基于Elastic Stack的简单日志分析

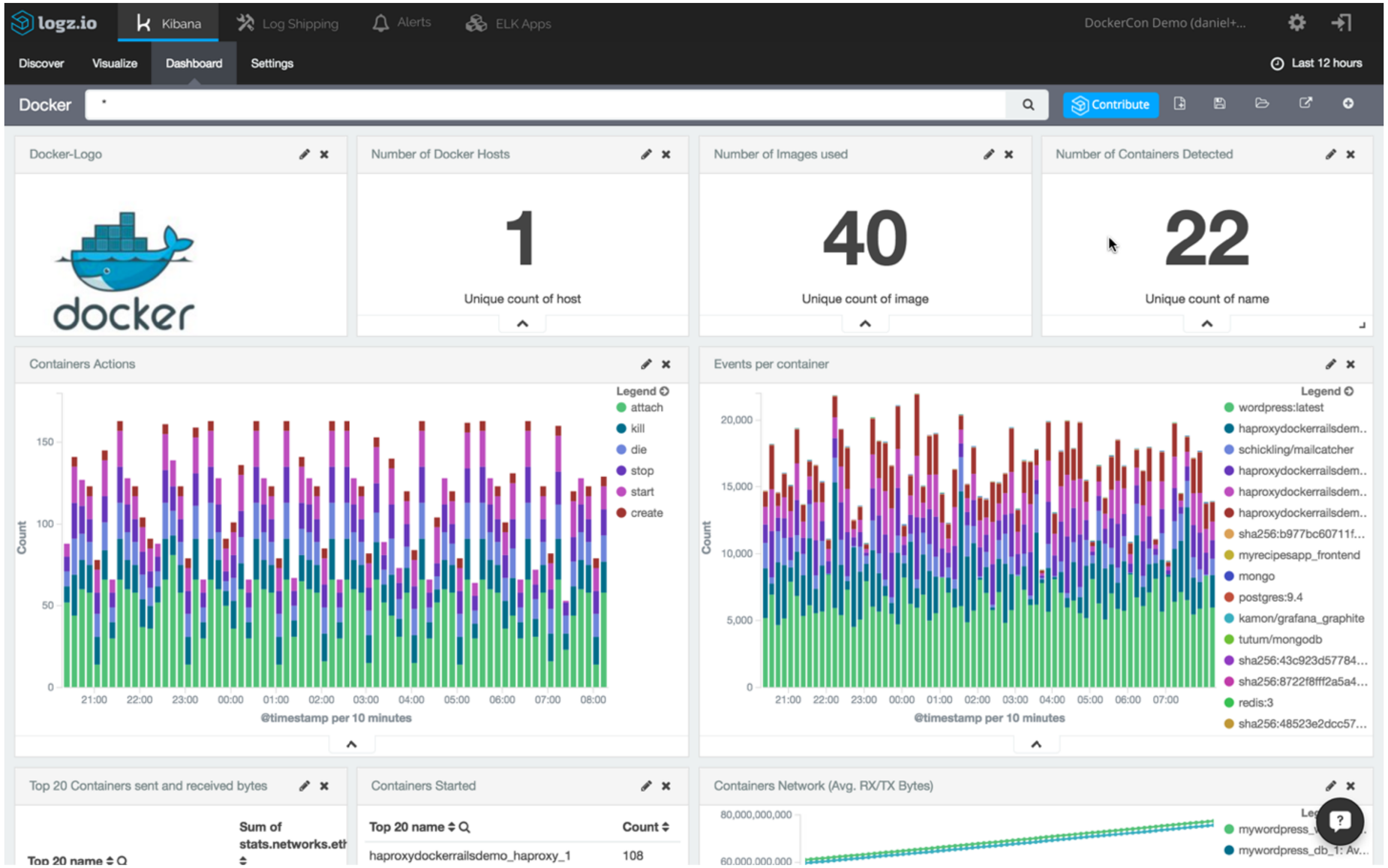
江水

关于我:
热爱前端的后端工程师

组织翻译了Vue2.x中文文档
https://vuefe.cn
https://github.com/vuefe/vuefe.github.io

so.com ~~sogou.com~~
监控中心、大日志平台、自动化运维平台、商业云
Docker、Kubernetes、Mesos

1.Elastic Stack究竟是什么
2.基于Elastic Stack分析Vue2.0中文站日志数据
3.Elastic Stack拓展与优化


logz.io
Kibana
Log Shipping
Alerts
ELK Apps
DockerCon Demo (daniel+...
Discover
Visualize
Dashboard
Settings
Last 12 hours
Docker
Contribute
Docker-Logo
docker
Number of Docker Hosts
1
Unique count of host
Number of Images used
40
Unique count of image
Number of Containers Detected
22
Unique count of name
Containers Actions
Legend
attach
kill
die
stop
start
create
Count
150
100
50
0
21:00
22:00
23:00
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
@timestamp per 10 minutes
Events per container
Legend
wordpress:latest
haproxydockerrailsdem..
schickling/mailcatcher
haproxydockerrailsdem..
haproxydockerrailsdem..
haproxydockerrailsdem..
sha256:b977bc60711f...
myrecipesapp_frontend
mongo
postgres:9.4
kamon/grafana_graphite
tutum/mongodb
sha256:43c923d57784...
sha256:8722f8fff2a5a4...
redis:3
sha256:48523e2dcc57...
Count
20,000
15,000
10,000
5,000
0
21:00
22:00
23:00
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
@timestamp per 10 minutes
Top 20 Containers sent and received bytes
Sum of stats.networks.eth
Top 20 name
Containers Started
Top 20 name
Count
haproxydockerrailsdemo_haproxy_1
108
Containers Network (Avg. RX/TX Bytes)
80,000,000,000
60,000,000,000
mywordpress_
mywordpress_db_1: Av...

谁在用Elasticsearch?

