chrome/78.0.3904.70 Safari/537.36"  
}
```



```
"user_agent" : {  
    "name" : "Chrome",  
    "original" : "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6) AppleWebKit/537.36  
(KHTML, like Gecko) Chrome/78.0.3904.70 Safari/537.36",  
    "os" : {  
        "name" : "Mac OS X",  
        "version" : "10.14.6",  
        "full" : "Mac OS X 10.14.6"  
    },  
    "device" : {  
        "name" : "Other"  
    },  
    "version" : "78.0.3904"  
}  
...
```

データのエンリッチ

- **enrich** プロセッサを使うと Elasticsearch 内の他のデータでデータをエンリッチできる
- 他のインデックスをクエリできる
 - その結果を投入されるデータに追加
- enrich プロセッサを利用するには:
 - step 1: enrich ポリシーを設定
 - step 2: enrich index を作成
 - step 3: enrich プロセッサを使った ingest pipeline を作成



Step 1: enrich ポリシーの設定

```
PUT _enrich/policy/address-policy
{
  "match": {
    "indices": "addresses",
    "match_field": "postal_code",
    "enrich_fields": ["coordinates"]
  }
}
```

ポリシー名

match か geo_match を設定

ひとつ以上の source index

source index の参照フィールド

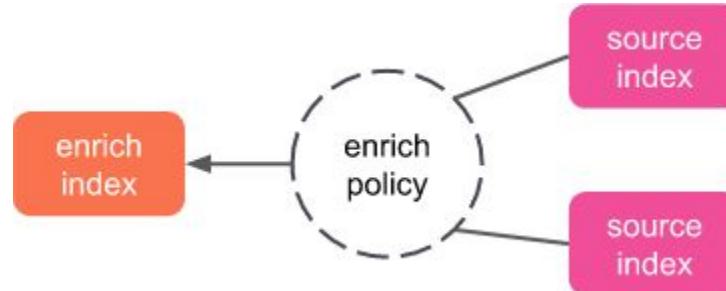
投入するドキュメントに source index
から enrich するフィールド

Step 2: enrich index の作成

- enrich ポリシーを実行して enrich index を作成:

```
POST _enrich/policy/address-policy/_execute
```

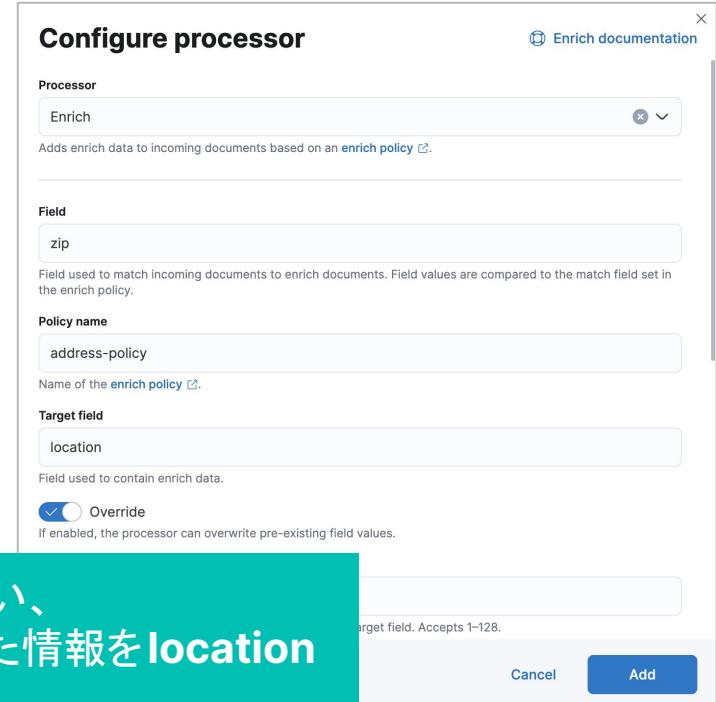
- enrich index は:
 - 簡素化されたシステムインデックス
 - ポリシーの source index から作成される
 - プロセッサにより参照され、投入するドキュメントを enrich するために利用



Step 3: enrich プロセッサを使った ingest pipeline を作成

- enrich プロセッサで以下を指定:
 - 利用する **enrich policy**
 - ポリシーの `match_field` とマッチさせる入力ドキュメントの **field**
 - 入力ドキュメントに追加される **target field**

このプロセッサは入力ドキュメントの **zip** フィールドを使い、**address-policy** の enrich index を検索し、マッチした情報を **location** フィールドに追加する



Summary:

イベントのロード

Module 5 Lesson 4



Summary

- **Geoip** プロセッサを使うと、IP アドレスをもとにドキュメントを geo 情報でエンリッチできる
- User agent 文字列は **user_agent** プロセッサでパースできる
- **Enrich** プロセッサは他のインデックスのデータでドキュメントを enrich できる

Quiz

1. **True or False:** リクエストの IP アドレスがあれば、そのリクエストが送信された都市の名前を追加できる
2. User agent スtringing のパーサーをデプロイするにはどのくらいかかる:
 - a. 2 分
 - b. 2 日
 - c. 2 週間
3. Enrich index はどのように作成する?

イベントのロード

Lab 5.4

イベントをロードしましょう



Agenda

- Module 1: Getting started
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- **Module 6: オブザーバビリティデータからアクションへ**
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理



オブザーバビリティデータから アクションへ

Module 6

Topics

- Machine Learning
- カスタム ML ジョブ
- アラート

Machine Learning

Module 6 Lesson 1



大量のデータの分析

- オブザーバビリティデータは何が、なぜ、いつ発生したのかを知るために非常に有益
- ただし、Elastic エージェントは Elasticsearch に大量のデータをインデックスする
- この大量のデータは俯瞰的な視野を維持することを困難にする
 - 75% のデータは実は全く使われていない！
- 1,000 を超えるソースから成るオブザーバビリティデータの恩恵をフルに享受するためには、モニタリングの自動化が不可欠

Machine Learning

- Machine Learning は Elasticsearch クラスタのデータを使って、教師なしのオブザーバビリティモデルを作成する
- 作成したモデルに対して最新のデータや代替となるデータを投入し、データの異常やデータの傾向を把握することで、オブザーバビリティデータの分析を自動化する
- Machine Learning は**異常検知と異常値スコアリング**の両方を行う

データに関する要件

- Elastic Machine Learning を適用する前に、次の3つの要件を考慮する必要がある
 - 時系列データであること
 - データは ML のユースケースに必要な KPI を含むこと
 - 必要なデータが一つの場所に集約されていること
- オブザーバビリティデータはこの要件をすべて満たしている

時系列的変異 (temporal deviation) に対する異常検知

- 時系列のデータの中から異常を検出するには、まず関心事を KPI として定める
- 例えば:
 - 時間ごとのログの数
 - 時間ごとの 404 エラーの数
 - 時間ごとのディスクの使用率
- KPI を定義したら、この KPI に対して分析のための関数を設定する
 - 例: 平均値、最大値、カウントなど
- この分析用の集計関数と KPI の組み合わせを Detector と呼ぶ
- Detector はシングルメトリック、またはマルチメトリックとして作成できる

過去・現在・未来

- Machine Learning は未来の動向の予測に利用できる
- 予測を行うには、Machine Learning ジョブを作成する
- 予測を作成する際には、対象となるデータの期間を指定する必要がある
 - どのくらいの期間のデータ傾向予測を行うか

集団 (Population) の傾向に対する異常検知

- ポピュレーションモデルでは以下のようなケースを異常として捉える:
 - 時間の経過と共にデータの傾向が変化する場合
 - 同じ集団内の他のデータと振る舞いが異なる場合
- ポピュレーションジョブは同じように振る舞うべき複数のデータエントリーの振る舞いを比較する
- つまり、ポピュレーションジョブは外れ値 (outliers) の検知にフォーカスしている

Elastic machine learning

- Machine Learning を使えば異常値や外れ値の検出、過去の傾向からの将来予測、データ内で注目すべき領域を特定することができる

The screenshot shows the Elasticsearch Machine Learning interface. On the left, there is a sidebar with the following navigation options:

- Analytics
- Discover
- Dashboard
- Canvas
- Maps
- Machine Learning** (highlighted with a pink arrow)
- Graph
- Visualize Library

The main area is titled "Machine Learning" and "Overview". It displays the following information:

- Total machine learning nodes: 1
- Table showing node details:

Name	Total memory	Memory usage
instance-0000000003	1GB	[Progress bar]
- Anomaly Detection** section with the following table:

Group ID	Overall score	Jobs in group	Latest timestamp	Docs processed	Actions
apm	[Colorful bar]	1	May 20th 2022, 13:52	48,580	[Icon]
logs-ui	[Colorful bar]	2	May 20th 2022, 13:52	6,699,314	[Icon]
my_ml_jobs	[Colorful bar]	2	May 20th 2022, 13:52	4,557,674	[Icon]
uptime	No results found	1		0	[Icon]

すぐに使える ML ジョブ

- Kibana には格納されたデータに対応した定義済みの ML ジョブが用意されている

The screenshot shows the Kibana interface for Machine Learning. At the top, there's a navigation bar with three horizontal dots, a 'D' icon, 'Machine Learning', 'Anomaly Detection', and 'Create job'. On the left, a sidebar menu includes 'Machine Learning' (selected), 'Overview', 'Anomaly Detection' (selected), 'Jobs', 'Anomaly Explorer', 'Single Metric Viewer', 'Settings', 'Data Frame Analytics' (selected), 'Jobs', 'Results Explorer', and 'Analytics Map'. The main content area has a title 'Create a job from the data view metrics-*' and a sub-section 'Use preconfigured jobs'. It says 'The fields in your data match known configurations. Create a set of preconfigured jobs.' Below this, there are three cards: 'APM' (detect anomalies in transaction latency, throughput, and failure rate from APM services), 'Metricbeat System' (detect anomalies in Metricbeat System data via ECS), and 'Security: Linux' (detect suspicious activity using ECS Linux events, tested with Auditbeat and the Elastic agent).

Create a job from the data view metrics-*

Use preconfigured jobs

The fields in your data match known configurations. Create a set of preconfigured jobs.

 **APM**
Detect anomalies in transaction latency, throughput and failure rate from your APM services for metric data.

 **Metricbeat System**
Detect anomalies in Metricbeat System data (ECS)

 **Security: Linux**
Detect suspicious activity using ECS Linux events. Tested with Auditbeat and the Elastic agent.

ML ジョブの結果を参照する

- ML ジョブを作成したら、Single Metric Viewer や Anomaly Explorer を使って結果を参照する

New job from data view metrics-*

The screenshot shows the Elasticsearch Machine Learning interface. On the left, the "Job settings" panel is open, showing fields for "Job ID prefix" (set to "apm_tx_metrics") and checkboxes for "Start datafeed after save" and "Use full metrics-* data". A "Create job" button is at the bottom. In the center, the "Jobs" page displays an "Anomaly Detection" job named "apm_tx_metrics" which detects anomalies in transaction latency, throughput, and error percentage for metric data. The main area shows the "Anomaly Detection Jobs" list with one entry:

ID	Description	Processed rec...	Mem...	Job state	Datafeed ...	Latest timestamp
apm-client-ip	my_ml_jobs	4,555,998	ok	opened	started	2022-05-20 13:59:58
apm_tx_metrics	Detects anomalies in transaction latency, throughput and error percentage for metric data.	48,595	ok	opened	started	2022-05-20 13:59:00

A pink oval highlights the three-dot menu icon next to the second row in the table.

Single Metric Viewer

- 実データ、正常値の範囲、正常範囲を超えた異常値を描画するチャート



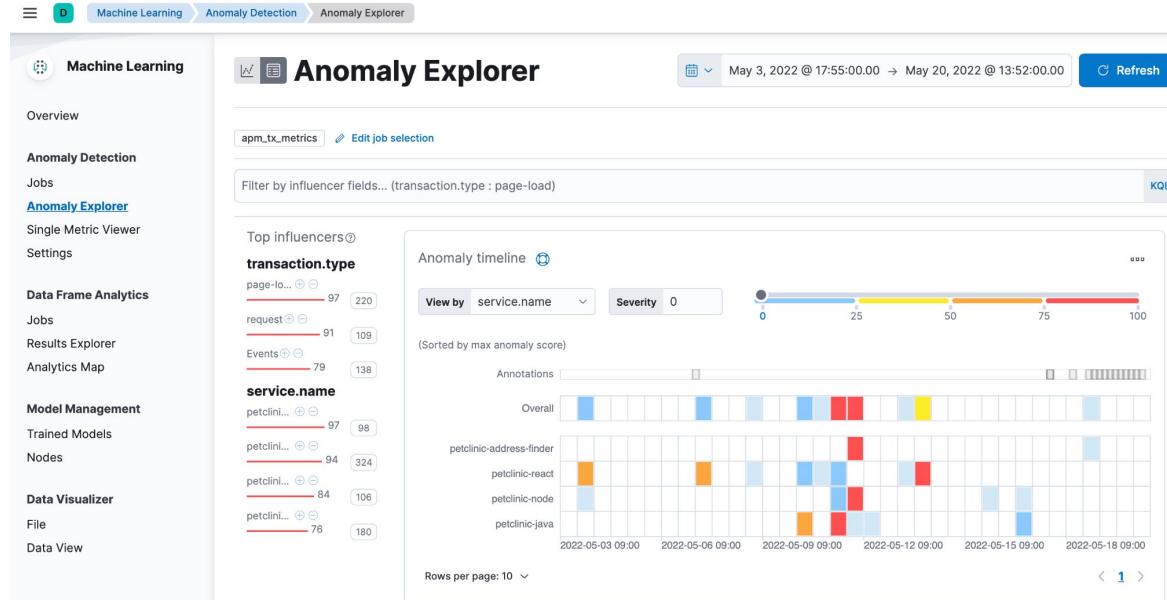
将来の動向を予測する

- Single Metric Viewer では、時系列データに対する特定の時点の傾向予測を行うことができる



Anomaly Explorer

- 最も注意を要する異常値とその異常値に最も影響を及ぼしている変数(influencer)を特定するためのリスト



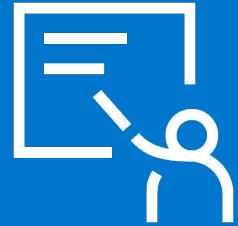
Observability アプリに統合された ML 機能

- Observability のアプリから ML ジョブを直接作成できる

