



# Apache Kafkaによる無停止運用

株式会社INTHEFOREST

# 自己紹介

---



## 富田 和孝 (@railute)

肩書き： 株式会社INTHEFOREST 代表取締役社長  
Cassandra商用サポート、Cassandraコンサルティング他  
日本Cassandraコミュニティー

Cassandra勉強会主宰

2か月に一度程度開催。現在、第35回まで開催。

職種：本職はDB・インフラ系エンジニア

以前、某レストランサーチのDBA

高負荷・大容量・大規模のOracleRACとPostgreSQLと  
MySQLに苦しめられ続けた経験あり。

NLPおよびテキストマイニング悩み中。

(実はもともと言語学(日本語)専攻。)

---



# 自己紹介



DATASTAX

## Kazutaka Tomita

\*\*\* HAS COMPLETED \*\*\*

### Apache Cassandra™ Professional Certification

as a part of the:  
DataStax Certification Program.



Tim Berglund  
Global Director of Training

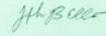


DATASTAX | O'REILLY

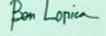
## Certified Administrator on Apache Cassandra

Kazutaka Tomita  
Has successfully completed the requirements for the Administrator Certification for Apache Cassandra, V2.1

Date of achievement: 9/22/2015



Jonathan Ellis, DataStax



Ben Lorica, O'Reilly Media, Inc.





# Agenda

---

- ▶ BigData基盤とHARUNIRE
- ▶ Apache Zeppelin
- ▶ Apache Kafuka
- ▶ fluentdとApache Kafka
- ▶ Apache Kafka+Spark Streaming + Apache Cassandra



# 例など

---



- ▶ Oracle (PL/SQL) から Kafka + Streaming + Cassandra へ
- ▶ IoT のデータ集約基盤として
- ▶ Harunire 有ります。