Elastic Stack：

Elasticsearch、Logstash、Kibana、Beats

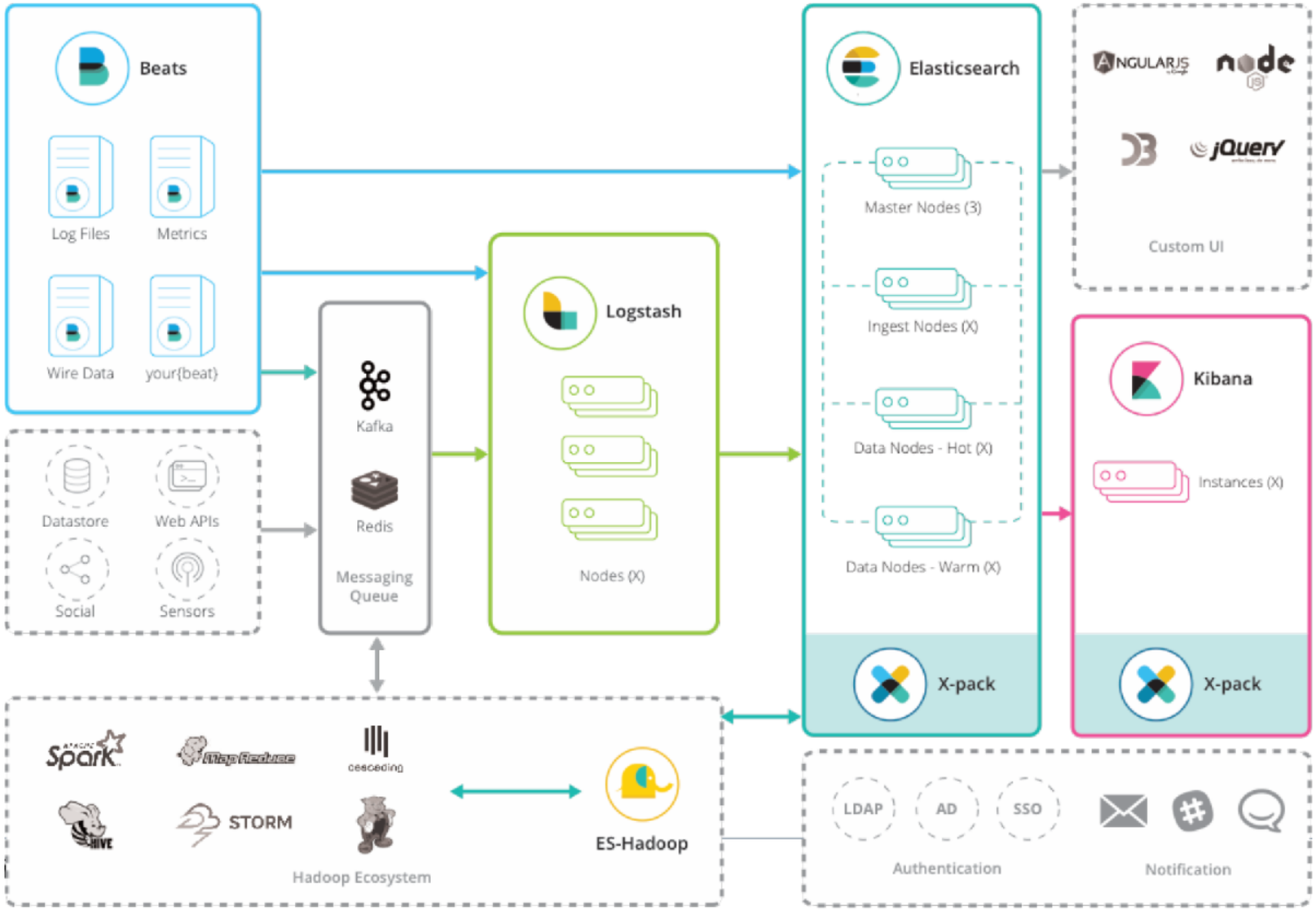
日志处理这个生态圈：

数据检索 Lucene、Solr

数据处理 Flume、Fluentd、Rsyslog、NXlog、Heka

数据展示 Grafana、Graphite

Elastic Stack



Elasticsearch：

A Distributed RESTful Search Engine.

一个分布式的REST风格的搜索引擎。

A New Near Real Time Search Engine.