The screenshot shows the Elasticsearch Observability interface. The top navigation bar includes 'Observability', 'APM', 'Settings', and 'Anomaly detection'. The left sidebar has sections for Overview, Alerts, Cases, Logs, Stream, Anomalies, Categories, Metrics, Inventory, Metrics Explorer, APM, Services, Traces, Dependencies, and Service Map. The main content area is titled 'Settings' and has tabs for Agent Configuration, Agent Keys, Anomaly detection (which is selected), Custom Links, Indices, and Schema. Below the tabs, a message says 'To add anomaly detection to a new environment, create a machine learning job. Existing machine learning jobs can be managed in Machine Learning.' It features a call-to-action button 'Create ML Job'. The 'Environments' section lists one environment with the status 'No anomaly detection jobs.' It includes a toggle for 'Show legacy jobs', a 'Manage jobs' button, and a 'Create job' button.

Summary: Machine Learning

Module 6 Lesson 1



Summary

- 分析の手が回らないために大量のデータが未使用のまま放置されている
- Machine Learning はこれらのデータを活用できる
- Machine Learning は既存のデータを使って正常な振る舞いを定義する
- Machine Learning は過去のデータを使って将来を予測することができる
- 母集団分析では、集団内のエントリー同士を比較して異常値を見つける

Quiz

1. **True or False:** Observability アプリから ML ジョブを直接作成できる
2. Machine Learning が異常値に対して行う2つのことは何か？
3. **True or False:** Machine Learning は予測に使うことができる

Machine Learning

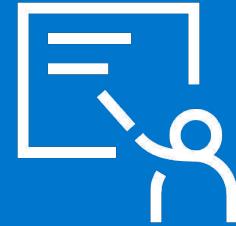
Lab 6.1



定義済み Machine Learning ジョブを探索してみましょう

カスタム ML ジョブ

Module 6 Lesson 2



ジョブ作成ウィザード

- 定義済みの ML ジョブに適切なものが見つからない場合は、ウィザードを使ってカスタム ML ジョブを作成することができる

Create a job from the data view metrics-*

Use preconfigured jobs

The fields in your data match known configurations. Create a set of preconfigured jobs.



APM

Detect anomalies in transaction latency, throughput and failure rate from your APM services for metric data.



Metricbeat System

Detect anomalies in Metricbeat System data (ECS)



Security: Linux

Detect suspicious activity using ECS Linux events. Tested with Auditbeat and the Elastic agent.

Use a wizard



Single metric

Detect anomalies in a single time series.



Multi-metric

Detect anomalies with one or more metrics and optionally split the analysis.



Population

Detect activity that is unusual compared to the behavior of the population.



Advanced

Use the full range of options to create a job for more advanced use cases.



Categorization

Group log messages into categories and detect anomalies within them.



Rare

Detect rare values in time series data.

利用可能なウィザード

- **シングルメトリック (single metric)** ジョブは単一の Detector を使用してフィールドと分析の種類を定義する
- **マルチメトリック (multi-metric)** ジョブはひとつのソースデータに対して1つ以上の Detector を使用して分析を効率化する
- **ポピュレーション (population)** ジョブは母集団と異なる振る舞いを検出するのに用いる
- **カテゴライゼーション (categorization)** ジョブはログメッセージをカテゴリーに分類し、カウントなどの関数を使用して異常を検出する
- **アドバンスド (advanced)** では複数の Detector を用いて、すべてを手動で設定する

シングルメトリックジョブ

- はじめに、利用するデータの期間を設定する

Create job: Single metric

Using data view metrics-*

1
Time range

2
Pick fields

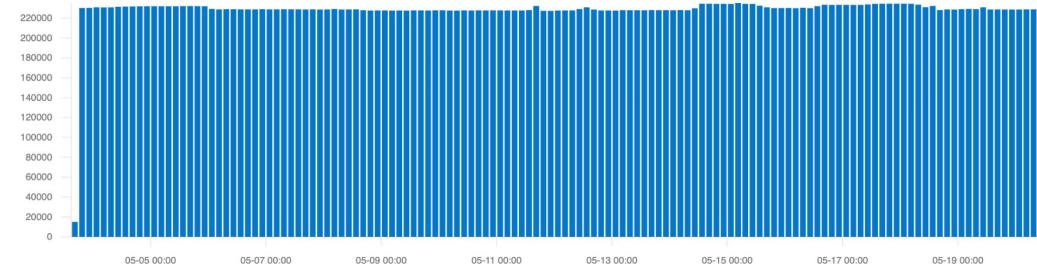
3
Job details

4
Validation

5
Summary

Time range

May 3, 2022 @ 17:47:23.757 → May 20, 2022 @ 14:53:59.000 [Use full data](#) [More](#)



[Next >](#)

フィールドの選択

- 分析対象のフィールドと実施する分析の種類を設定する

Create job: Single metric

Using data view metrics-*

1 Time range 2 Pick fields 3 Job details 4 Validation 5 Summary

Pick fields

Mean(docker.cpu.user.pct)

Bucket span

Set the interval for time series analysis, typically between 15m to 1h.

Bucket span: 15m

Estimate bucket span

Sparse data

Select if you wish to ignore empty buckets from being considered anomalous. Available for count and sum analysis.

Sparse data

Convert to multi-metric job

分析の間隔を指定するこ
ともできる

< Previous Next >

ジョブの詳細の設定

- ジョブの名前、説明、ジョブのグループを定義する

Create job: Single metric

Using data view metrics-*



Job details

Job ID

A unique identifier for the job. Spaces and the characters / ?, , " > | * are not allowed

Job ID

docker-cpu-mean

Job description

Optional descriptive text

Job description

Groups

Optional grouping for jobs. New groups can be created or picked from the list of existing groups.

Groups

my_ml_jobs x ▼

› Additional settings

› Advanced

◀ Previous

Next ▶

検証

- ジョブの設定がモデルに沿って正しく行われていることをチェックする

Create job: Single metric

Using data view metrics-*



Validation

✓ Time range

Valid and long enough to model patterns in the data.

✓ Model memory limit

Valid and within the estimated model memory limit. [Learn more](#)

◀ Previous

Next ▶

サマリ

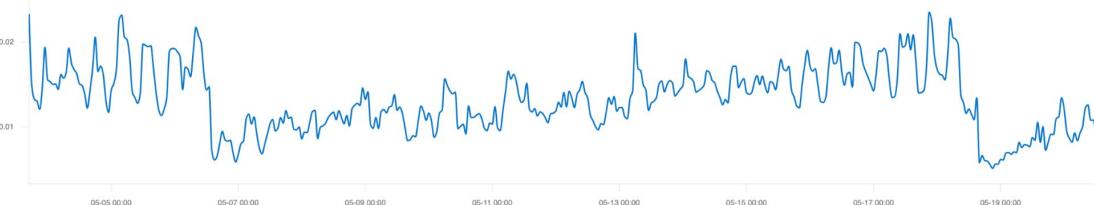
- 設定した ML ジョブを作成する前に設定を最終確認する

Create job: Single metric

Using data view metrics-*

Time range Pick fields Job details Validation Summary

New job from data view metrics-*



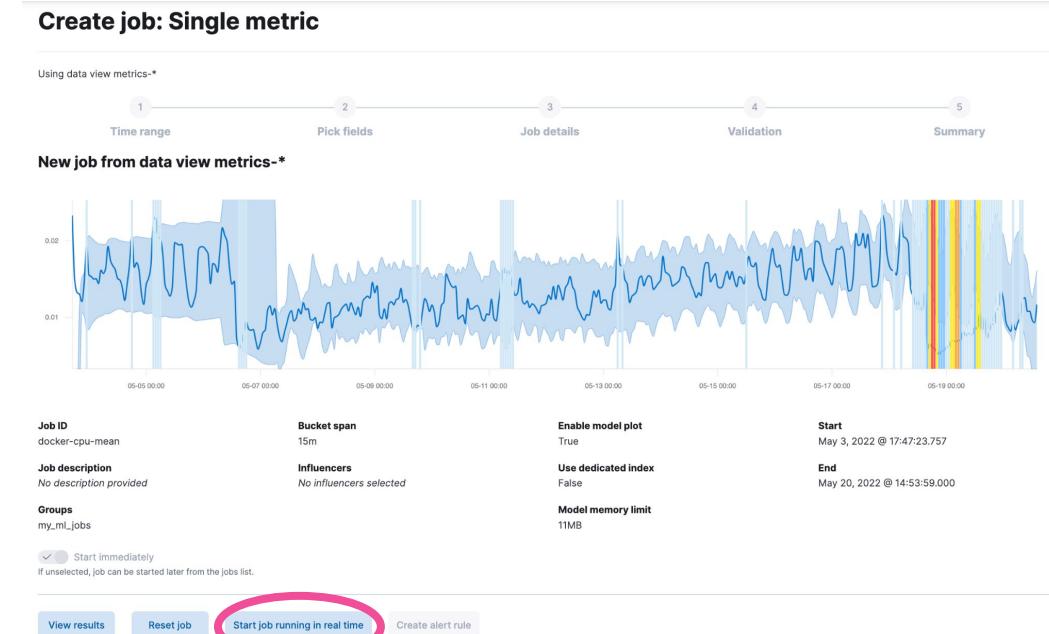
Job ID: docker-cpu-mean
Bucket span: 15m
Enable model plot: True
Start: May 3, 2022 @ 17:47:23.757
Job description: No description provided
Influencers: No influencers selected
Use dedicated index: False
End: May 20, 2022 @ 14:53:59.000
Groups: my_ml_jobs
Model memory limit: 11MB

Start immediately
If unselected, job can be started later from the jobs list.

< Previous Create job Preview JSON Convert to advanced job

リアルタイムでジョブを実行する

- 特定の期間のデータでジョブを作成した後、リアルタイムにデータを処理するように設定することもできる



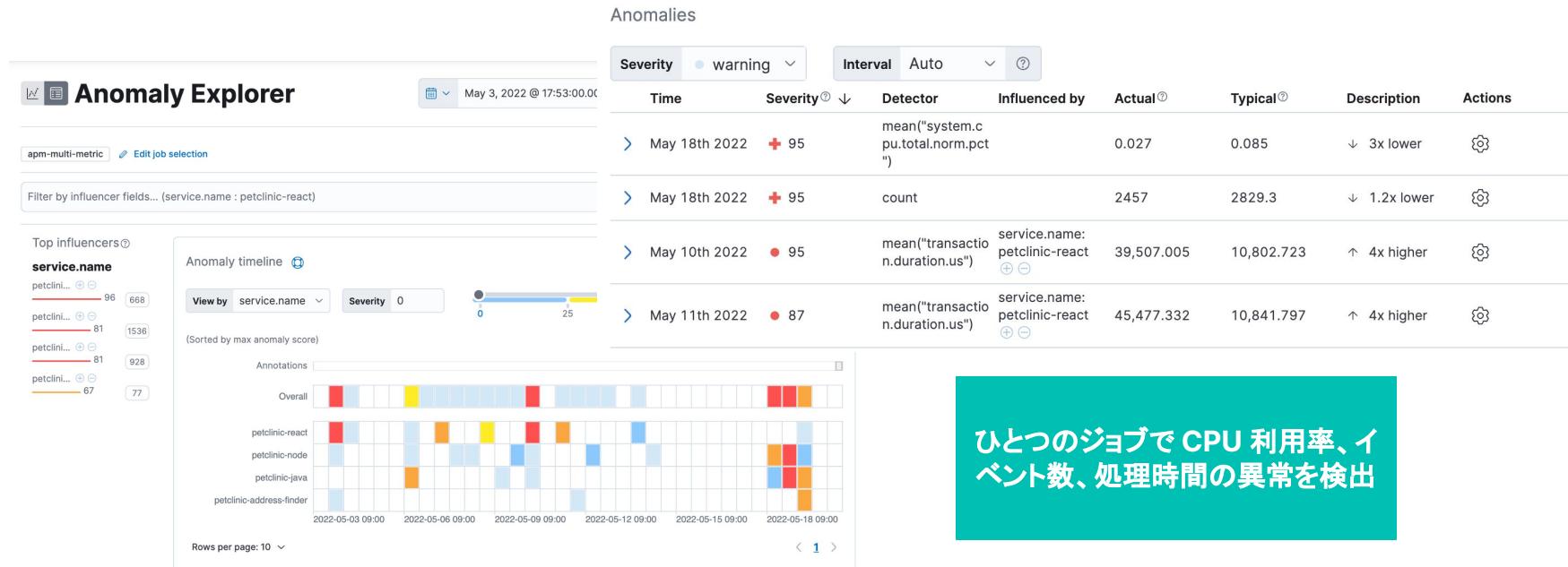
結果の確認

- 異常値は青から赤のスペクトラムで表現される



マルチメトリックジョブ

- 1つ以上のフィールドを使用して例外を発見できる



Summary: カスタム ML ジョブ

Module 6 Lesson 2



Summary

- Elastic ML は、シングルメトリック (single metric)、マルチメトリック (multi-metric)、ポピュレーション (population)、カテゴライゼーション (categorization)、アドバンスド (advanced) のカスタムジョブを定義するためのウィザードを提供する
- シングルメトリックジョブは単一の Detector を使用してフィールドと分析の種類を定義する
- マルチメトリックジョブはひとつのソースデータに対して1つ以上の Detector を使用して分析を効率化する

Quiz

1. **True or False:** マルチメトリックジョブは、一見すると相関関係のなさそうな複数のフィールドから異常を検出することができる
2. **True or False:** シングルメトリックジョブの Detector は、ひとつの KPI に対する分析の種類を定義する
3. **True or False:** カテゴライゼーションジョブは母集団と異なる振る舞いをするデータを検出する

カスタム ML ジョブ

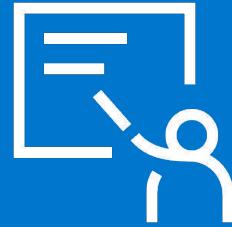
Lab 6.2

カスタム ML ジョブを探索してみましょう



アラート

Module 6 Lesson 3



アラート

- アラート機能では、複数の条件を元にデータを検知するルールを定義し、条件に合致した際にアクションをトリガーすることができる

The screenshot shows the Elasticsearch Management interface under the 'Rules' tab. The left sidebar includes sections for Ingest, Data, and Alerts and Insights, with 'Rules and Connectors' selected. The main area displays the 'Rules and Connectors' page with the title 'Rules and Connectors'. It features a search bar and filters for Type (0), Action type (0), and Last response (0). Below this, a table lists two rules:

Name	Last run	Interval	Duration	P50	Success ratio	Last response	State