---

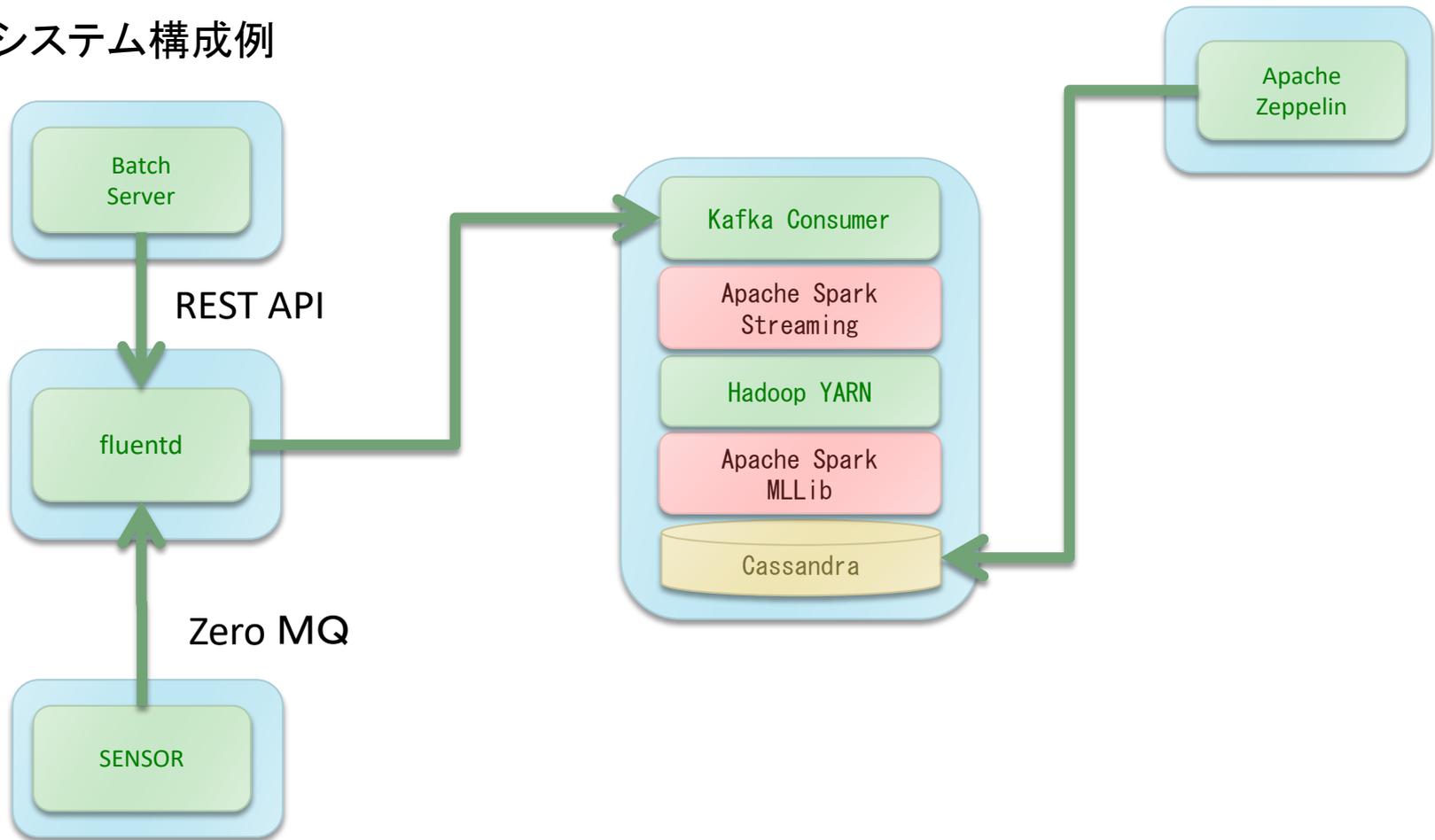
# BigData基盤とHARUNIRE



# BigData基盤とHARUNIRE



## システム構成例





---

# BigData基盤構成要素





## Webベースのインタラクティブデータ解析UI

- ブラウザ実行
- 各種データソース対応
- 簡易グラフ機能付き



# Apache Zeppelin



 **Zeppelin**

Notebook ▾

Search your Notebooks



anonymous ▾

/test1/test1



   default ▾

```
%cassandra  
USE "test1";  
SELECT * FROM test1;
```

FINISHED    



id	intvalue	textvalue
5	5	e
1	1	a
2	2	b
4	4	d
3	3	c

Took 0 sec. Last updated by anonymous at September 23 2016, 1:01:51 PM. (outdated)



# Apache Zeppelin



**Zeppelin** Notebook ▾

Search your Notebooks 🔍

anonymous ▾

/test1/test1

▶ ⌂ 📄 🗑️ 🔄

🖨️ ⚙️ 🔒 default ▾

```
%cassandra  
USE "test1";  
SELECT * FROM test1;
```

FINISHED ▶ ⌂ 📄 ⚙️

📊 📈 📉 📌 📍 ⌵ settings ▾

● 1 ● 2 ● 3 ● 4 ● 5

Took 0 sec. Last updated by anonymous at September 23 2016, 1:01:51 PM. (outdated)