一个新的准实时的搜索引擎。

Elasticsearch：

基于Java

基于Apache Lucene

同样基于Apache Lucene的还有Apache Solr

1.Solr利用Zookeeper进行分布式管理

Elasticsearch自身带有分布式协调管理功能

2.Solr支持更多格式的数据

Elasticsearch仅支持json文件格式

3.Solr官方提供的功能非常多

Elasticsearch更注重核心功能，高级功能有第三方插件

http://solr-vs-elasticsearch.com/

关于搜索引擎的几个基本概念：

结构化数据、非结构化数据、半结构化数据

全文检索 Full-text Search

正排索引 Forward Index、倒排索引 Inverted Index

分词 / 中文分词 Ansj、jieba、IK

Elasticsearch基本概念：

索引 index：

Elasticsearch将数据存数在一个或者多个索引中，可以向索引写入或读取文档。

写入和检索的过程，是由Lucene完成的。

Elasticsearch集群由一个或者多个Lucene索引构成。

过大的 Index 会导致超出磁盘的限制同时可能导致降低检索请求相应速度。

文档 document：

Elasticsearch中的主体，由字段构成。

映射 mapping：

每个索引的配置叫做映射。

类型 type：

Elasticsearch允许用户在一个索引中存储多种文档类型。

这是个伪概念。

Elasticsearch基本概念：

分片 shard：

Elasticsearch将数据存储在多个Lucene索引中，这种索引被成为分片。

副本 replica：

Elasticsearch会为每个分片创建冗余的副本。

节点 node：

单个Elasticsearch实例被称为节点。

默认情况下，一个节点既可以是master node也可以是data node。

随着集群的数据量增长再考虑设置独立的master node、data node、client node。

集群 cluster：

这些节点组成的系统就是集群。

对等架构 p2p：

节点会自动连接到集群中其他节点，进行数据交互。

这也是个伪概念，真正p2p架构的是Apache Cassandra。

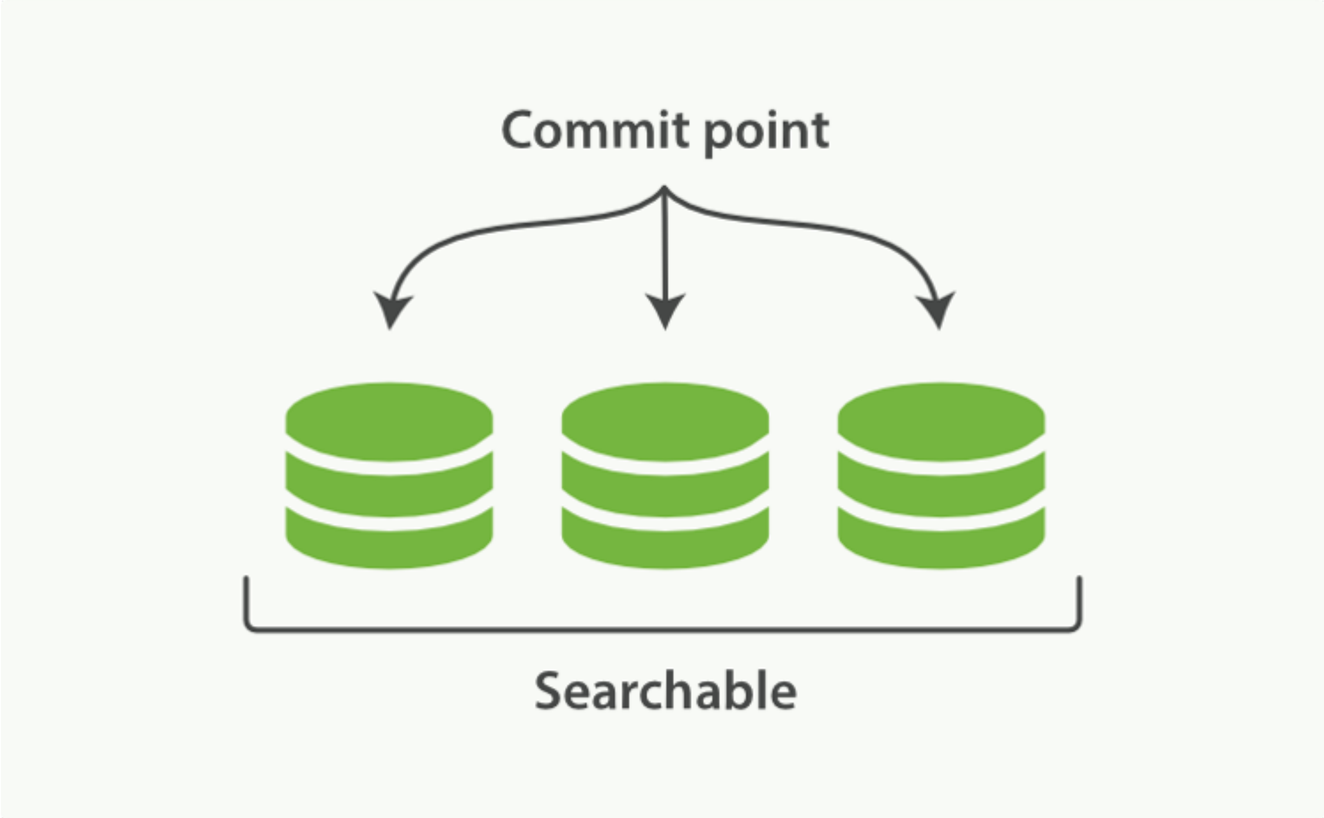
RDBMS vs Elasticsearch

RDBMS	Elasticsearch
Database数据库	Index索引
Table表	Type类型
Row记录	Document文档
Column列	Field字段
Schema表结构	Mapping映射
SQL语句	DSL领域特定语言