apm-alert Anomaly	May 20, 2022 09:52:15am a few seconds ago	1 min	00:00	00:00	100%	Ok	Enabled
uptime-alert Uptime monitor status	May 20, 2022 09:52:15am a few seconds ago	1 min	00:00	00:00	100%	Ok	Enabled

At the bottom, there are pagination controls for 'Rows per page: 10' and navigation arrows.

Observability へのアラート機能の統合

- アラートは **Rules and Connectors** の画面、または Observability の各アプリから管理することができる

The screenshot shows the Elastic Observability interface with the 'Alerts' tab selected. The top navigation bar includes 'Observability' and 'Alerts'. On the left, a sidebar lists 'Overview', 'Alerts' (which is active), 'Cases', 'Logs', 'Stream', 'Anomalies', 'Categories', and 'Metrics'. The main area is titled 'Alerts' with a 'TECHNICAL PREVIEW' badge. It displays a summary of alert counts: Rule count 2, Disabled 0, Snoozed 0, Errors 0, and a 'Manage Rules' button. Below this is a search bar with placeholder 'Search alerts (e.g. kibana.alert.evaluation.threshold > 75)', a KQL button, a date range selector set to 'Last 7 days', and a 'Refresh' button. A table below shows alert details with columns: Actions, Alert Status, Last updated, Duration, and Reason. One alert is listed: 'Recovered' status, last updated on May 18, 2022, duration 228 min, reason 'Pet Clinic Servers from Unnamed-location 30 days availability is 99.00%. Alert when < 99%.'.

概念と言葉の定義

- アラートはルールに定義された条件とスケジュールに基づいて定期的にチェックを実行し、異常を検知する
- 条件に合致した場合、ルールは一つ以上のアクションを実行する
- アクションは Connector を利用して、実際にアラートを送信する Kibana のサービスやサードパーティのインテグレーションと通信を行う

Rule

Condition	<i>server avg CPU > 0.9 for last 2 minutes</i>						
Schedule	<i>every minute</i>						
Actions	<table border="1"><tr><td>Type:</td><td><i>email</i></td></tr><tr><td>Connector:</td><td><i>host = my.co port = 587</i></td></tr><tr><td>Properties</td><td><i>subject = "high CPU" body = "CPU on {{server}} is high"</i></td></tr></table>	Type:	<i>email</i>	Connector:	<i>host = my.co port = 587</i>	Properties	<i>subject = "high CPU" body = "CPU on {{server}} is high"</i>
Type:	<i>email</i>						
Connector:	<i>host = my.co port = 587</i>						
Properties	<i>subject = "high CPU" body = "CPU on {{server}} is high"</i>						

アプリの表示内容からアラートを作成する

- 関連するデータの作業中にアラートを直接作成することができる

The screenshot shows the Elasticsearch Metrics Explorer interface. On the left, there's a sidebar with navigation links like Observability, Metrics, and Metrics Explorer. The main area displays four time-series charts for metrics: docker.cpu.total.pct, address-finder, client-loadgen, fleet-server, and kibana. A pink circle highlights the "Alerts and rules" tab in the top right corner of the Metrics Explorer header. A modal window titled "Create rule" is open, divided into three sections: "ルール" (Rule), "スケジュール" (Schedule), and "条件" (Conditions). The Rule section contains fields for Name ("High average CPU") and Tags (optional). The Schedule section shows a dropdown for "Check every" set to 15 minutes and a "Notify" dropdown set to "Only on status change". The Conditions section shows a condition: "WHEN Average OF docker.cpu.total.pct IS ABOVE 20 %". Below this, a histogram chart titled "Last 100 minutes of data for elasticsearch" shows CPU usage over time. At the bottom of the modal are "Cancel" and "Save" buttons.

ルールとタイプ

- Kibana のアプリとその作業中の内容に応じて、作成するアラートのルールとタイプを選択することができる

Create rule

APM 4

Anomaly
Alert when either the latency, throughput, or failed transaction rate of a service is anomalous.

Error count threshold
Alert when the number of errors in a service exceeds a defined threshold.

Failed transaction rate threshold
Alert when the rate of transaction errors in a service exceeds a defined threshold.

Latency threshold
Alert when the latency of a specific transaction type in a service exceeds a defined threshold.

INFRASTRUCTURE 2

Inventory
Alert when the inventory exceeds a defined threshold.

Metric threshold
Alert when the metrics aggregation exceeds the threshold.

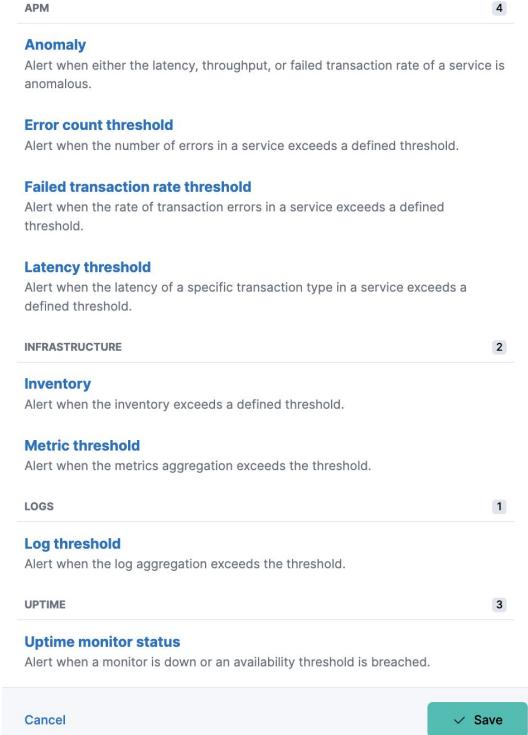
LOGS 1

Log threshold
Alert when the log aggregation exceeds the threshold.

UPTIME 3

Uptime monitor status
Alert when a monitor is down or an availability threshold is breached.

[Cancel](#) [Save](#)



好みの方法で通知を受け取る

- ビルトインのインテグレーションを利用するアクションと接続したアラートを拡張することができる

Actions

Select a connector type



Email



IBM Resilient



Index



Jira



Microsoft
Teams



PagerDuty



Server log



ServiceNow
ITOM



ServiceNow
ITSM



ServiceNow
SecOps



Slack



Swimlane



Webhook



xMatters

Connectors

- 各アクションは特定の Connector のインスタンスを指定する必要がある

Actions

⌄ ⏚ Elastic-Cloud-SMTP (preconfigured) ⌄

Run when Alert ⌄

Email connector Add connector

Elastic-Cloud-SMTP ⌄

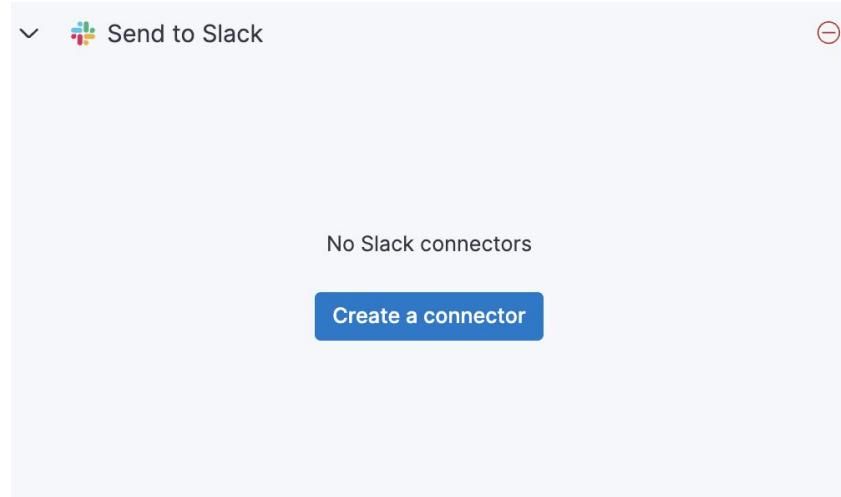
アクションのタイプごとに異なる設定

- 例えば Email アクションには、メールの宛先、件名、マークダウン形式で記載する本文などを設定することができる

The screenshot shows the configuration of an Email action named "Elastic-Cloud-SMTP (preconfigured)". The "Run when" dropdown is set to "Alert". The "Email connector" dropdown is set to "Elastic-Cloud-SMTP". The "To" field contains "kibana-alerts@myco.com". The "Subject" field contains "Alert {{rule.name}} - site: {{context.group}} value: {{context.value}}". The "Message" field contains "{{alertName}} - {{context.group}} is in a state of {{context.alertState}}
Reason:
{{context.reason}}". At the bottom, there is a blue "Add action" button.

複数のアクションの定義

- ひとつのルールにアクションを複数定義することができる
- 選択したアクションのタイプに紐付けられた Connector が存在しない場合は Connector を追加する必要がある



すべてのアラートを **Kibana** 上で集中管理する

- Kibana Management タブの **Alerts and Insights** のセクションで、作成したすべてのルールと Connector の参照、検索、フィルター、管理を行うことができる

Alerts and Insights ?

[Rules and Connectors](#)

Cases

Reporting

Machine Learning

Watcher

Rules and Connectors

Documentation

Detect conditions using rules, and take actions using connectors.

[Create rule](#) Type 0 Action type 0 Last response 0 Refresh

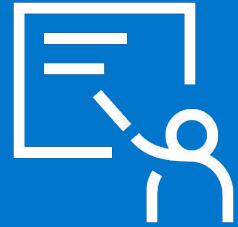
Showing: 2 of 2 rules. ● Active: 0 ● Error: 0 ● Warning: 0 ● Ok: 2 ● Pending: 0 ● Unknown: 0

Name ↑	Last run	Inter...	Duration	P50	Success ratio	Last response	State
apm-alert Anomaly	May 20, 2022 11:27:28am a few seconds ago	1 min	00:00	00:00	100%	● Ok	Enabled
uptime-alert Uptime monitor status	May 20, 2022 11:27:28am a few seconds ago	1 min	00:00	00:00	100%	● Ok	Enabled

Rows per page: 10 < 1 >

Summary: アラート

Module 6 Lesson 3



Summary

- アラートは、複数の条件を元にデータを検知するルールを定義し、条件に合致した際にアクションをトリガーすることができる
- アラートは、Observability のアプリに統合されている
- アラートは、Kibana の Stack Management ページにある **Rules and Connectors** 画面から集中管理することができる