# DEMO





# Apache Kafka



# Apache Kafka

---



## ▶ 特徴

### ▶ Fast

1つのブローカーで数千のクライアントから毎秒数百MBの読込、書込を処理することが可能。

### ▶ Scalable

大規模なプラットフォームに耐えられるように、弾力的且つ透過的に拡張することが可能。

### ▶ Durable

メッセージはクラスタ内で複製されて保持するので欠損することが無い。

### ▶ Distributed by Design

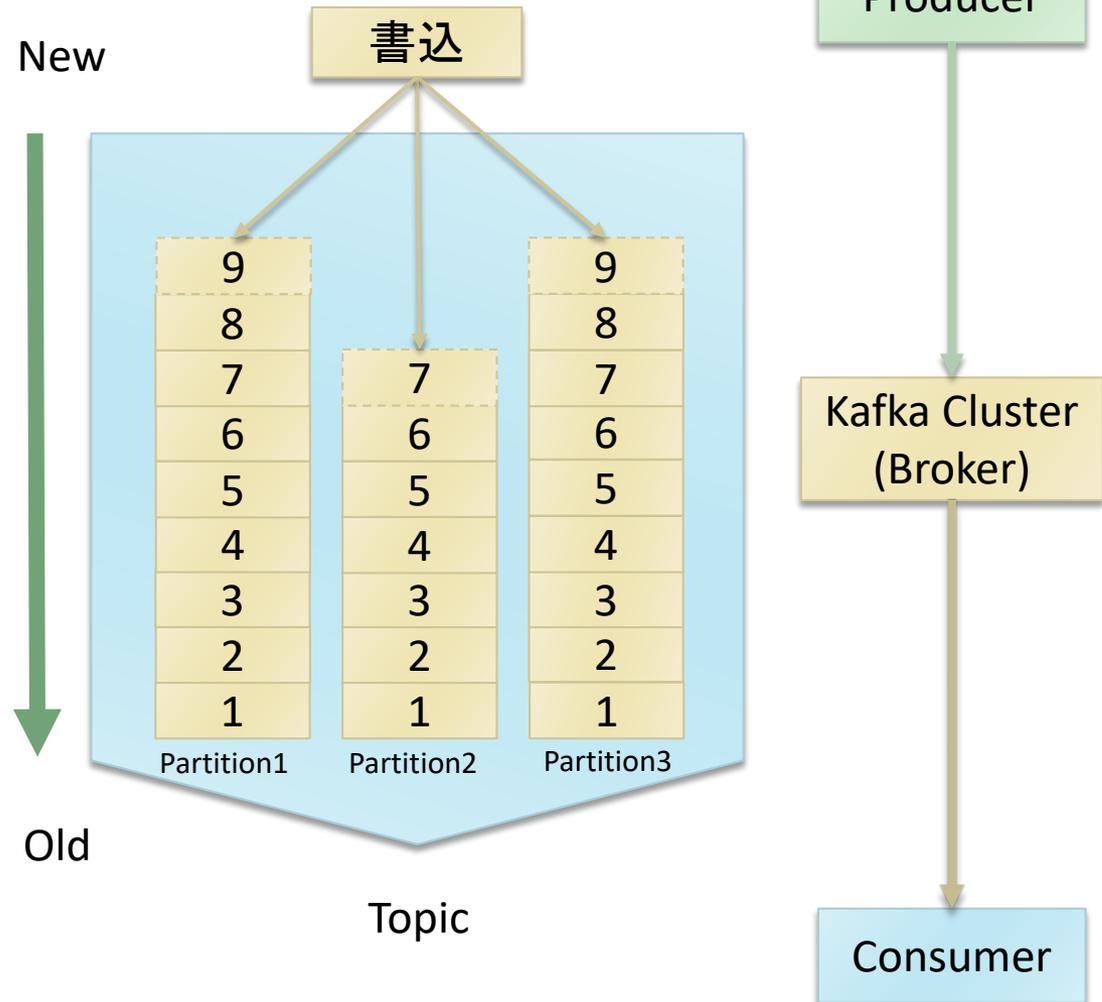
強い耐久性と耐障害性の保証出来るようにデザイン。



# Apache Kafka



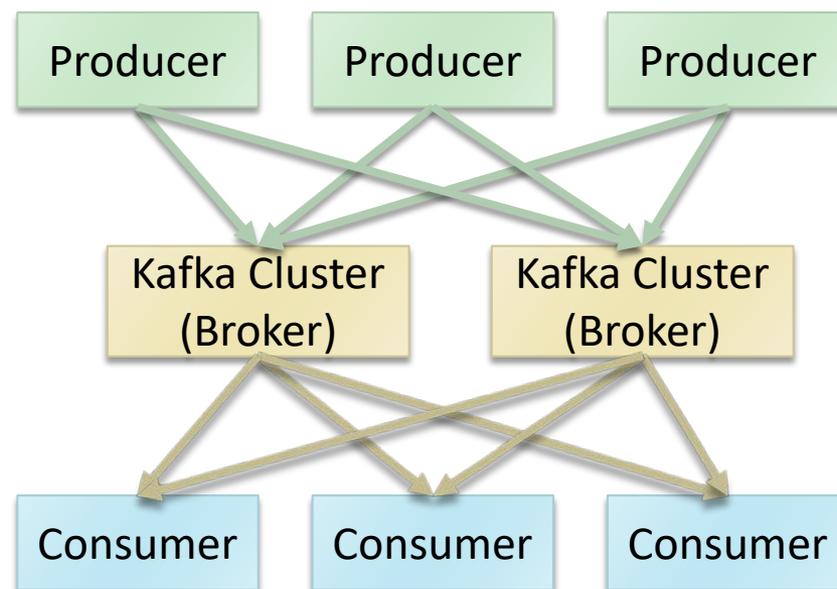
1. トピックと呼ばれるメッセージの場所を提供
2. トピック内に複数のパーティションを維持
3. メッセージを順序付IDを割り当てて保持
4. 各パーティションは設定期間メッセージを保持



# Apache Kafka



1. プロデューサーは指定のトピックにメッセージを配信する。
2. Brokerは各種トピックを管理。
3. コンシューマーは指定のトピックからメッセージを取り出す。