Near Real Time：

当前索引有3个segment可用。

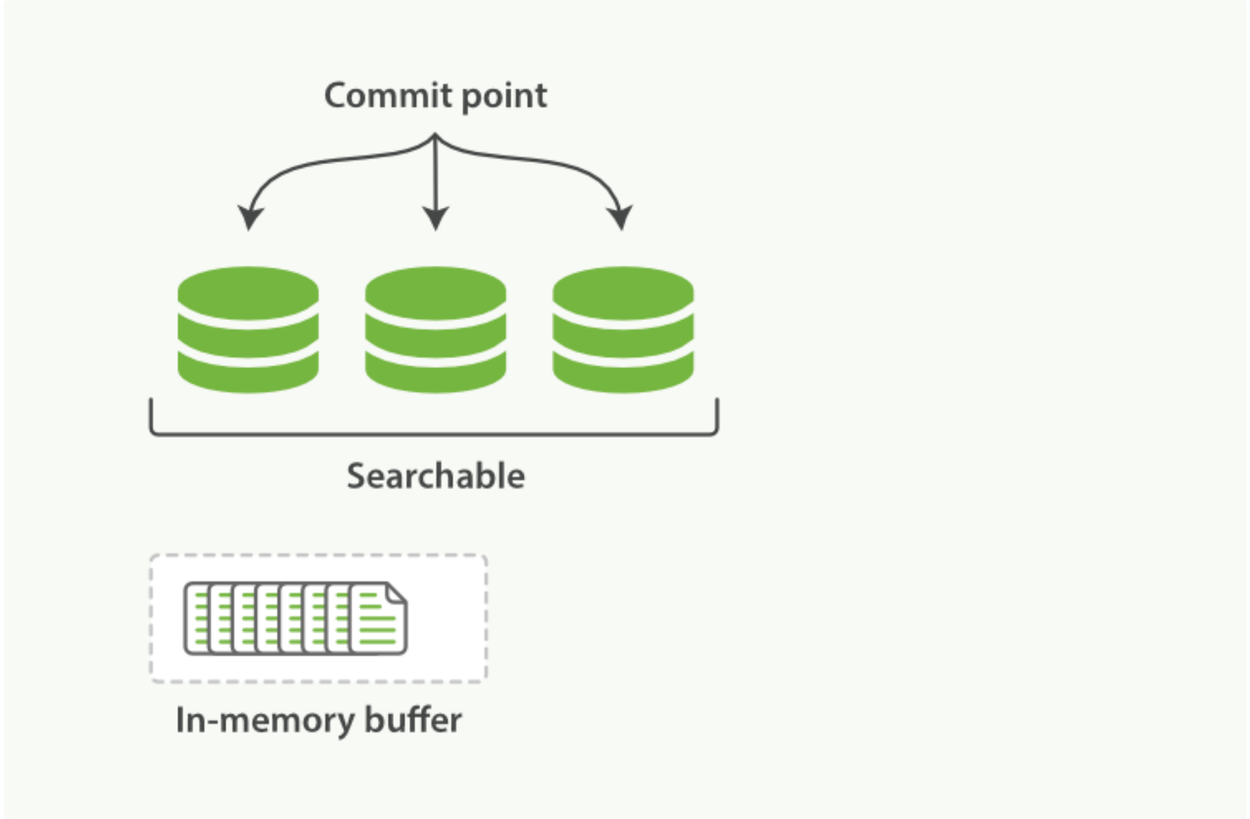
一个segment就是一个倒排索引。



Near Real Time：

新的文档首先写入内存buffer。

没有生成新的倒排索引，文档也不能被检索。

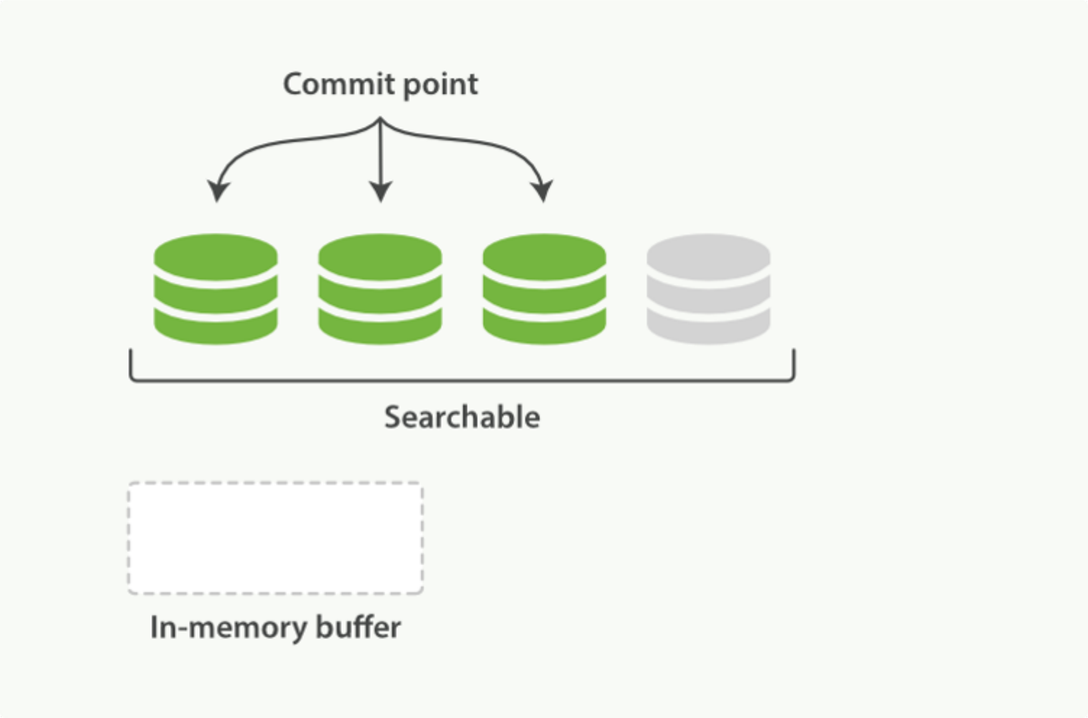