- アラートでは、あらかじめ定義された Connector とルールが提供されている

Quiz

1. アラートの3つの構成要素とは何か？
2. **True or False:** Elastic のアラートを使って、ホスト上のディスクスペースが枯渇した際にメールを送ることができる
3. **True or False:** アラートルールを作成するには、専用の API と JSON が必要である

アラート

Lab 6.3

アラートを作成しましょう



Agenda

- **Module 1: Getting started**
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- Module 8: オブザーバビリティデータの管理

オブザーバビリティデータの 可視化

Module 7

Topics

- ダッシュボード
- カスタムビジュализーション

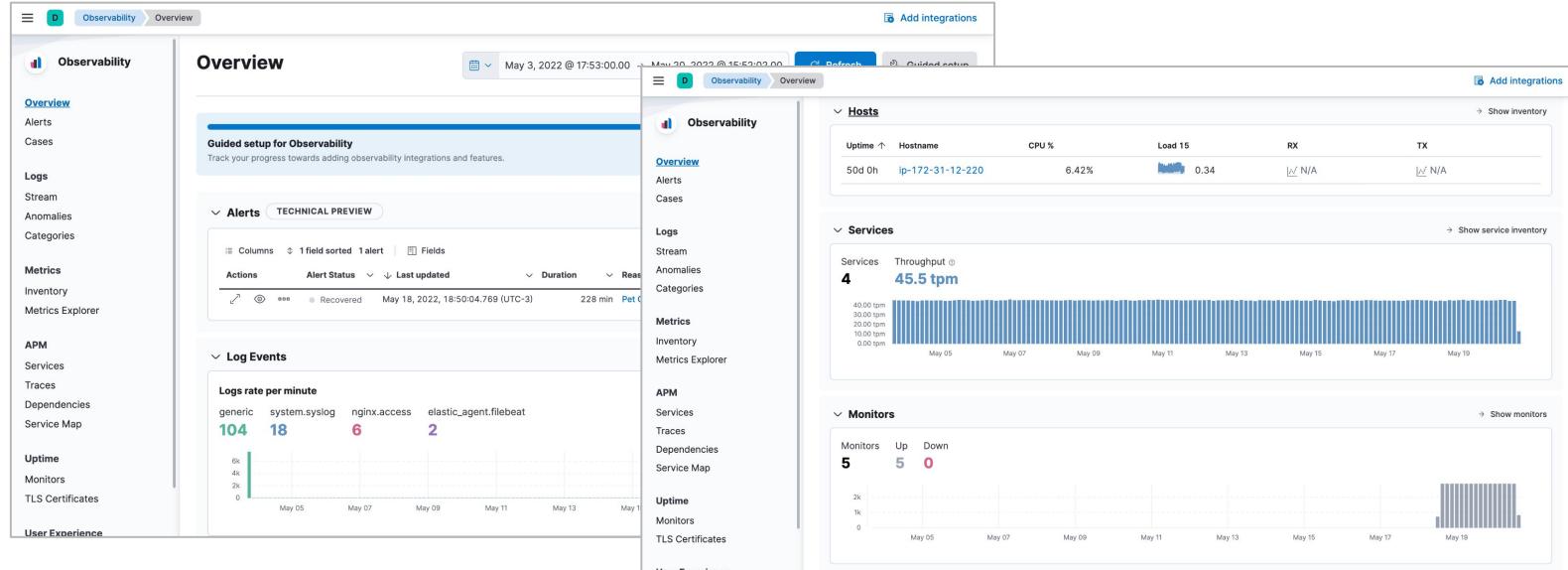
ダッシュボード

Module 7 Lesson 1



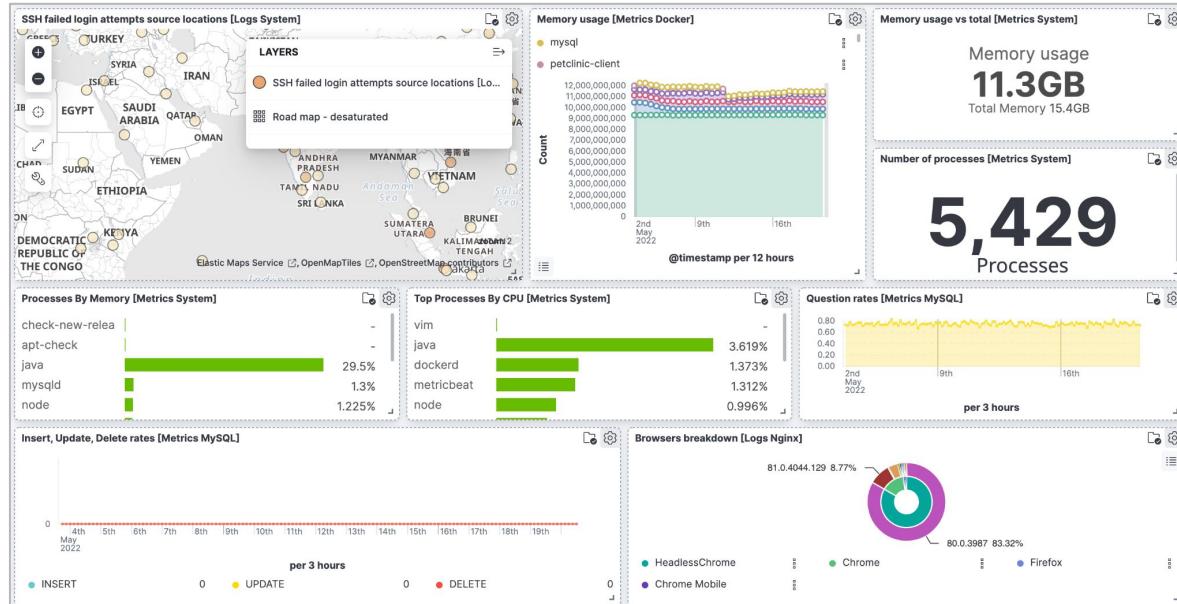
Observability アプリ

- アプリ間の連携、オブザーバビリティデータの概要を提供、しかしあと統合されたビューが欲しい場合もある



統合されたビュー

- ダッシュボードを使えば MySQL メトリック、システムログ、ユーザ体験など異なるソースのデータを組み合わせたビューを作成できる



Integration のアセット

- Integration をインストールするとダッシュボードがいくつか追加される

The screenshot shows the Elasticsearch interface with the 'Integrations' tab selected. Under the 'System' integration, the 'Assets' tab is active, displaying five new dashboards:

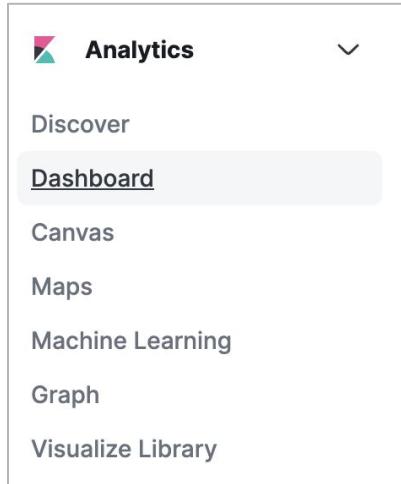
- [System Windows Security] Group Management Events - Simple Metrics
- [System Windows Security] User Logons - Simple Metrics
- [Logs System] New users and groups
- [Logs System] Sudo commands
- [Logs System] SSH login attempts

ダッシュボードとは？

- Kibana のダッシュボードはビジュアリゼーションのコレクション
 - MySQL のように特定のサービスに関連したものだけにすることも
 - アプリケーション内の全てのサービスの集合にすることもできる
- Integration のダッシュボードは通常前者

ダッシュボードへのアクセス

- ダッシュボードは Kibana のメインメニューからアクセスできる
- 特定のダッシュボードを見つけるには検索が便利

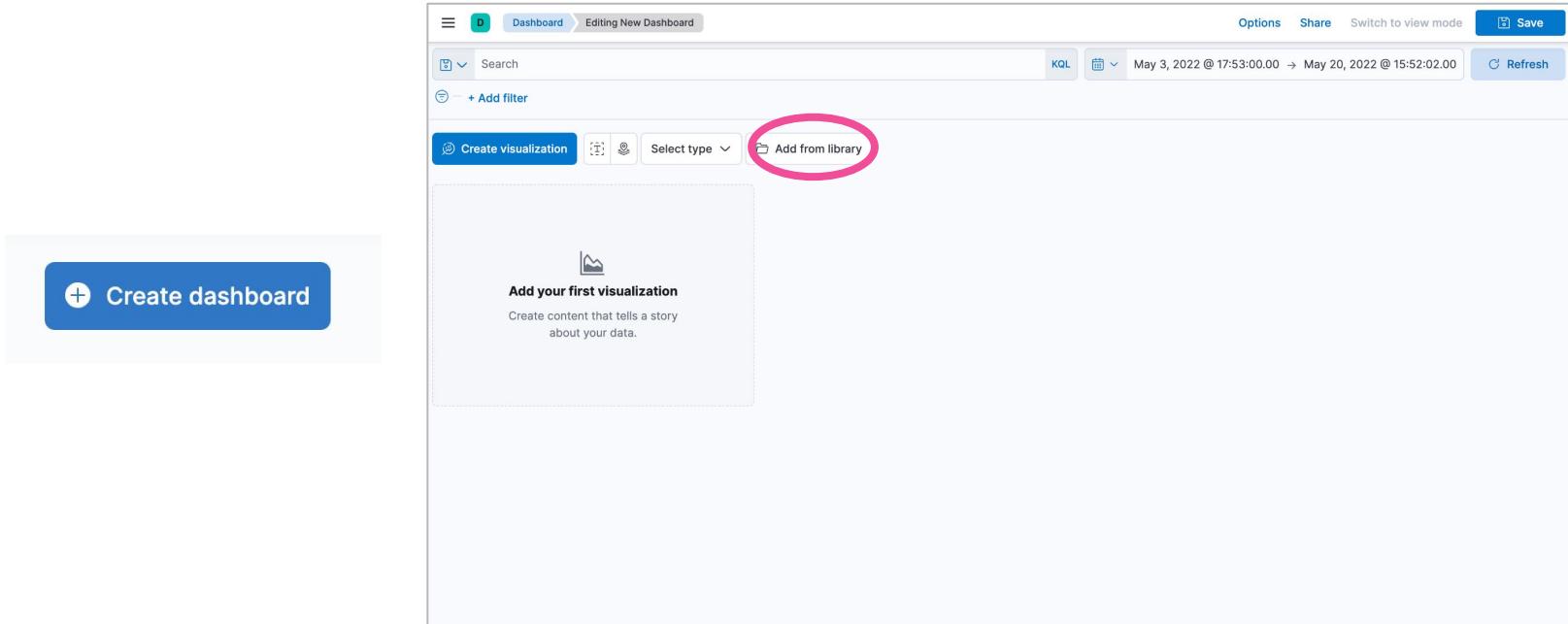


The screenshot shows the 'Dashboards' page in Kibana. A search bar at the top contains the text 'mysql', which is circled in pink. Below the search bar, there is a table with three rows of dashboard entries. The first row has a checkbox next to 'Title' and a link '[Logs MySQL] Overview'. The second row has a checkbox next to 'Title' and a link '[Metrics MySQL] Database Overview'. The third row has a checkbox next to 'Title' and a link 'Overview of MySQL server'. The table includes columns for 'Description', 'Tags', and 'Actions'.

Title	Description	Tags	Actions
<input type="checkbox"/> [Logs MySQL] Overview	Overview dashboard for the Logs MySQL integration		
<input type="checkbox"/> [Metrics MySQL] Database Overview	Overview of MySQL server		
Rows per page: 20 < 1 >			

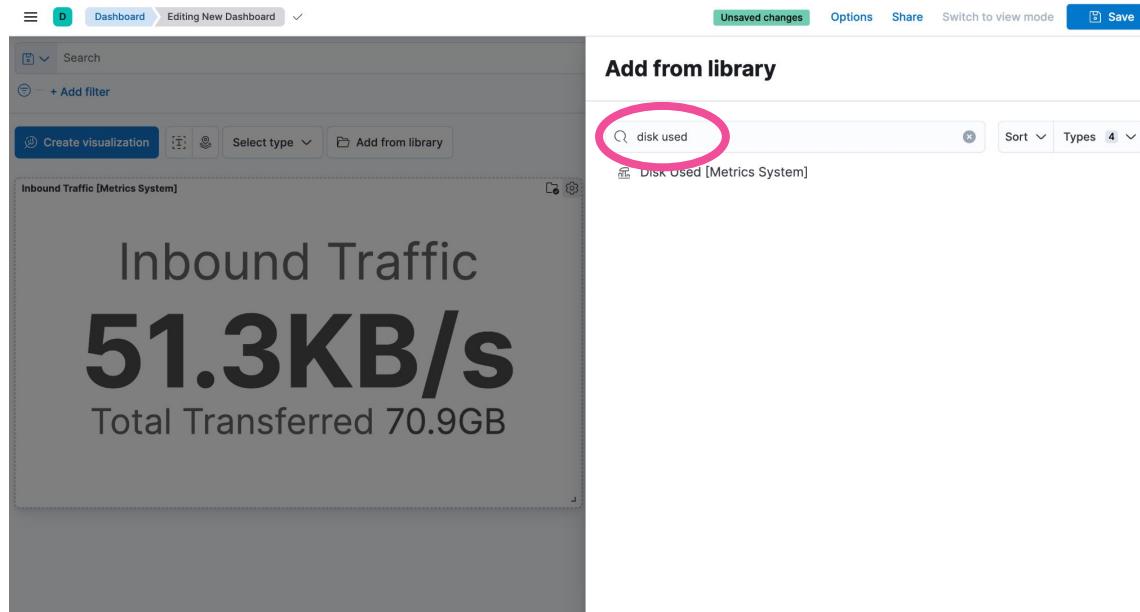
新しいダッシュボードを作成する

- 新しいダッシュボードを作成し、必要なビジュアリゼーションを追加



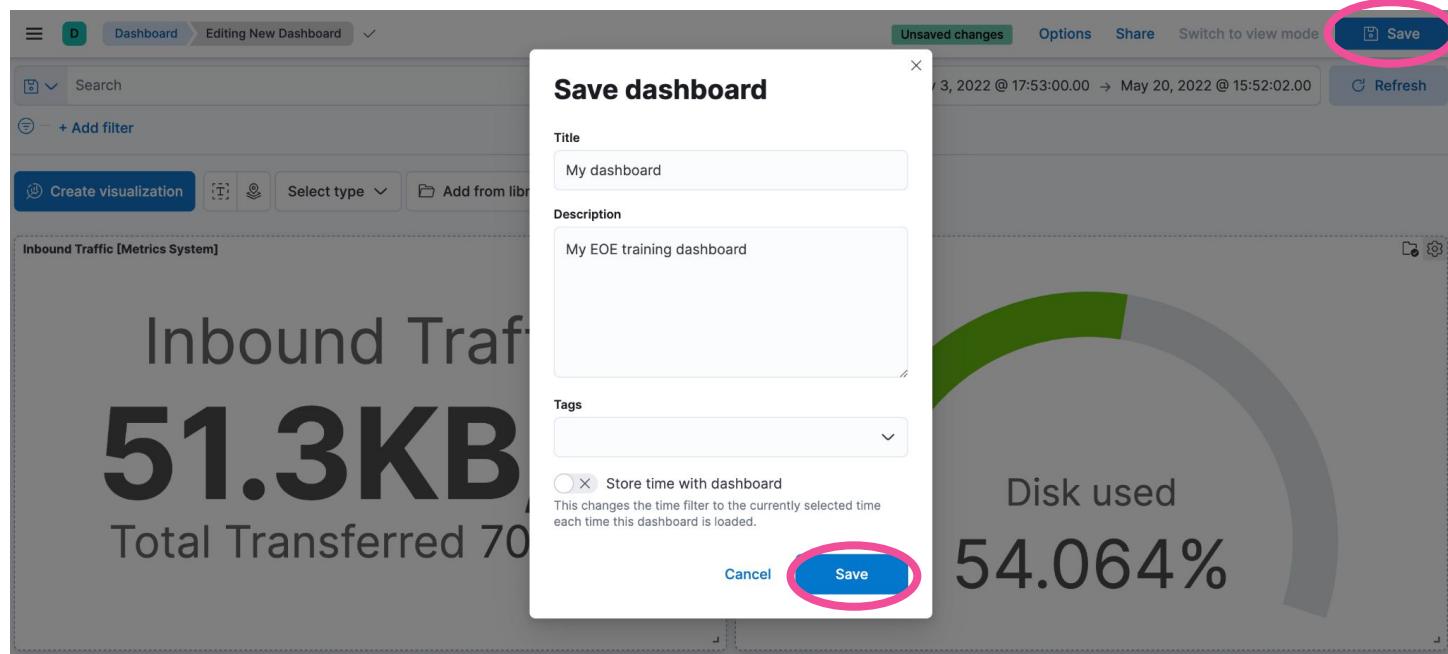
ライブラリからビジュアリゼーションを追加

- Integration によってインストールされた多くのビジュアリゼーションの中から必要なものを検索できる



ダッシュボードを保存する

- ダッシュボードが作成できたら説明を付けて保存する



ダッシュボードの編集

- ビューアリゼーションはダッシュボードにいつでも追加できる

≡ D Dashboard

Dashboards

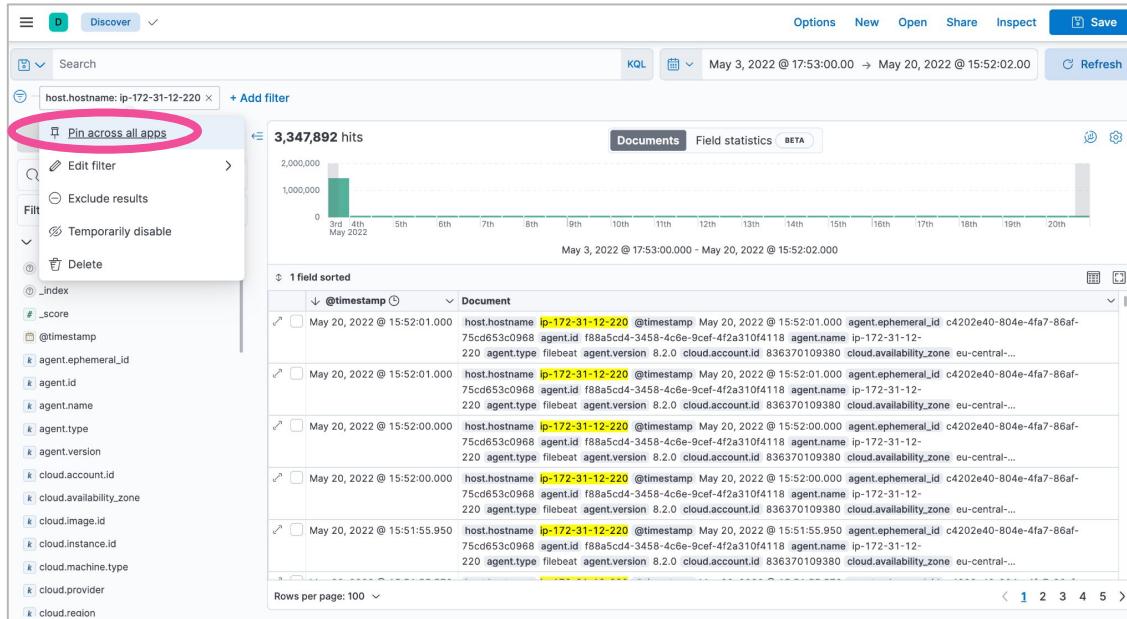
Create dashboard

Search... Tags ▾

Title	Description	Tags	Actions
<input type="checkbox"/> My dashboard	My EOE training dashboard		
<input type="checkbox"/> MyKPIs			
<input type="checkbox"/> [Elastic Agent] Agent metrics	Elastic Agent metrics dashboard		
<input type="checkbox"/> [Logs MySQL] Overview	Overview dashboard for the Logs MySQL integration		
<input type="checkbox"/> [Logs Nginx] Access and error logs	Dashboard for the Logs Nginx integration		
<input type="checkbox"/> [Logs Nginx] Overview	Dashboard for the Logs Nginx integration		
<input type="checkbox"/> [Logs System] New users and groups	New users and groups dashboard for the System integration in Logs		
<input type="checkbox"/> [Logs System] SSH login attempts	SSH dashboard for the System integration in Logs		
<input type="checkbox"/> [Logs System] Sudo commands	Sudo commands dashboard from the Logs System integration		
<input type="checkbox"/> [Logs System] Syslog dashboard	Syslog dashboard from the Logs System integration		

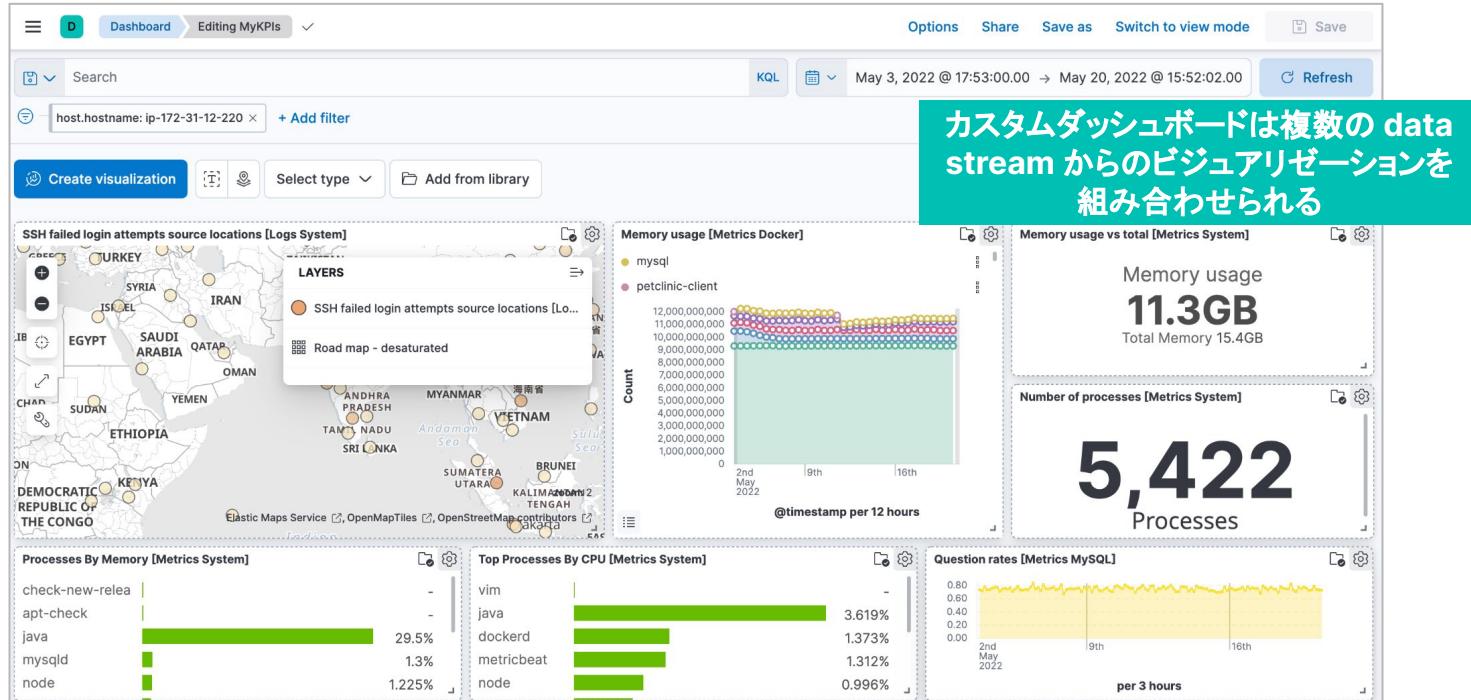
ビジュアリゼーションを固定する

- Discover で作成、ピンしたフィルターはダッシュボードのビジュアリゼーションにも適用される、逆もまた然り



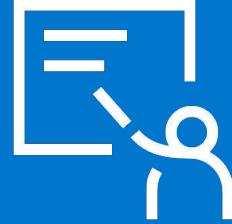
ビジュアリゼーションは協調動作する

- 共通のフィールドに対するフィルターは複数の data view に反映される



Summary: Dashboards

Module 7 Lesson 1



Summary

- Integration はダッシュボードとビジュアリゼーションを追加する
- フィルターは Discover とダッシュボード間を移動できる
- カスタムダッシュボードは複数の data stream からのビジュアリゼーションを組み合わせられる

Quiz

1. **True or False:** Integration を追加するとダッシュボードも追加される
2. **True or False:** Integration のダッシュボードは変更できない
3. **True or False:** Discover で作成したフィルタをダッシュボードに引き継ぐことができる

Dashboards

Lab 7.1

ダッシュボードを使ってみましょう



カスタム ビジュアリゼーション

Module 7 Lesson 2



Observability データの活用

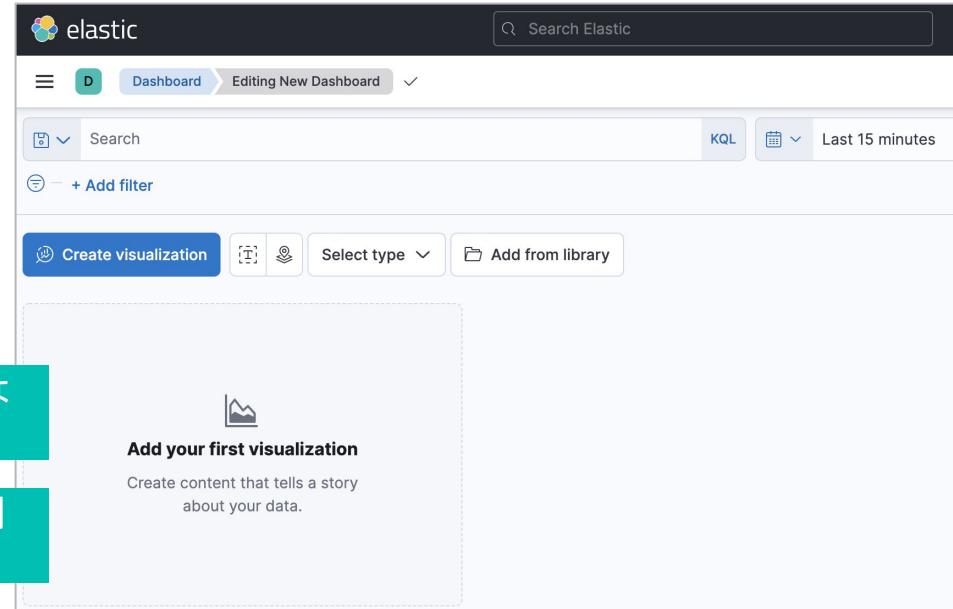
- Observability アプリには多くの素晴らしいビューがある
 - しかし、これらはビジュアリゼーションではなくダッシュボードで直接利用できない
- Integration は多くのビジュアリゼーションをインストールする
 - しかし、違う形式でデータを表示したい場合もある
- そんな時は独自のビジュアリゼーションを作成する

ビジュアリゼーションの作成

- Kibana にはいくつかのビジュアリゼーション作成用のエディタがある:
 - Lens
 - Text
 - Maps
 - など

Create visualization をクリックしてはじめる

Lens ビジュアリゼーションエディタが開く



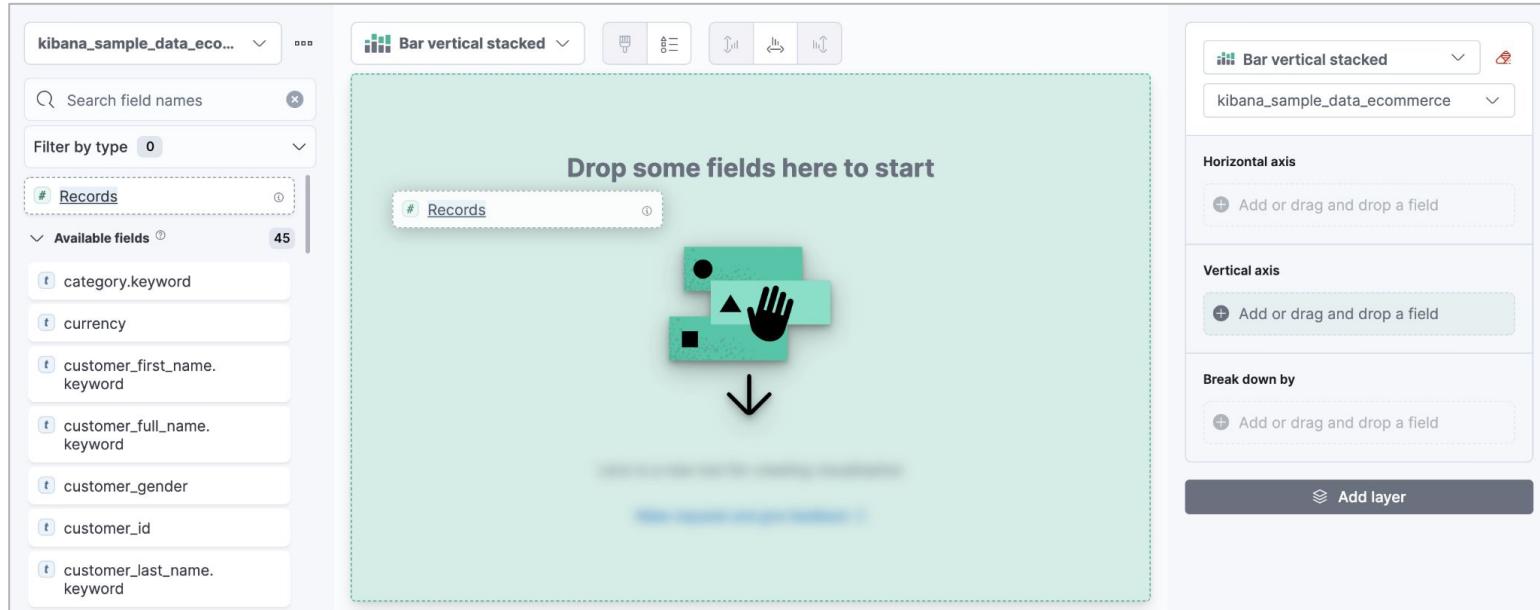
Lens インターフェース

The diagram illustrates the Lens interface with four main panels:

- Data view**: Contains a search bar for field names, a filter by type dropdown, and a list of available fields.
- Fields list**: Shows a visualization type selector (Bar vertical stacked), a search bar, and a list of fields.
- Workspace**: A central area where fields can be dropped to start creating a visualization. It features a hand icon and a placeholder message: "Drop some fields here to start". Below this is a note: "Lens is a new tool for creating visualization" and a link: "Make requests and give feedback".
- Layer pane**: A panel on the right containing sections for Horizontal axis, Vertical axis, and Break down by, each with an "Add or drag and drop a field" button. At the bottom is a "Add layer" button.

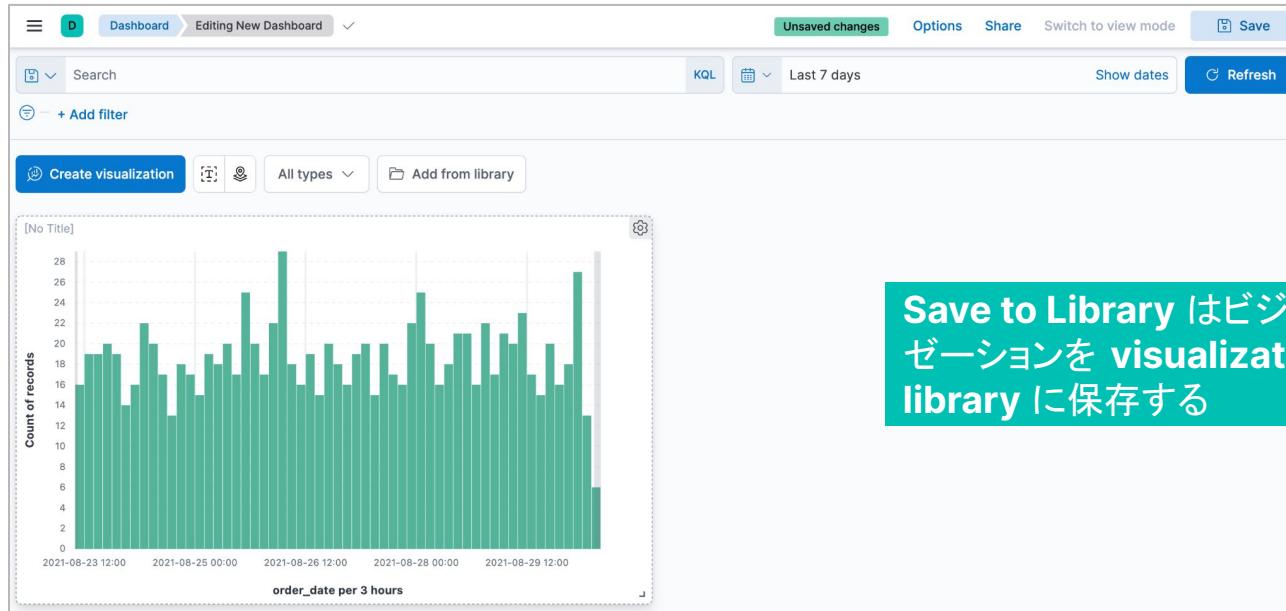
はじめてのビジュアリゼーション作成

- 正しい **data view** と **time filter** 範囲を選択
- フィールドリストからワークスペースにフィールドを drag & drop



ダッシュボードに保存する

- **Save and return** をクリックしダッシュボードに戻る
 - Lens ビジュアリゼーションが新しいパネルとして追加された

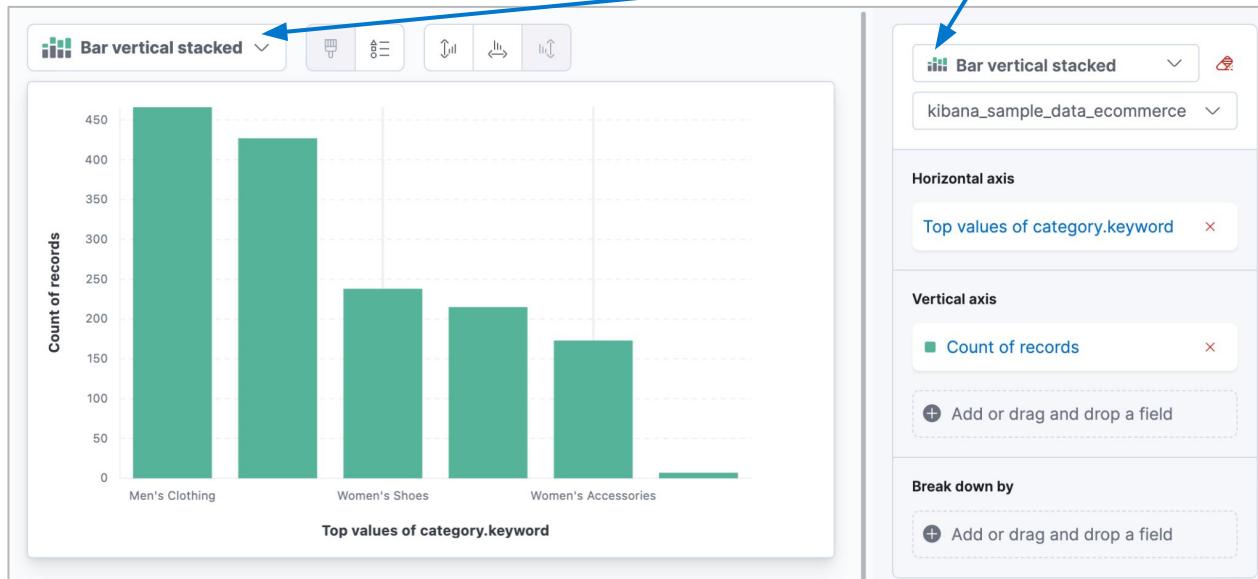


Save to Library はビジュアリゼーションを visualization library に保存する

さらにパネルを追加

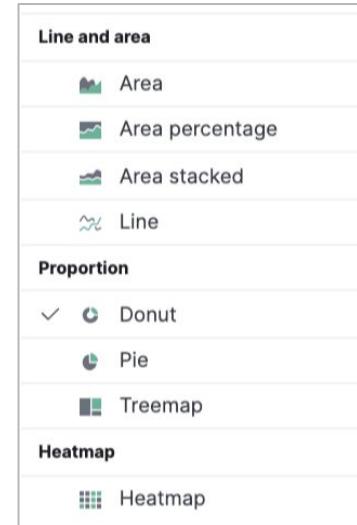
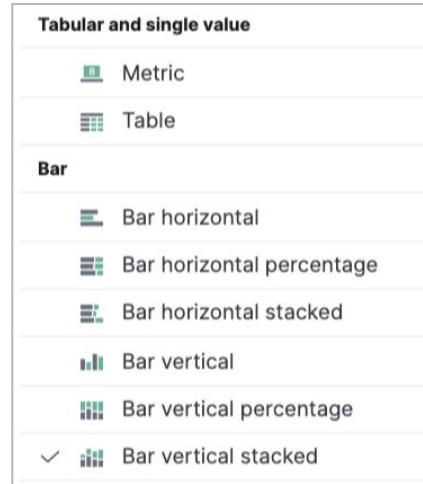
- **Create visualization** をクリックし Lens に戻る
- いろいろなチャートを試してみる

チャートタイプ



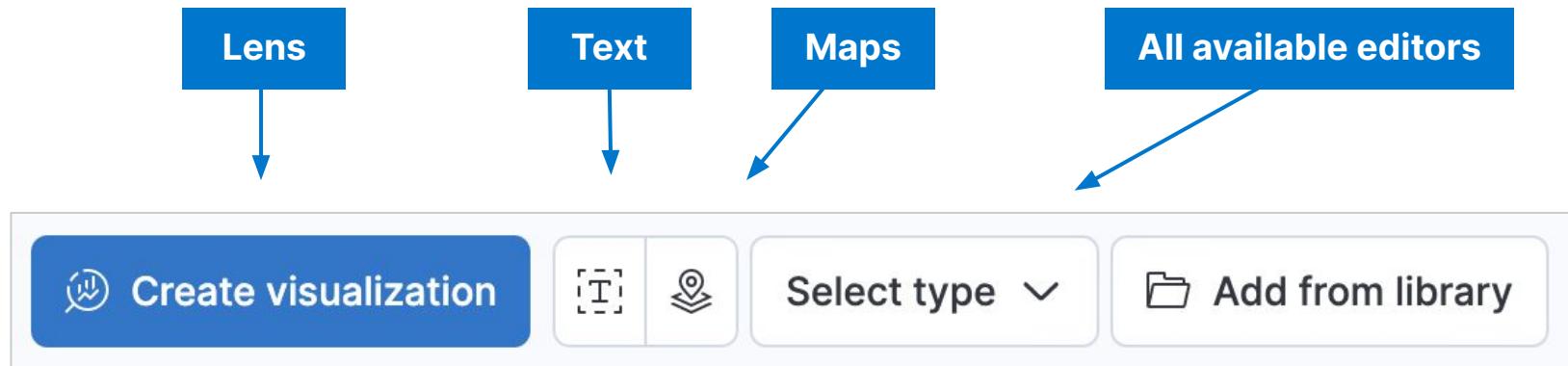
チャートタイプの変更

- いろいろなチャートタイプが選べる



さらにパネルを追加

- 他のエディタを使って別の種類のパネルを作成
 - レガシーなエディタは **Select type → Aggregation based** にある



説明を追加する

- Text エディタを使ってダッシュボードにテキストを追加

Hello, Dashboard!

This is my first dashboard.

It uses the **eCommerce** sample dataset.

[Shopping is fun!](#)

Text エディタは GitHub フレーバーの markdown シンタックスを利用

The screenshot shows the Elasticsearch Text editor interface. The main area displays the following Markdown content:

```
# Hello, Dashboard!
## This is my first dashboard.
It uses the **eCommerce** sample dataset.

[Shopping is fun!](https://www.google.com/search?q=ecommerce)
```

The interface includes tabs for "Data" and "Options", a "Help" button, and a "Discard", "Update", and "Off" button at the bottom. A green callout box on the right side says "詳細は Help をクリック".

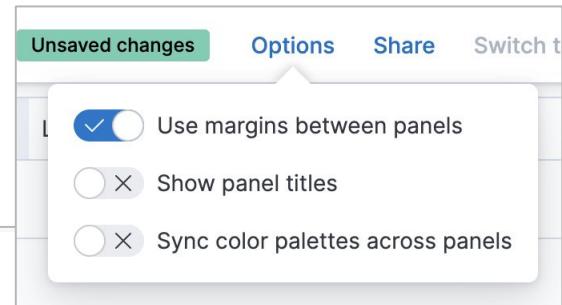
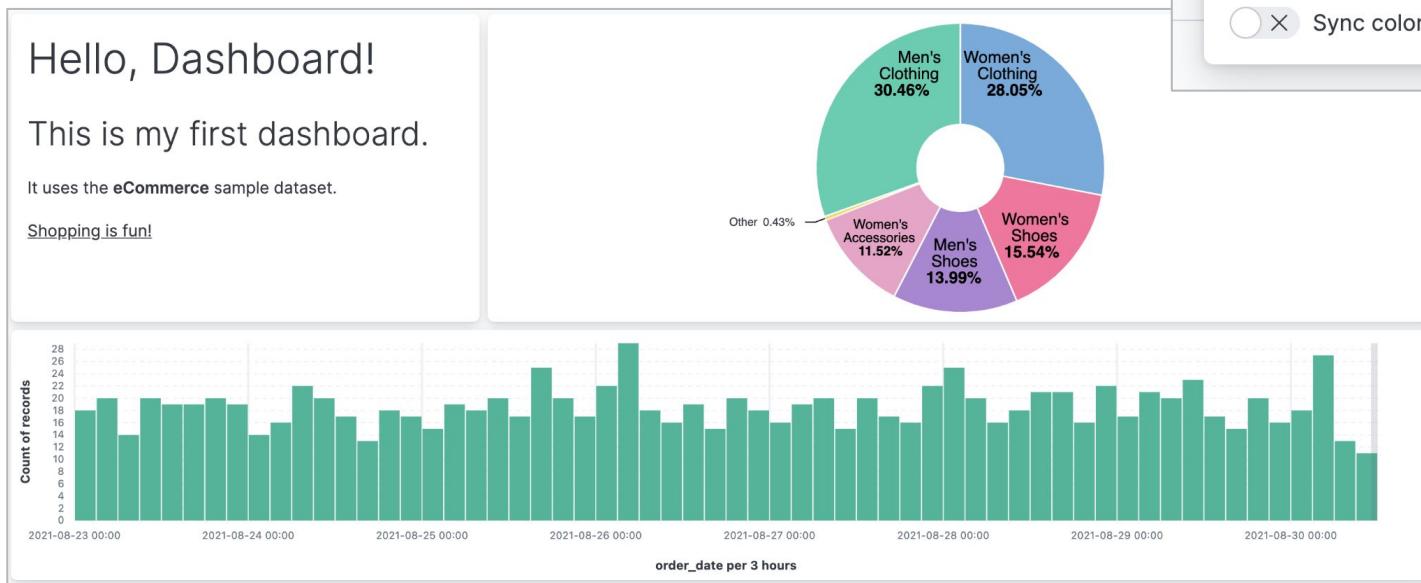
Maps

- 地理的なデータを地図上に可視化する
 - 座標 (longitude and latitude)
 - 国や地域の名前



ダッシュボードをアレンジ

- パネルは移動、リサイズできる
- パネルタイトルを追加、除去できる



ダッシュボードの保存

- ダッシュボードを保存する
 - **Switch to view mode** が利用可能になる
 - **Edit** で編集モードに戻る

Store time with dashboard
を有効にするとダッシュボードのデフォルトの時間範囲として保存できる

Save dashboard

Title
Hello, Dashboard!

Description
My first dashboard

Tags

Store time with dashboard
This changes the time filter to the currently selected time each time this dashboard is loaded.

Cancel **Save**

Summary:

カスタム ビジュアリゼーション

Module 7 Lesson 2



Summary

- 独自のカスタムビジュアリゼーションで見たいデータを可視化できる
- Lens** でビジュアリゼーションを作成する
- Text** エディタでダッシュボードにテキストを追加する
- Maps** では地理情報をもとにデータを可視化できる
- 編集モードでダッシュボードのパネルをアレンジする

Quiz

1. ビジュアリゼーションを作成するためのエディタの名前は?
2. ダッシュボードにテキストを追加するために使う markup 言語の名前は?
3. **True or False:** Map ビジュアリゼーションには地理的なデータが必要

カスタム ビジュアリゼーション

Lab 7.2

ビジュアリゼーションを作成しましょう



Agenda

- Module 1: Getting started
- Module 2: ログとメトリックの収集
- Module 3: APM データの収集
- Module 4: オブザーバビリティデータの活用
- Module 5: データ処理と構造化
- Module 6: オブザーバビリティデータからアクションへ
- Module 7: オブザーバビリティデータの可視化
- **Module 8: オブザーバビリティデータの管理**

オブザーバビリティデータの管理

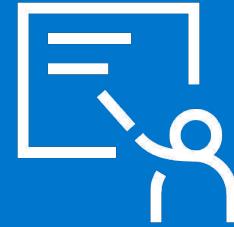
Module 8

Topics

- データストリーム
- インデックスライフサイクル管理
- サーチャブルスナップショット

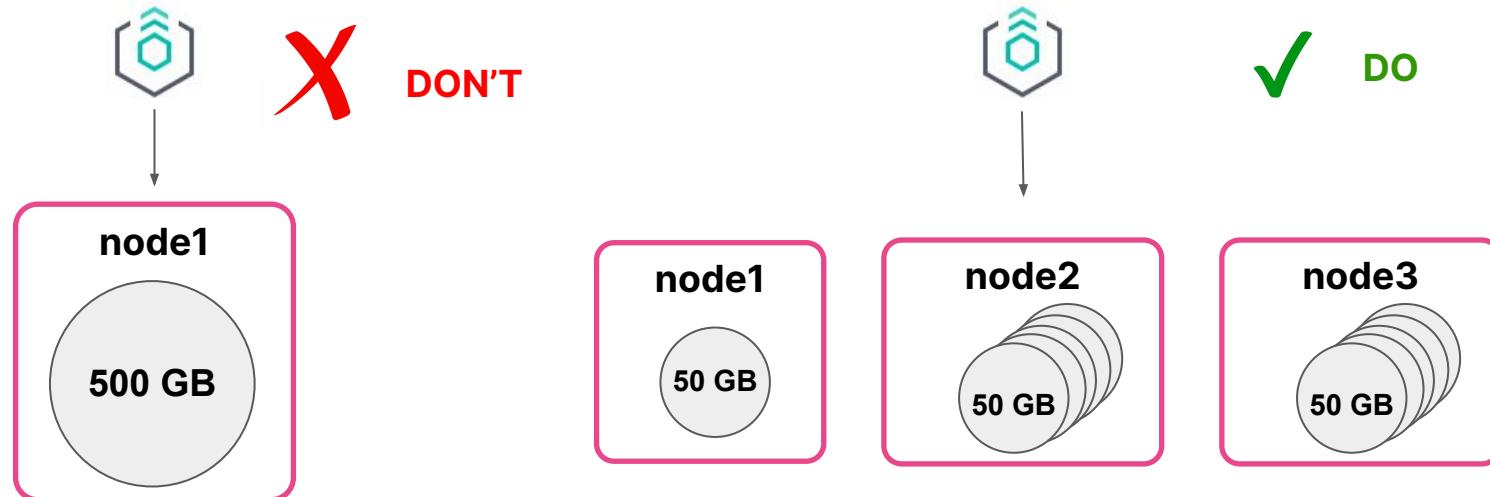
データ管理

Module 8 Lesson 1



インデックスの戦略

- ログやメトリックは時間を追うごとにサイズが大きくなる
 - すべてのデータをひとつのインデックスに保持すると問題を招く
 - ゆえにドキュメントをスマートに分散する必要がある

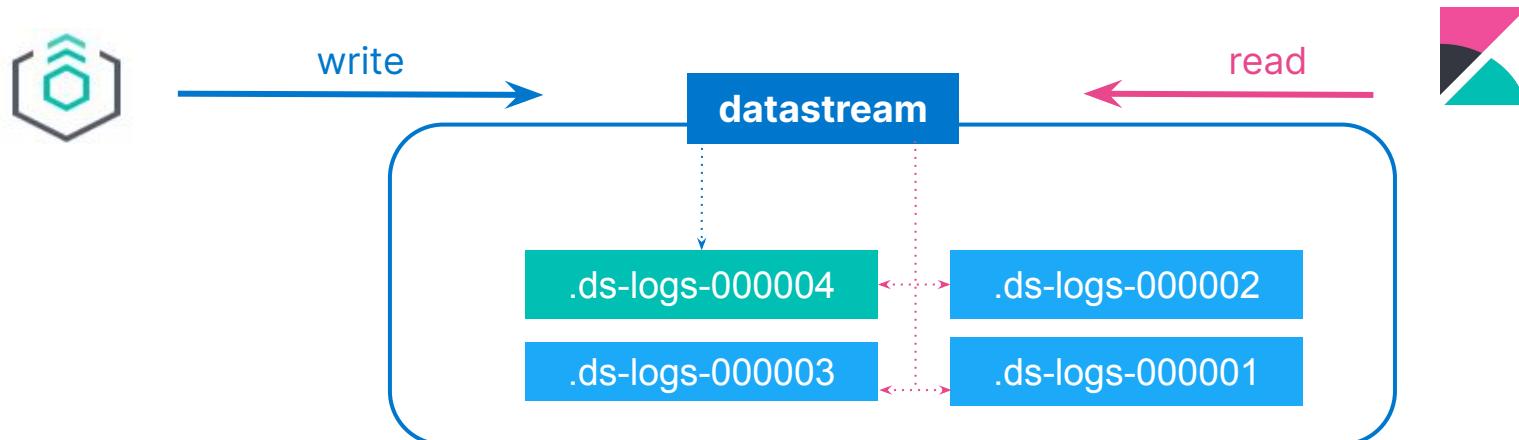


データストリーム

- Elastic エージェントは複数のインデックスにまたがってデータを保存する際にデータストリームを利用する
- Elastic エージェントはリクエストに対して単一のリソース名を提供する
- データストリームはログ、メトリック、トレースなどのデータの保存に適している
- データストリームはいくつかの点で他のインデックス戦略よりも優れている
 - インデックスあたりのフィールド数の増加を抑える
 - より詳細なデータ制御を可能にする
 - 柔軟性を提供する
 - 他の方程式よりも Ingest で要求されるパーミッションが少ない

バックインデックス

- 各データストリームは複数の**非表示のバックインデックス**で構成される
 - 書込み可能なインデックスは1つだけ

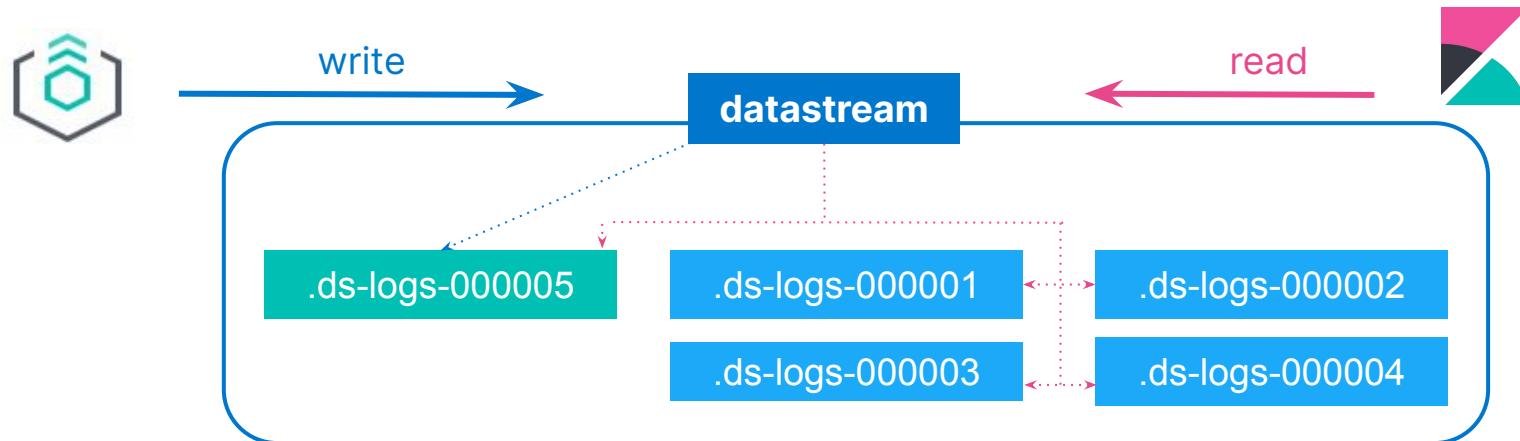


インデックステンプレート

- データストリームのバックティングインデックスは同じ `mappings` と `settings` を使用する必要がある
 - インデックステンプレートはパターンに合致する名前を持つインデックスに適用される
- インデックステンプレートはひとつ以上の `コンポーネントテンプレート` で構成される
- コンポーネントテンプレートは以下の要素を持つ、再利用可能な、インデックステンプレートの構成要素である:
 - `settings`
 - `mappings`
 - `aliases`

ロールオーバー

- ロールオーバーは新しいバックティングインデックスを作成する
 - 経過日数、またはサイズに基づいて
 - ストリームの書込み可能なインデックスとなる



データストリーム名の慣習

- データストリームは以下の規則で命名する:
 - ***type***: データの概要を表現
 - ***dataset***: データの特定のサブセットの表現
 - ***namespace***: ユーザー独自の詳細の表現
- データストリームは次の **constant_keyword** フィールドを持つべき:
 - ***data_stream.type***
 - ***data_stream.dataset***
 - ***data_stream.namespace***
- **constant_keyword** は全てのドキュメントで同じ値を持つ

`metrics-system.cpu-production`

`type`

`dataset`

`namespace`

データストリームの利用例

- app と env で分類されたログデータ
- 各データストリームで個別のライフサイクルを持つ
- データセットごとに異なるフィールドを持たせることができる

logs-app1-prod

logs-app2-prod

logs-app1-dev

