



**adaptTo()**

APACHE SLING & FRIENDS TECH MEETUP  
BERLIN, 22-24 SEPTEMBER 2014

# Integrating ElasticSearch with AEM

Vivek Sachdeva, Intelligrape Software

- Introduction to Elasticsearch
- Analyzing Data
- Integration with AEM
- Approaches Considered
- Features Integrated
- Extension Points

# About Me :)

- Started career as Java Developer in 2010
- Worked on Groovy and Grails
- Currently working as AEM(plus Groovy) Developer
- Intelligrape Software
- [vivek.sachdeva@intelligrape.com](mailto:vivek.sachdeva@intelligrape.com)
- [in.linkedin.com/pub/vivek-sachdeva/20/192/934/](https://in.linkedin.com/pub/vivek-sachdeva/20/192/934/)

# Elastic Search – An Introduction

# What & Why??

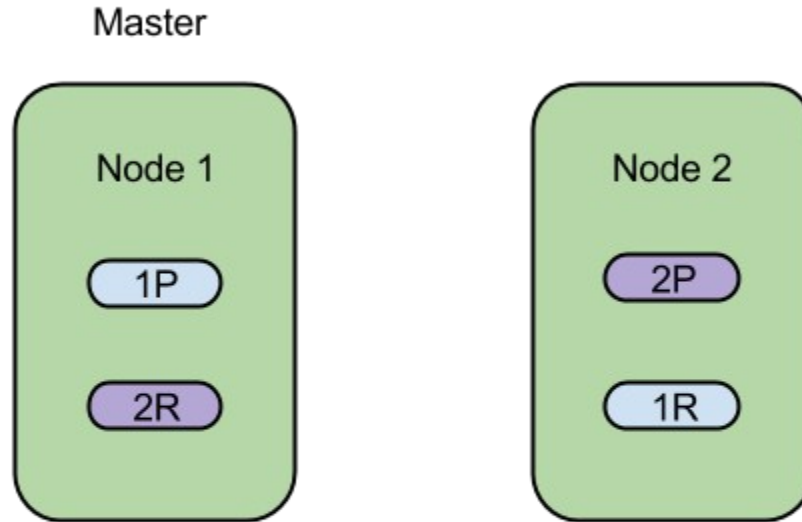
- Elasticsearch is a distributed, RESTful, free/open source search server
- Based on Apache Lucene
- Developed by Shay Banon
- Latest Version – 1.2

# What & Why??

- Document Oriented and Schema free
- Support for multi-tenancy
- Support for dynamic schema update
- High Performance
- Just like JCR