# Apache Kafka

---



## ▶ Kafkaが保証する物

- ▶ プロデューサーにより特定のトピックパーティションに送信されたメッセージのBroker内での順序。
- ▶ コンシューマーがブローカーからメッセージを取得する際のメッセージの順序
- ▶ トピックのレプリカ数から1引いた数分のブローカー障害までメッセージとサービス



# Apache Kafka

---



## 特徴

- メッセージ・バッファリング
- 耐障害性
- アプリケーション間通信の非同期化

## 想定用途

- Messaging
  - Website Activity Tracking
  - Metrics
  - Log Aggregation
  - Stream Processing
  - Event Sourcing
  - Commit Log
- 





---

# fluentdとApache Kafka



# fluentdとApache Kafka

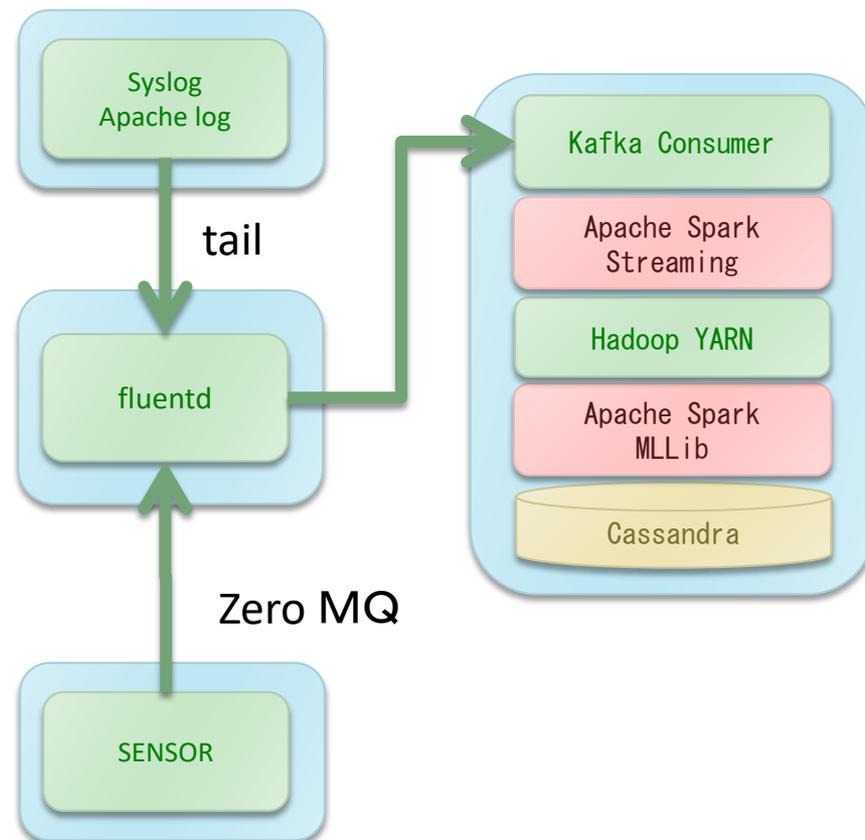


## fluentd

豊富なプラグインを持つEventCollector。  
プラグインを使用することにより多彩な  
インターフェースを容易に構築

各種inputpluginを用いてMQTTや  
ZeroMQ、HTTPの受付、Tailを用  
いてのLogの読込など。

outputpluginに  
「fluent-plugin-kafka」  
fluentd内にプロデューサーを作成





---

# Apache KafkaとSpark Streaming



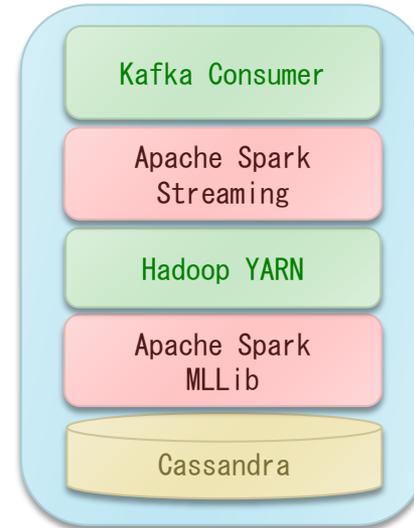
# Apache KafkaとSpark Streaming



## Spark Streaming

Streaming用マイクロバッチ  
ショートウィンドウでのデータ集約などを行う。

Spark Streaming内にConsumer作成





# まとめ

---

- ▶ 単独から分散
- ▶ 非同期システム
- ▶ 耐障害性向上
- ▶ 止まらないシステムへ