```
GET logs-*-*/_search
{
  "query": {
    "bool": {
      "filter": {
        "term": {
          "data_stream.namespace": "prod"
        }
      }
    }
  }
}
```

適切な `constant_keyword` はフィルタ処理を高速化する

Elastic Common Schema

Elasticsearch のデータを構造化する

- データは異なるソースで構成されている
- 一貫性のあるデータはその理解を助ける
- Elastic Common Schema (ECS) は Elasticsearch 内のデータを構造化する一貫した方法を提供する

なぜ common schema なのか？

- データは共通の方法で検証できる必要がある
- しかし、データが一つだけのソースでできていない限り、
- データフォーマットの矛盾に直面する
 - バラバラのデータタイプ
 - 別の種類の環境のデータ
 - 別々のベンダーのデータ

ECS とは何か？

- Open Source の仕様
- Elasticsearch にイベントデータを保存する際の共通のフィールドセットの定義
- Elastic で使用できる分析モデルの統一化
 - 例: サーチ、ビジュアリゼーション、アラート、Machine Learning

```
src:10.42.42.42 OR client_ip:10.42.42.42 OR apache2.access.remote_ip:10.42.42.42 OR  
context.user.ip:10.42.42.42 OR src_ip:10.42.42.42
```



```
source.ip:10.42.42.42
```

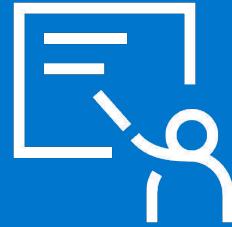
ECS にデータをマッピングする

- インテグレーションはデータを ECS に自動的にマップする
- インテグレーションにはないデータソースを扱うケースも多々ある
- ECS にマッピングするのがお勧め:
 - クエリがシンプルになる
 - 効果的なビジュализーションが簡単に作成できる
 - 通常、特定のパース処理が必要になる

Summary:

データストリーム

Module 8 Lesson 1



Summary

- Elastic エージェントはデータストリームを利用してデータを保存する
- データストリームはエイリアスの裏にあるバックティングインデックスの集合体
- すべてのバックディングインデックスはインデックステンプレートを通じて同じ設定を共有する
- データストリーム は **一度書き込まれたら変更されない** データに対して有効
- Elastic エージェントによって作成された データストリームは ECS 規約に従う

Quiz

1. Elastic エージェントは複数のインデックスに対してデータを送信する
2. Observability データに Elastic Common Schema を使うのはなぜか？
3. バッキングインデックスを作成するプロセスを何と呼ぶか？

データストリーム

Lab 8.1

データストリームを探索しましょう



インデックス ライフサイクル管理

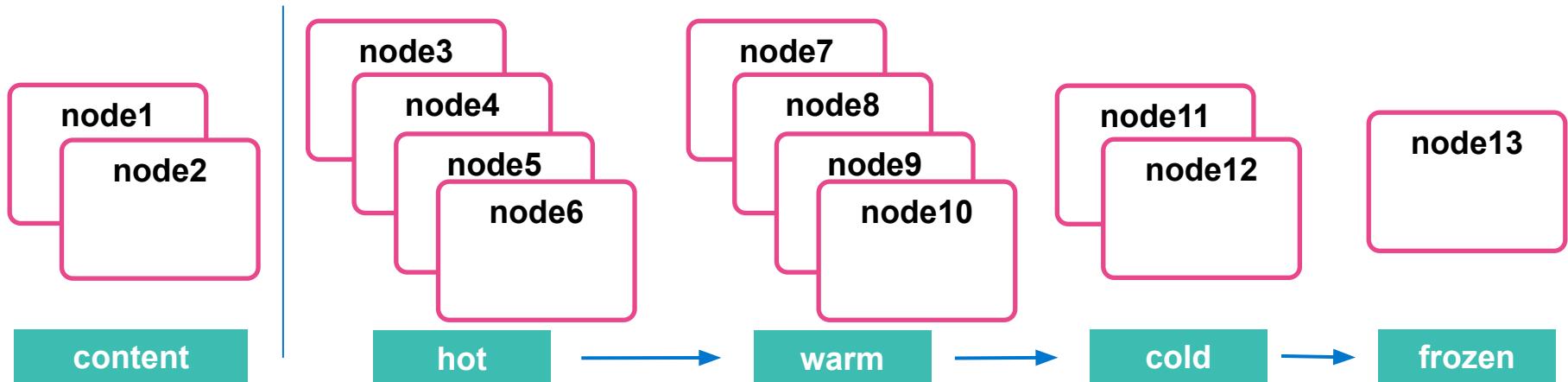
Module 8 Lesson 2



データティア

データティアとは？

- データティアは同じデータロールを持つノードの集合のこと
 - 通常、同じハードウェア構成を持つ
- 5つのデータティアがある:



5つのデータティアの概要

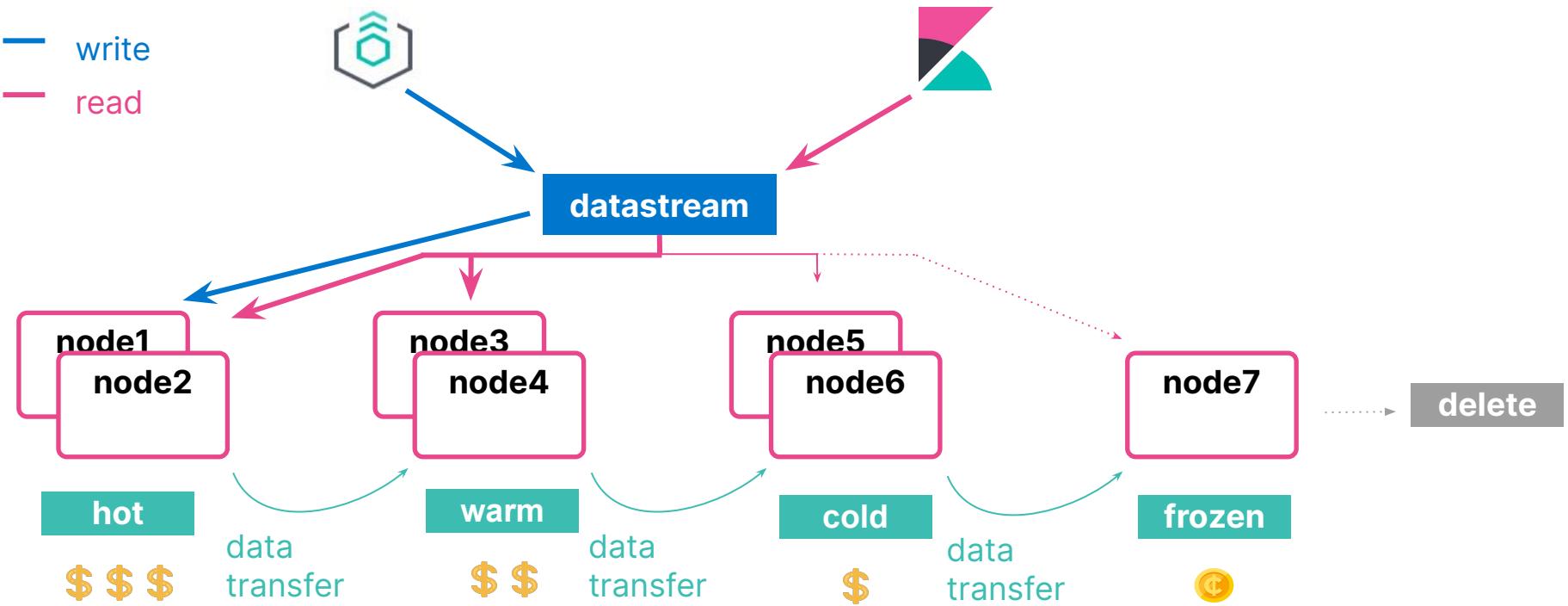
- **Content** ティアは静的なデータセットの保存に便利
- **Hot → warm → cold → frozen** アーキテクチャ は以下のデータティアを利用して実装できる:
 - **Hot** ティア: 頻繁な検索と書き込みのためのデータ、高速ストレージを利用
 - **Warm** ティア: たまに参照する読み取り専用のデータ
 - **Cold** ティア: まれに検索されるデータ
 - **Frozen** ティア: 更新されずめったにアクセスされないデータ

データティア、ノード、インデックス

- 各ノードはデフォルトで *all* のデータティアとして動作
 - `node.roles` パラメータで設定する
 - クラウドのデプロイメントでは自動的にノードロールを設定
- データストリームのインデックスはデフォルトで Hot ティアに作成される
- インデックスが古くなるに従って、より冷たい (colder) ティアに移動する
 - ILM (Index Lifecycle Management)* ポリシーを定義して管理

インデックス ライフサイクル管理

Index lifecycle management (ILM)



ILM のアクション

- ILM は以下のようなアクションを実行するポリシーで構成する

Action	Description
rollover	日数・サイズ・ドキュメント数を元にインデックスを作成
shrink	プライマリーシャードの数を削減
force merge	シャードのセグメントをマージ
サーチャブルスナップショット	めったに使わないインデックスのメモリを削減
delete	物理的にインデックスを削除

エージェントと ILM

- エージェントはロールオーバーに ILM ポリシーを利用する
- デフォルトのエージェントポリシーでは:
 - Hot フェーズに無期限でデータを保持する
 - 削除しない
 - インデックスは 30 日経過後、または 50GB 以上でロールオーバーする
- デフォルトのエージェントポリシーは Kibana で変更できる

ILM ポリシーの例

- ***Hot* フェーズ:**
 - 2週間おきに新しいインデックスを作成
- ***Warm* フェーズ:**
 - インデックスを読み取り専用にし、データを1週間保持
- ***Cold* フェーズ:**
 - 完全にマウントされたインデックス (Fully-mounted index) に変換、レプリカ数を削減し、データを3週間保持
- ***Delete* フェーズ:**
 - ロールオーバーから 28日経過したインデックスを削除
- これを ILM ポリシーとしてどのように設定するか、見てみましょう

Hot フェーズの定義

- インデックスを 2 週間、Hot フェーズに保持する:

```
PUT _ilm/policy/my-hwcd-policy
{
  "policy": {
    "phases": {
      "hot": {
        "actions": {
          "rollover": {
            "max_age": "14d"
          }
        }
      }
    }
  },
}
```

14日経過後、新しいインデックスにロールオーバー

Hot phase Required

Store your most recent, most frequently-searched data in the hot tier. The hot tier provides the best indexing and search performance.

Rollover

Start writing to a new index when the current index reaches a certain size, document count, or age. Enables you to optimize performance and manage resource usage when working with time series data.

Note: How long it takes to reach the rollover criteria in the hot phase can vary. [Learn more](#)

Enable rollover

Maximum primary shard size

Maximum age

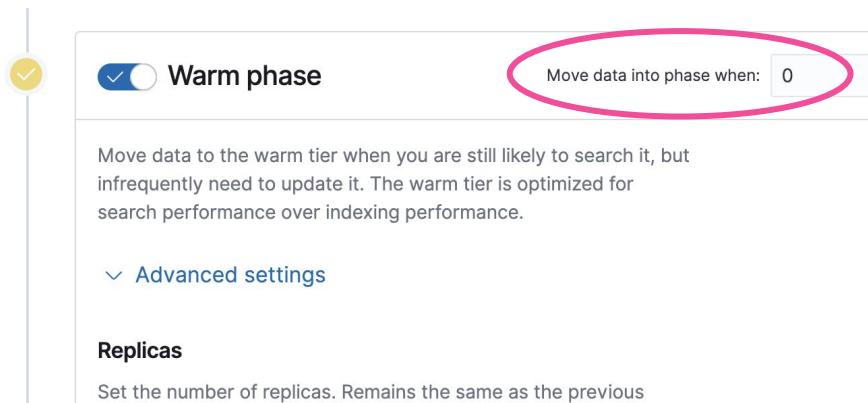
14

Warm フェーズの定義

- 古いインデックスを Warm ティアに即時移動し、インデックスを読み取り専用にする:
 - データの `age` はロールオーバーからの経過日数である点に注意

ロールオーバーの直後に
インデックスを Warm に
移動

```
"warm": {  
    "min_age": "0d",  
    "actions": {  
        "readonly": {}  
    }  
},
```



Cold フェーズの定義

- Warm で1週間経過後、インデックスを Cold フェーズに移動し、レプリカを完全にマウントされたインデックス (fully-mounted index) に変換:

```
"cold": {  
    "min_age": "7d",  
    "actions": {  
        "searchable_snapshot" : {  
            "snapshot_repository" :  
                "my_snapshot"  
        }  
    }  
},
```

ロールオーバーから7日
経過後に Cold に移動

Cold phase Move data into phase when days old

Move data to the cold tier when you are searching it less often and don't need to update it. The cold tier is optimized for cost savings over search performance.

Searchable snapshot Convert to a fully-mounted index that contains a complete copy of your data and is backed by a snapshot. You can reduce the number of replicas and rely on the snapshot for resiliency. [Learn more](#)

Convert to fully-mounted index

Snapshot repository
 Each phase uses the same snapshot repository.

Delete フェーズの定義

- ロールオーバーから4週間後、データを削除する:
 - すなわちドキュメントは 14 日間 Hot に、
 - その後 7 日間 Warm に、
 - 更に 21 日間 Cold に滞留する

```
"delete": {  
    "min_age": "28d",  
    "actions": {  
        "delete": {}  
    }  
}
```

 Delete phase [Remove](#) Move data into phase when: days ▼

Delete data you no longer need.

ロールオーバーから 28 日後に削除

ILM ロールオーバーのトラブルシューティング

- インデックスが green でないと次のフェーズに進まない
- クラスタのデフォルトのポーリング間隔は 10 分
 - `Indices.lifecycle.poll_interval` で設定可能
- サーバーログでエラーが発生していないかをチェック
- データ遷移のためのデータティアが利用できることを確認
- 次のコマンドで ILM のステータスに関する詳細を取得:

```
GET <data-stream>/_ilm/explain
```

Summary:

インデックス

ライフサイクル管理

Module 8 Lesson 2



Summary

- Elasticsearch のデータの保存場所をデータティアで管理
- データストリームは規定で Hot ティアに作られる
- ***Index lifecycle management (ILM)*** は設定を簡単にし、ロールオーバーパターンの自動化を可能にする
- ILM ポリシーは “いつ”、“何をするか” を定義する
 - 各ポリシーは5つのフェーズに分解できる:
Hot・Warm・Cold・Frozen・Delete
- ILM ポリシーは API または Kibana から管理できる

Quiz

1. **True or False:** Hot にあるインデックスがロールオーバーすると、書き込みのリクエストは自動的に新しく作成されたバックティングインデックスに送られる
2. ILM フェーズの5つの名前は？
3. インデックスを Warm フェーズに5日間保存した後、Cold フェーズに移したい場合、Cold フェーズの `min_age` には何をセットするべきか？

インデックス ライフサイクル管理

Lab 8.2

ILM ポリシーをデプロイしてロールオーバーを
管理しましょう



サーチャブルスナップショット

Module 8 Lesson 3



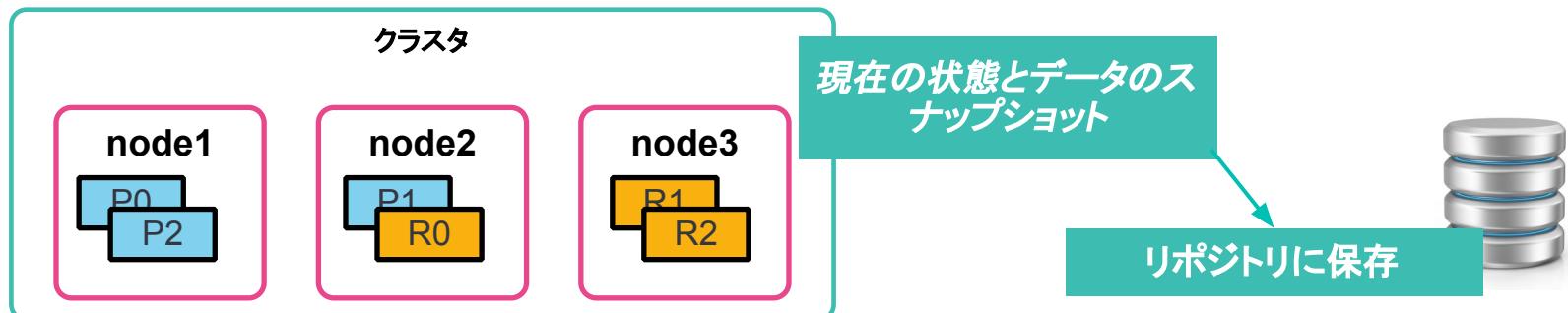
Cold/frozen データを扱う

- データストリームや時系列データが増加するにつれ、ストレージやメモリも拡張が必要
 - それと同時に古くなったデータの利用頻度は下がっていく
- もちろん古くなったデータは削除しても良い
 - ただ、そのデータに価値があるのなら、可能な限り使用できるようにしておきたい
- このようなケースのため、Cold フェーズにはサーチャブルスナップショットというアクションが用意されている
 - その前に、まずはスナップショットについて理解しましょう ...

スナップショット

スナップショットとリストア

- ***Snapshot and restore*** ページでは、稼働中の Elasticsearch クラスタに対するバックアップの作成と管理を可能にする
 - クラスタの現在の状態とデータをリポジトリにバックアップ
- リポジトリはローカルの共有ファイルシステムやクラウドが使える
 - Elasticsearch サービスはスナップショットを自動的に取得



リポジトリの種類

- バックアップ処理はリポジトリの作成からはじまる
 - 下記のリポジトリタイプがサポートされている:

Shared file system	各ノードに設定された path.repo を利用
Read-only URL	複数のクラスタでリポジトリを共有する際に利用
repository-s3 plugin	AWS S3 を利用
repository-azure plugin	Microsoft Azure storage を利用
repository-gcs plugin	Google Cloud Storage を利用
repository-hdfs plugin	Hadoop HDFS を利用

リポジトリのセットアップ

- クラウドのデプロイメントには無料のリポジトリが設定されている
- Kibana を利用してリポジトリを登録できる:

Register repository

Repository name

A unique name for the repository.

Name

my-repo

Repository type

Elasticsearch supports file system and read-only URL repositories. Additional types require plugins. [Learn more about plugins.](#)



Shared file system

[Learn more](#)

✓ Selected



Read-only URL

[Learn more](#)

Select

スナップショットを手動で作成

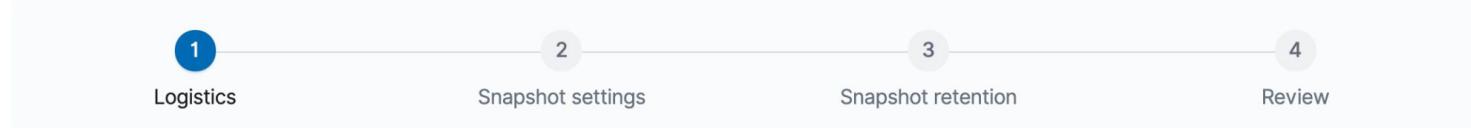
- リポジトリの設定が終わったら、スナップショットを作成できる
 - _snapshot エンドポイントまたは UI を使用する
 - スナップショットはある特定の時点 (“*point-in-time*”) のデータのコピーであり、インクリメンタルに作成される
- 特定のインデックスだけをバックアップすることもできる
- クラスタの状態をバックアップできる

```
PUT _snapshot/my_repo/my_logs_snapshot_1
{
  "indices": "logs-*",
  "ignore_unavailable": true,
  "include_global_state": true
}
```

スナップショットの自動化

- `_snapshot` エンドポイントは手動で実行可能
 - スナップショットはいつでも作成できる
 - 外部のツールを利用して、定期的に実行することも可能
- または、***Snapshot lifecycle management (SLM)*** ポリシーを利用してスナップショットの作成を自動化できる
 - ポリシーは Kibana 上で作成
 - または `_slm` API を利用して作成

Create policy



スナップショットからのリストア

- スナップショット ID の `_restore` エンドポイントで、そのスナップショットから全てのインデックスをリストア:

```
POST _snapshot/my_repo/my_snapshot_2/_restore
```

- Kibana UI からもリストア操作が可能:

Restore 'daily-snap-jwh-r8drtao2clxom8ioq'

The screenshot shows the Kibana Restore interface. At the top, there is a progress bar with three numbered steps: 1 Logistics, 2 Index settings, and 3 Review. Step 1 is highlighted with a blue circle. Below the progress bar, the word "Restore details" is displayed. To the right of "Review", there is a link to "Snapshot and Restore docs". A yellow banner at the bottom left contains the text "⚠ This snapshot contains data streams".

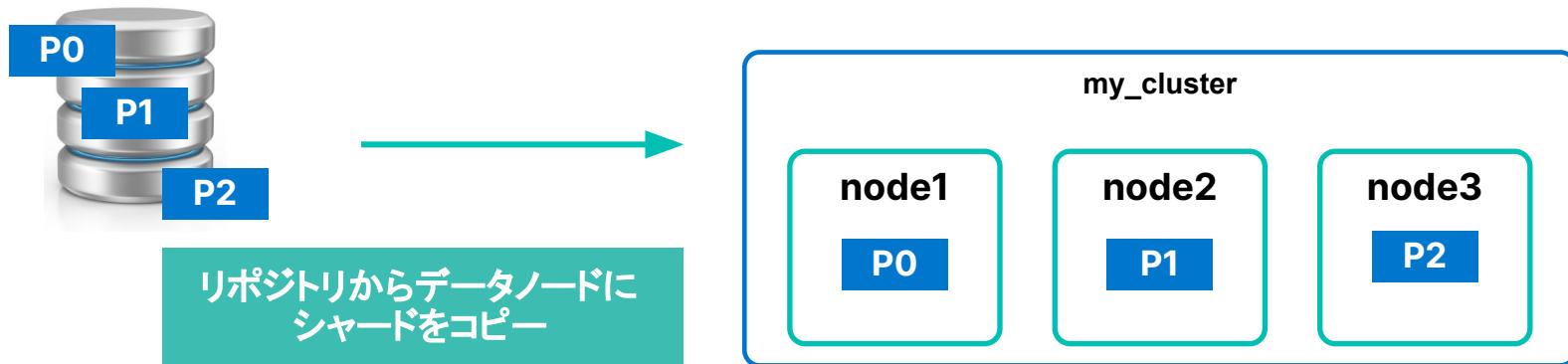
サーチャブル スナップショット

サーチャブルスナップショット

- Cold フェーズと frozen フェーズでは サーチャブルスナップショットと呼ばれるアクションが利用できる
- 利点:
 - とてもコスト効率の良い方法で凍結したデータを検索
 - ストレージコストの削減 (レプリカシャードを持たない)
 - すでに利用しているのと同じ仕組みを使う (スナップショット)

サーチャブルスナップショットの仕組み

- サーチャブルスナップショットインデックスの検索は、通常のインデックスの検索と同じ
 - インデックスのスナップショットを検索するには、インデックスを一時的なインデックスとしてローカルにマウントする必要がある
 - インデックスのシャードはクラスタのデータノードに割当てられる



サーチャブルスナップショットのセットアップ

- cold, frozen フェーズでサーチャブルスナップショットを設定するにはシンプルに登録されたリポジトリを選択すればよい:

The screenshot shows the 'Cold phase' configuration in the Elasticsearch Phase Settings interface. The 'Cold phase' is selected, indicated by a blue checked icon. The 'Move data into phase when:' field is set to 7 days old. A pink oval highlights the 'Snapshot repository' section, which contains the 'my_snapshot' repository. Below this, a note states 'Each phase uses the same snapshot repository.' At the bottom right, there is a 'Delete data after this phase' button with a trash can icon.

Cold phase

Move data into phase when: 7 days old

Snapshot repository

my_snapshot

Each phase uses the same snapshot repository.

Convert to fully-mounted index

> Advanced settings

Delete data after this phase

サーチャブルスナップショットを ILM に追加する

- ILM ポリシーを変更し、*cold* や *frozen* フェーズにサーチャブルスナップショットを追加
 - ILM で自動的にインデックスをマウント
 - Cold フェーズはインデックスを完全に (*fully*) マウント
 - Frozen フェーズはインデックスを部分的に (*partially*) マウント
- *Delete* フェーズがアクティブなら、デフォルトでサーチャブルスナップショットを削除する:
 - 無効にするには "`delete_searchable_snapshot": false`"
- ポリシーがデータストリームに適用されると、サーチャブルスナップショットはデフォルトで検索に含まれる

サーチャブルスナップショットの利点

- 大量の過去データを管理するのに最適
 - 検索やシャード割り当ては通常のインデックスと同じ扱い
- レプリカが不要、スナップショット自身がレプリカとして振る舞う
- ストレージスペースを劇的に削減
- 検索速度は低速だが、検索できる

Summary:

サーチャブル スナップショット

Module 8 Lesson 3



Summary

- ***Snapshot and Restore API*** で稼働中の Elasticsearch クラスタから取得したバックアップの作成、管理ができる
- スナップショットは “ある時点” のデータのコピーである
- ***Snapshot Lifecycle Management (SLM)*** でスナップショットを自動化できる
- サーチャブルスナップショットでは古くなったデータを多くのリソースを消費することなくクラスタに残しておくことができる
- サーチャブルスナップショットはインデックスライフサイクル管理ポリシーの一部として自動化できる

Quiz

1. ILM でサーチャブルスナップショットを設定するのに必要な唯一の設定とは?
2. **True or False:** サーチャブルスナップショットの検索は遅い
3. **True or False:** 単一の REST リクエストでクラスタ全体のスナップショットを作成できる

サーチャブル スナップショット

Lab 8.3

リポジトリをセットアップして、ILM ポリシーに
サーチャブルスナップショットを追加しましょう



More Resources

- More details on the various ILM actions:
 - www.elastic.co/guide/en/elasticsearch/reference/current/ilm-actions.html
- Data management:
 - www.elastic.co/blog/elasticsearch-data-lifecycle-management-with-data-tiers
- Frozen tier:
 - www.elastic.co/blog/introducing-elasticsearch-frozen-tier-searchbox-on-s3
- Searchable snapshots:
 - www.elastic.co/blog/introducing-elasticsearch-searchable-snapshots

Conclusion

Resources

- <https://www.elastic.co/learn>
 - <https://www.elastic.co/training>
 - <https://www.elastic.co/community>
 - <https://www.elastic.co/docs>
- <https://discuss.elastic.co>
 - <https://ela.st/training-forum>
- <https://ela.st/slack>

Elastic Certification

Validate Skills

Apply practical knowledge with performance-based testing



Elasticsearch
Engineer

Boost Productivity

Overcome obstacles with confidence and ease as you move from dev to production



Data Analyst

Open New Doors

Enhance professional visibility and expand opportunities

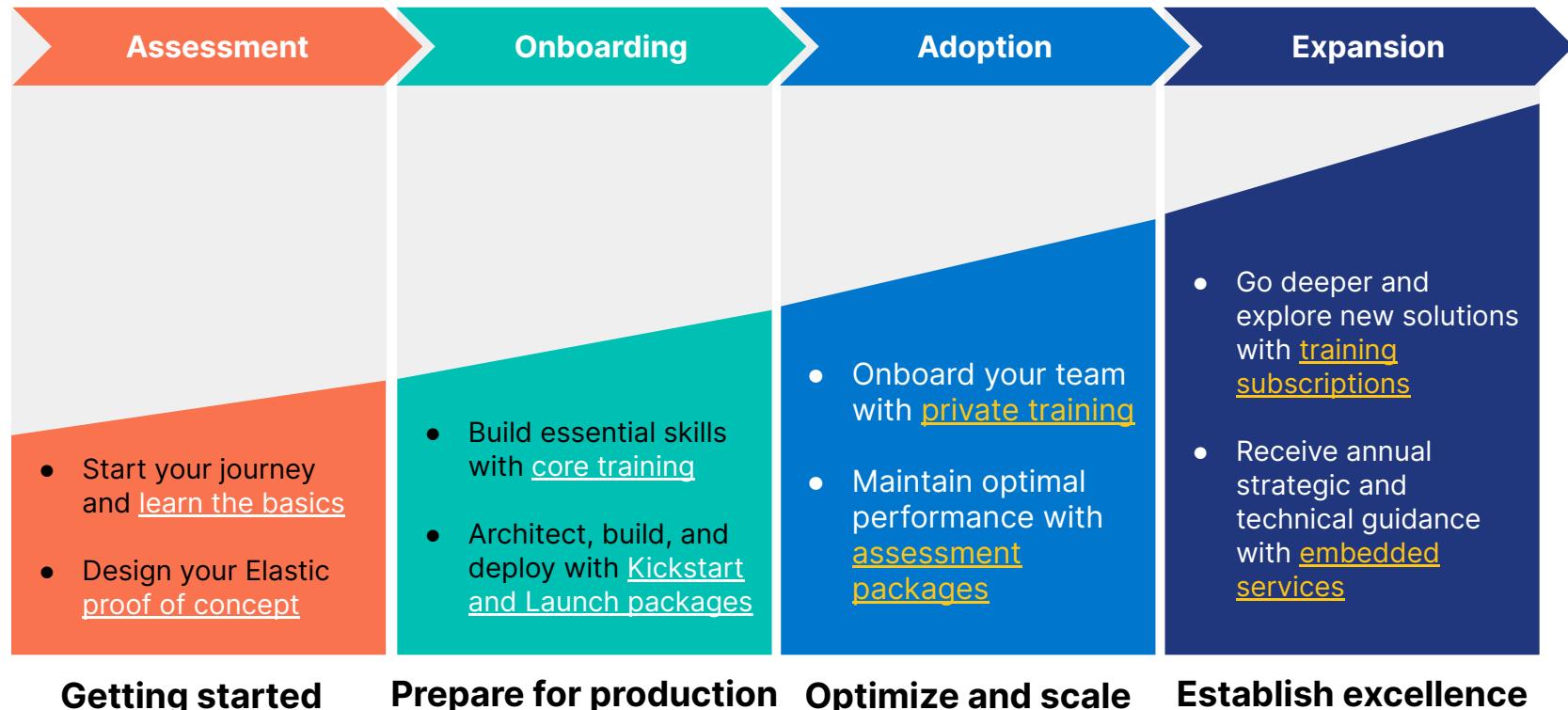
Join the Community

Become part of an exclusive network of over 1,000 certified professionals in nearly 70 countries



Observability
Engineer

Elastic Enablement Journey





Thank You.

Please complete
the survey



Quiz Answers

1.1 Elastic Observability

1. False. Observability is not just about monitoring uptime, but also logs, metrics, apm, user experience, and synthetic data.
2. False. Observability is an attribute of a system you build and want to monitor.
3. True. Elastic Observability provides you with a single stack to unify your logs, metrics, uptime data, application traces, user experience data, and synthetics. Ingest your data directly to Elasticsearch, where you can further process and enhance the data, before visualizing it in Kibana.

1.2 Uptime

1. ICMP, TCP and HTTP
2. False. It also shows expiration time on TLS certs for https monitors.
3. False. A single Heartbeat instance can monitor multiple services on multiple systems.

1.3 Discover

1. Which fields are populated, searching for similar records, exploring specific records.
2. True. Most integrations already include the data views you need to explore data, but you can always create your own data views too.
3. True. To make it easier to find the results you want, Discover allows you to filter fields for specific values.

2.1 Elastic Agent

1. False. With Elastic Agent you can collect all forms of data from anywhere with a single unified agent per host.
2. True. The integrations that come with Elastic Agent provide an easy way to connect Elastic to external services and systems, and quickly get insights or take action.
3. True. Elastic Agent policies specify which integrations you want to run and on which hosts.

2.2 Logs

1. Timestamp + Data
2. False. Providing web server log files throughout the company is not a safe practice and also not scalable. Providing Kibana is a much better way to solve this problem.
3. True. Elastic Agent is a single, unified agent that you deploy to hosts or containers to collect data and send it to the Elastic platform.

2.3 Metrics

1. False. Even though most of the monitored values are numeric, a typical Elasticsearch document that contains services' metrics will contain numeric and string values. Also, a system can return a string as the metric value being monitored. For example, the status of a docker container.
2. True. Remember that you can use the Elastic Agent to process log files and also for gathering periodic metrics.
3. True. Elasticsearch prefers the ISO 8601 format for timestamps with the timezone included, so Kibana can adapt to the local timezone of the user.

3.1 Elastic APM

1. False. Real User Monitoring captures user interaction with clients such as web browsers, while distributed tracing enables you to analyze performance throughout your microservices architecture all in one view.
2. APM agents, Elastic APM integration, Elasticsearch and Kibana.
3. True. To reduce overhead and storage requirements, you can start with a sample rate of 0.1.

3.2 Java agent

1. False. The first step is downloading the agent and start your application with the -javaagent flag pointing to the agent path. You need to declare a dependency to your application if you want to use programmatic API setup or the Public API.
2. True. The public API of the Elastic APM Java agent lets you customize and manually create spans and transactions, as well as track errors.
3. True. The **service_name** is used to group events and when it is not defined the agent will try to guess it for you.

3.3 Node.js agent

1. True. You need to add the elastic-apm-node module as a dependency to your application, so you can start it.
2. True. It's important that the agent is started before you require any other modules in your Node.js application.
3. False. You can also use the agent config file or environment variables.

3.4 RUM agent

1. True. You need to enable the RUM endpoint in the Elastic APM integration, though it is enabled by default.
2. False. It sends to the configured **serverUrl**.
3. True. However, it only includes requests made to the same origin. In order to include cross-origin requests, you must set the **distributedTracingOrigins** configuration option.

4.1 Logs app

1. True. You can use the Logs app to monitor all of the log events flowing in from your servers, virtual machines, and containers in a centralized view.
2. False. It is possible to stream live to view a continuous flow of log messages in real time.
3. True. You can consider the Logs app a tail -f in your browser, along with the power of search.

4.2 Metrics app

1. True. The Metrics app allows you to visualize infrastructure metrics to help diagnose issues in your infrastructure.
2. False. The Inventory page allows you to filter infrastructure metrics by hosts, Docker containers or Kubernetes pods that you are monitoring.
3. True. The Metrics Explorer page enables you to create time-series visualizations based on aggregation of your metrics.

4.3 APM app

1. True. Service Map is a real-time visual representation of the instrumented services in your application's architecture.
2. False. The APM app shows only basic host level metrics.
3. True. When analyzing traces waterfall it is possible to understand the parent/child hierarchy of transactions and spans, and ultimately determine why a request was slow.

4.4 User Experience app

1. True. User Experience provides a way to quantify and analyze the perceived performance of your web application.
2. False. Core Web Vitals are important metrics that are being used by search engines to rank websites based on user experience.
3. True. Understanding when and where your users are visiting from helps you prioritize optimizations.

5.1 Pipelines

1. Logstash pipelines or Elasticsearch ingest pipelines.
2. False. You can handle errors using on-failure processors, for example to save the document to a separate index for later analysis.
3. True. You can add a condition to processors so documents are only processed if they match this condition.

5.2 Extracting events

1. False. You need to handle whitespace that occurs in your messages.
2. False. The remove processor will delete unwanted fields from a document, while the drop processor allows for documents to be ignored by setting a simple conditional test.
3. `%{@timestamp} %{status.code} %{status.message}`

5.3 Transforming events

1. False. It assumes UTC.
2. MM/dd/yyyy - capitalized "M"s. Lowercased "m"s match minutes
3. locale

5.4 Loading events

1. True. The geoip processor provides this functionality through local databases, or custom databases.
2. a - 2 minutes, you can use the off-the-shelf user_agent processor to parse the user agent string so you don't need to spend 2 weeks writing a parser.
3. To create an enrich index (1) create an enrich policy then (2) execute it.

6.1 Machine Learning

1. True. You can create machine learning jobs for your observability data from either the Machine Learning app or the Observability app.
2. Finding them and evaluating them.
3. True. You can forecast trends with machine learning.

6.2 Custom ML jobs

1. True. With a multi-metric job, you can find anomalies based on fields that might seem unrelated, like the amount of web traffic on your website vs. the weather outside.
2. True. For example, you might be interested in finding anomalies on your CPU or RAM usage.
3. False. A population job detects activity that is unusual compared to the behavior of the population, while a categorization job groups log messages into categories to try finding anomalies on them.

6.3 Alerting

1. Condition, Schedule and Action
2. True. You can create an inventory rule that check either **system.fsstat.total_size.used** or **system.fsstat.total_size.free**.
3. False. You can either use the Observability apps or the Rules and Connectors UI in Kibana to create alerts.

7.1 Dashboards

1. True. Usually integrations come with prebuilt dashboards.
2. False. Not only can they be edited, visualizations from different Integrations may be combined in.
3. True. If you pin them.

7.2 Custom visualizations

1. Lens
2. Markdown
3. True. The geoip processor can be used to generate this data during ingest.

8.1 Data streams

1. False. It's using a data stream that has only one write index.
2. To make comparisons across many different log formats possible.
3. Rollover

8.2 Index Lifecycle Management

1. True
2. Hot/Warm/Cold/Frozen/Delete
3. 5 days. To execute this pattern, you'd set rollover in hot to 1 day, and then set the min_age in the cold phase to 5 days.

8.3 Searchable snapshots

1. The repo name. The repo must first be set up using the Snapshot and Restore API.
2. True. It's a tradeoff between cost and speed.
3. True

Elastic Observability Engineer

© 2015-2022 Elasticsearch BV. All rights reserved. Decompiling, copying, publishing and/or distribution without written consent of Elasticsearch BV is strictly prohibited.