# ElasticSearch Server View

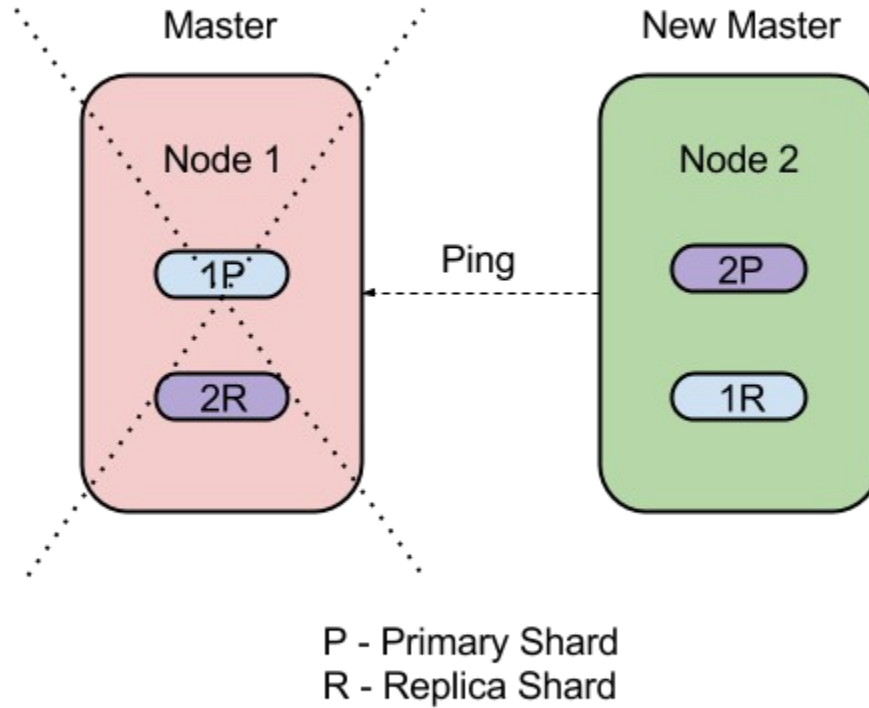
# Index Creation



P - Primary Shard  
R - Replica Shard



# Fail Safe



# Analyzing Data

# How to Index??

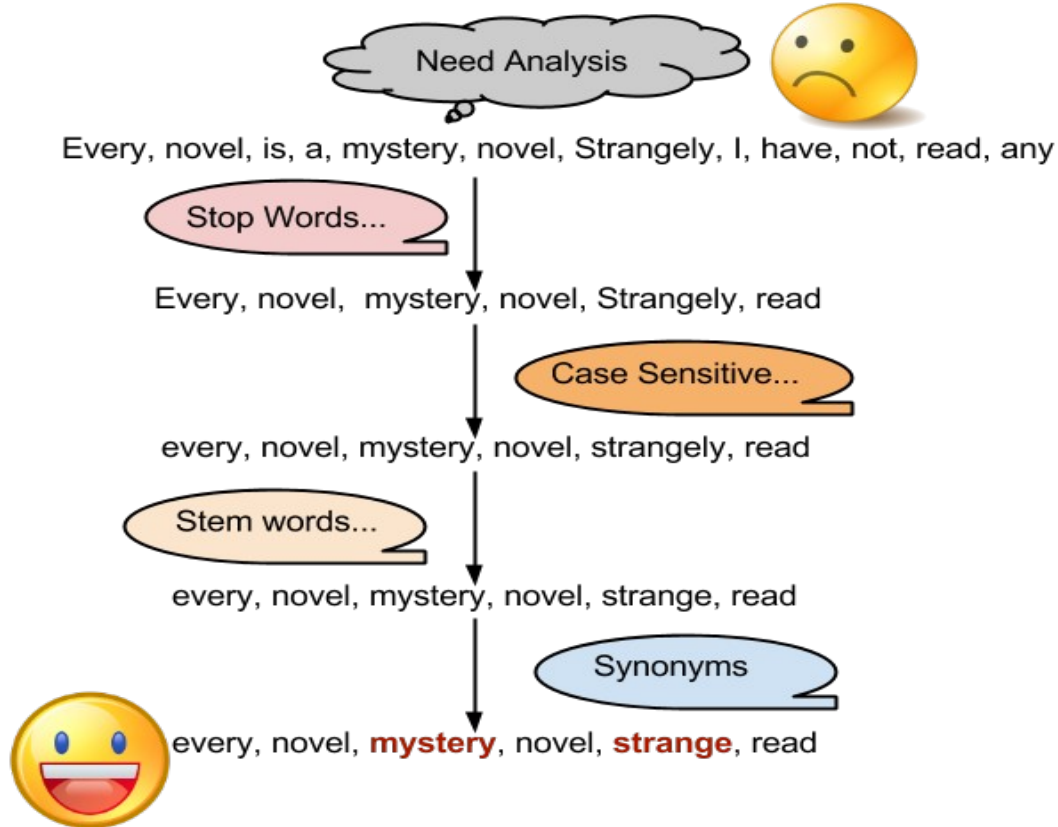
- First step is to identify the relevant data
  - Every novel is a mystery novel.
  - Strangely I have not read any.

# Bad Indexing

Every	1
novel	1
is	1
a	1
mystery	1
novel	1
Strangely	2
I	2
have	2
not	2
read	2
any	2

- Every novel is a mystery novel.
- Strangely I have not read any.

# Smart Indexing!!



# Integrating with AEM

# What All is Needed??

- ElasticSearch Server
- CQ instance.. :)

Thats it...

# ElasticSearch dependency

- Available in Maven repository

```
<dependency>
```

```
  <groupId>org.elasticsearch</groupId>
```

```
  <artifactId>elasticsearch</artifactId>
```

```
  <version>1.2.0</version>
```

```
</dependency>
```

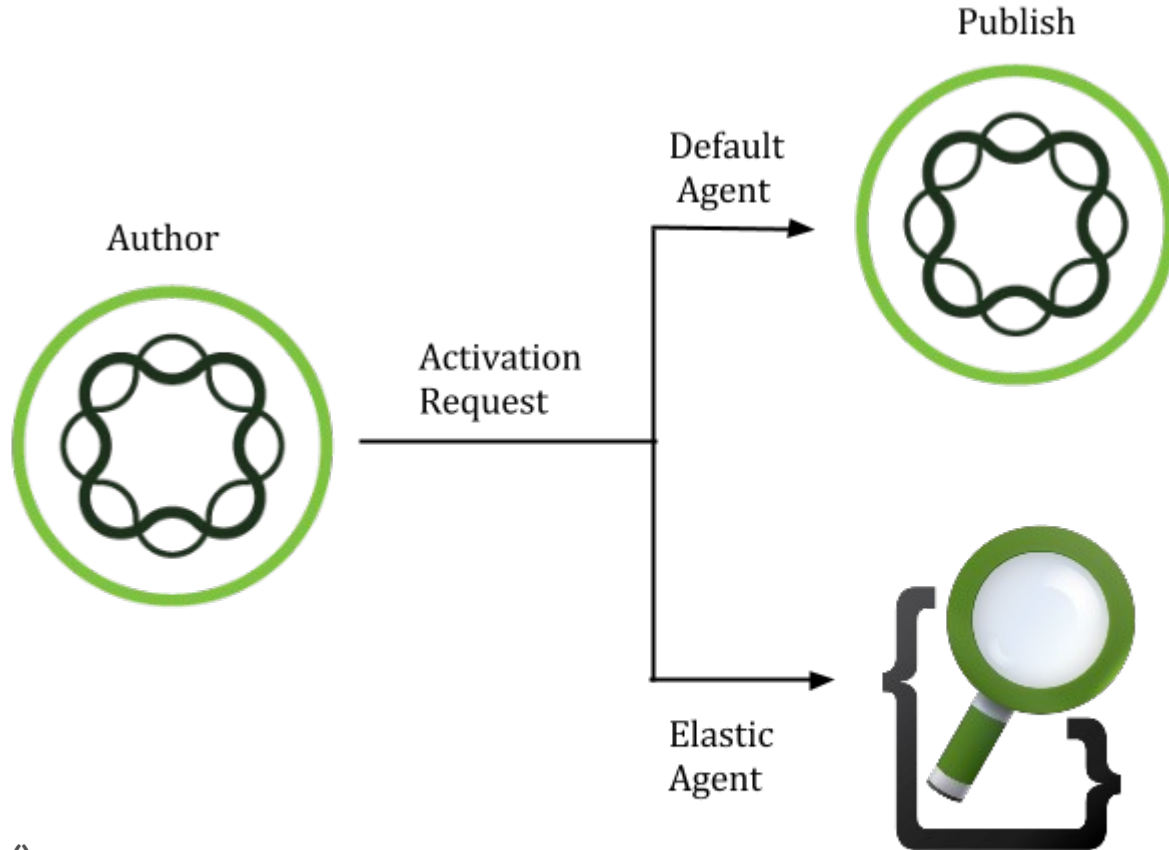


# Approaches Considered

# Push & Pull Indexing

- Push Indexing
  - CQ Server pushes data to Elastic Server whenever needed
- Pull Indexing
  - Elasticsearch Server pulls data at regular intervals and indexes it

# Push Indexing.. Using Replication Agent

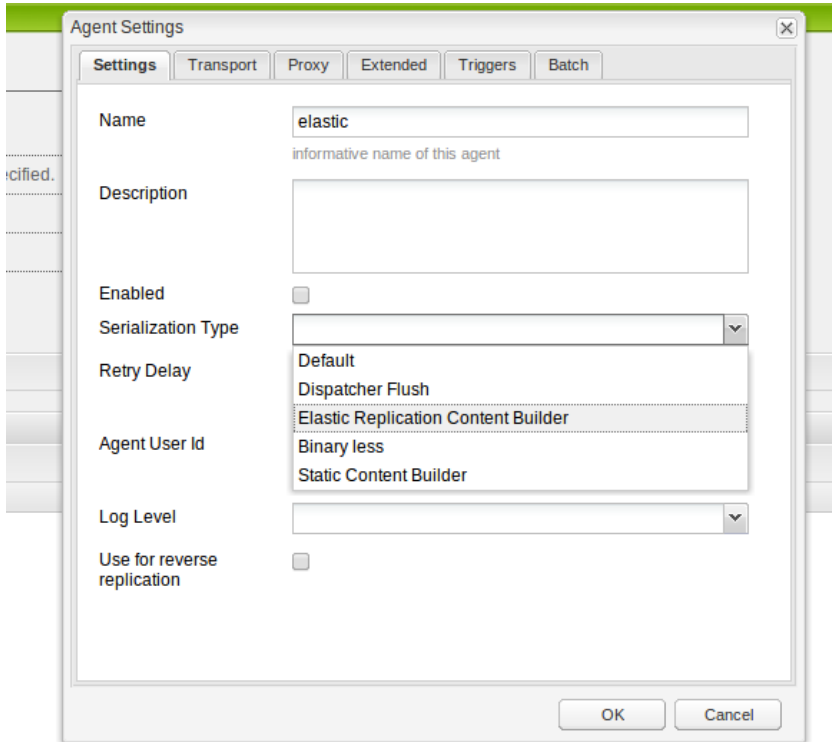


# Push Indexing... Using Replication Agent

- Create Serialization Type and setup a replication agent.

```
@Component(label = "Elastic Replication Content Builder", description = "This servlet
    replicates data to Elastic", immediate = true, enabled = true, metatype = true)
@Service(ContentBuilder.class)
class ElasticReplicationContentBuilderImpl implements ContentBuilder{
    //Override create method
}
```

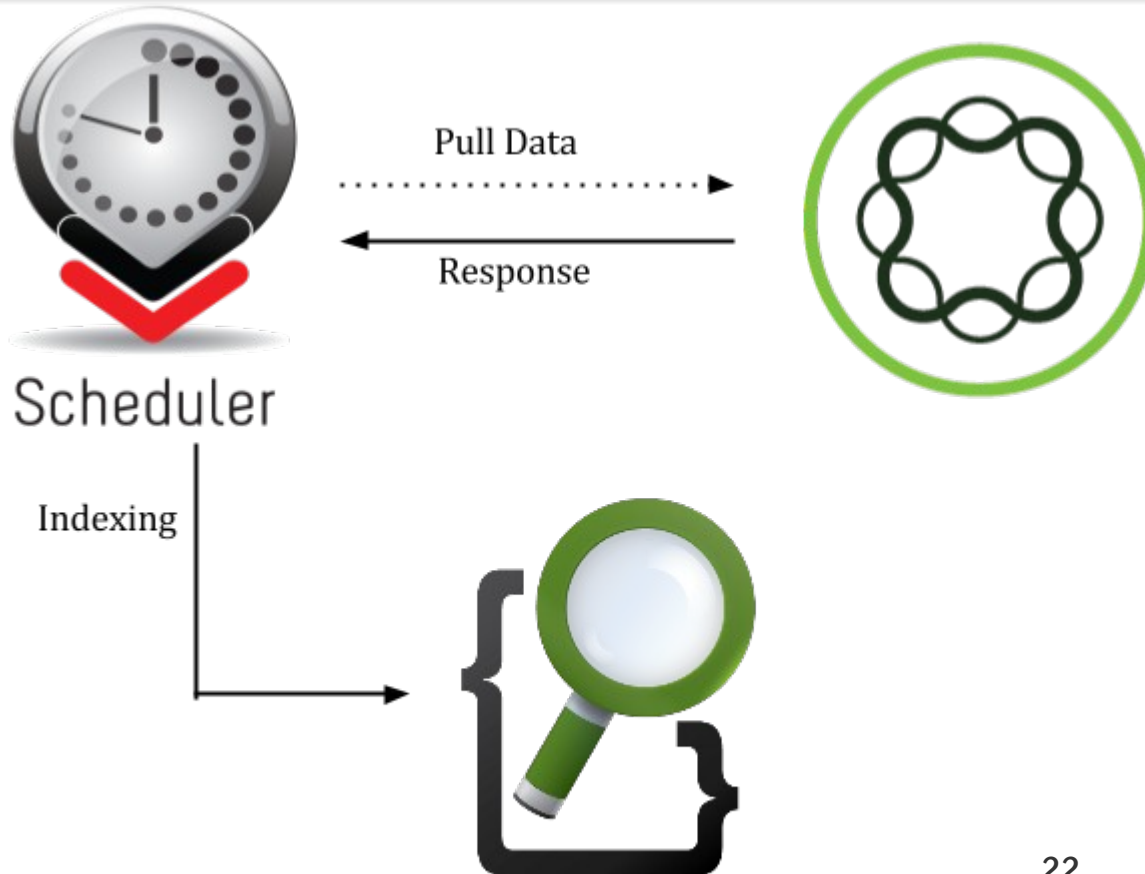
# Push Indexing... Using Replication Agent



## Pros:

- Data can be pushed as and when needed.

# Pull Indexing



# Features Integrated

# Features Integrated

- **Basic Aggregation**
  - Tag based faceting
  
- **Free Text Search**
  - Finding all pages in order of relevancy based on some text



## BRAND



Search by Brand



- Crocodile
- Flippd
- Proline
- Reebok
- Converse
- Being Human Clothing
- Chromatome

## SIZE



XS - 38

S - 39

M - 40

L - 42

XL - 44

XXL - 46

## PRICE



- Rs. 499 And Below
- Rs. 500 - Rs. 999
- Rs. 1000 - Rs. 1499
- Rs. 1500 - Rs. 1999
- Rs. 2000 And Above

## SLEEVE



- 3/4 Sleeve
- Full Sleeve
- Half Sleeve
- Roll-Up Sleeve
- Sleeveless

# DEMO

# More Cool Features/Extension Points...



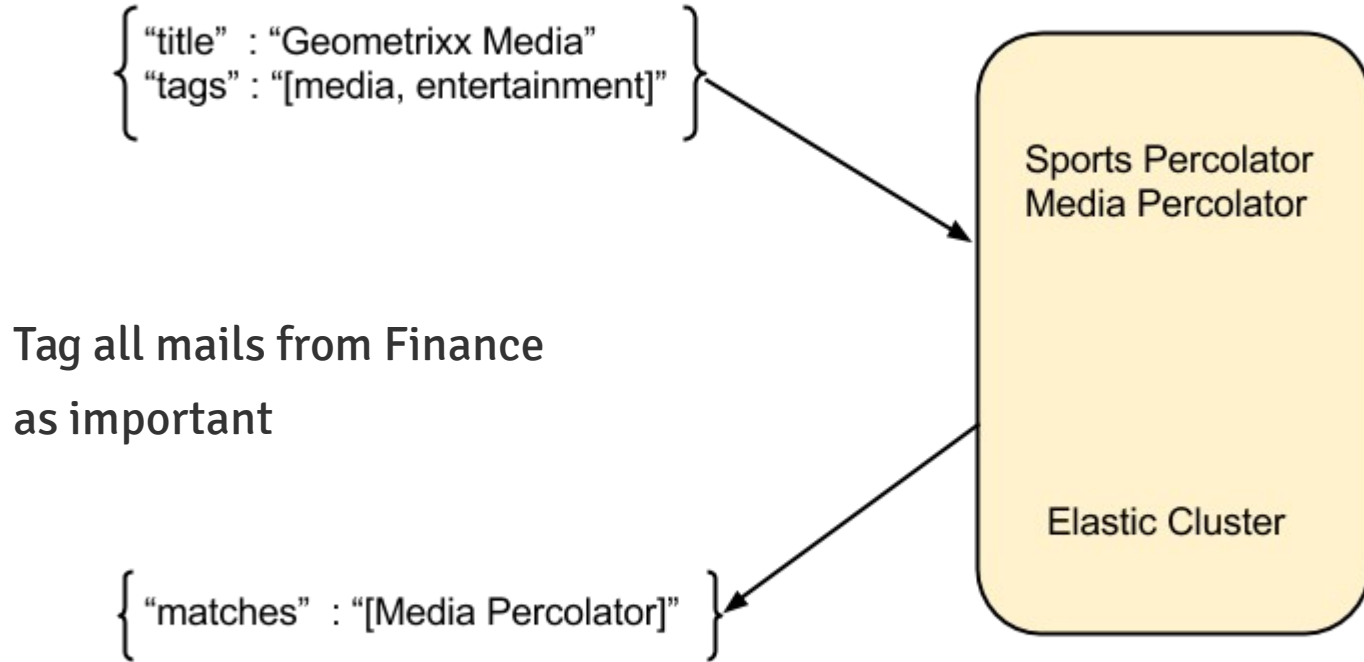
Lets say we need to  
find all restaurants  
that are within 1  
mile

# Advanced Aggregation

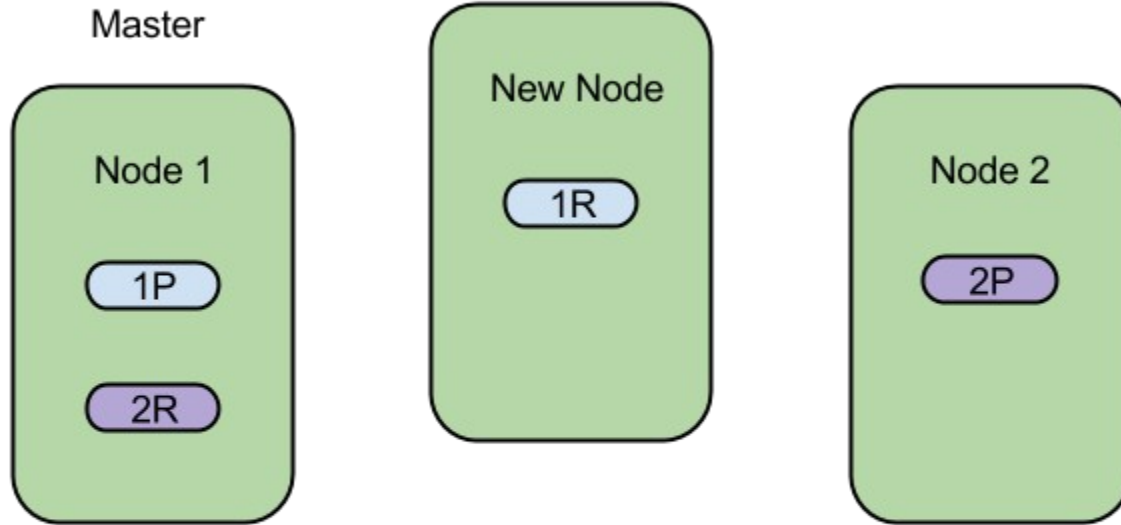
```
aggregations: {
  group_by_state: {
    buckets: [
      { key: wv
        doc_count: 1
        average_age: {
          value: 22
        }
      }
      { key: al
        doc_count: 2
        average_age: {
          value: 22
        }
      }
    ]
    .....
  }
}
```

Group all the employees of an MNC by project location and find average age of each group.

# Percolator – Search.... Reversed.....



# Ease of Clustering



P - Primary Shard  
R - Replica Shard

# DEMO



# Appendix

- <http://www.elasticsearch.com/>
- <http://www.elasticsearch.org/guide/>
- <https://forums.adobe.com>
- Google :)

# Where is the code??

- <https://github.com/viveksachdeva/elasticsearch-cq.git>

*Thank you*