Near Real Time：

内存中的文档会被写到一个新的segment。

先被写到内核文件系统缓存，然后flush到磁盘。（1s）

flush过程很耗时，写到内核就可以被访问。（30m）

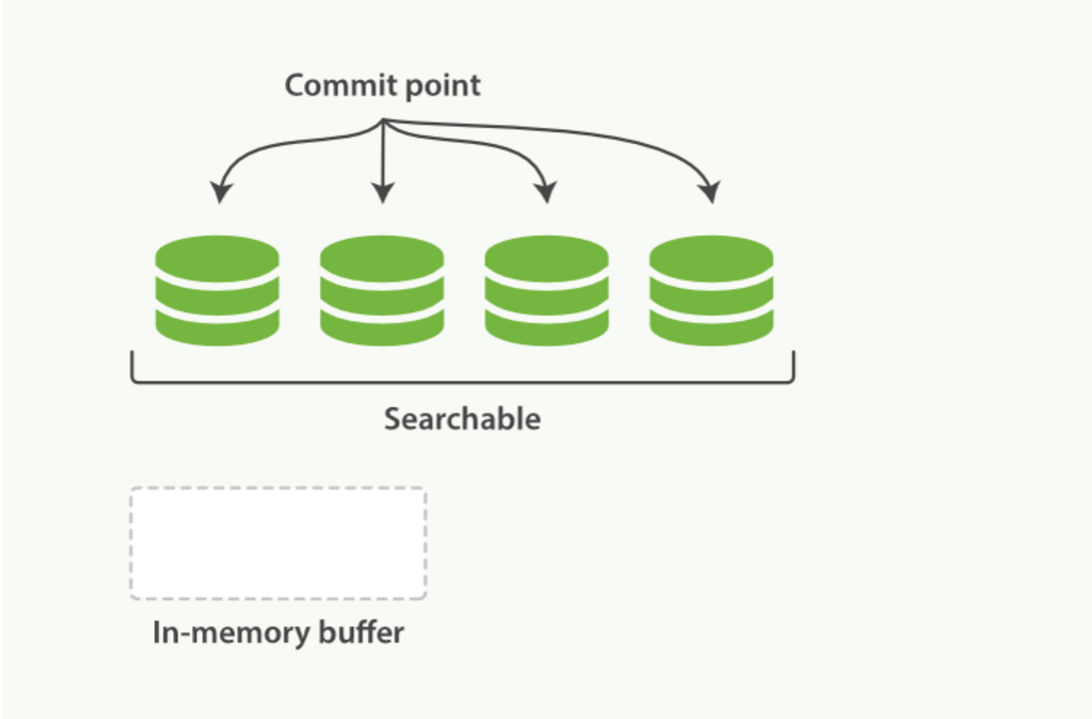


Near Real Time：

buffer被提交，一个新的segment被写到磁盘。

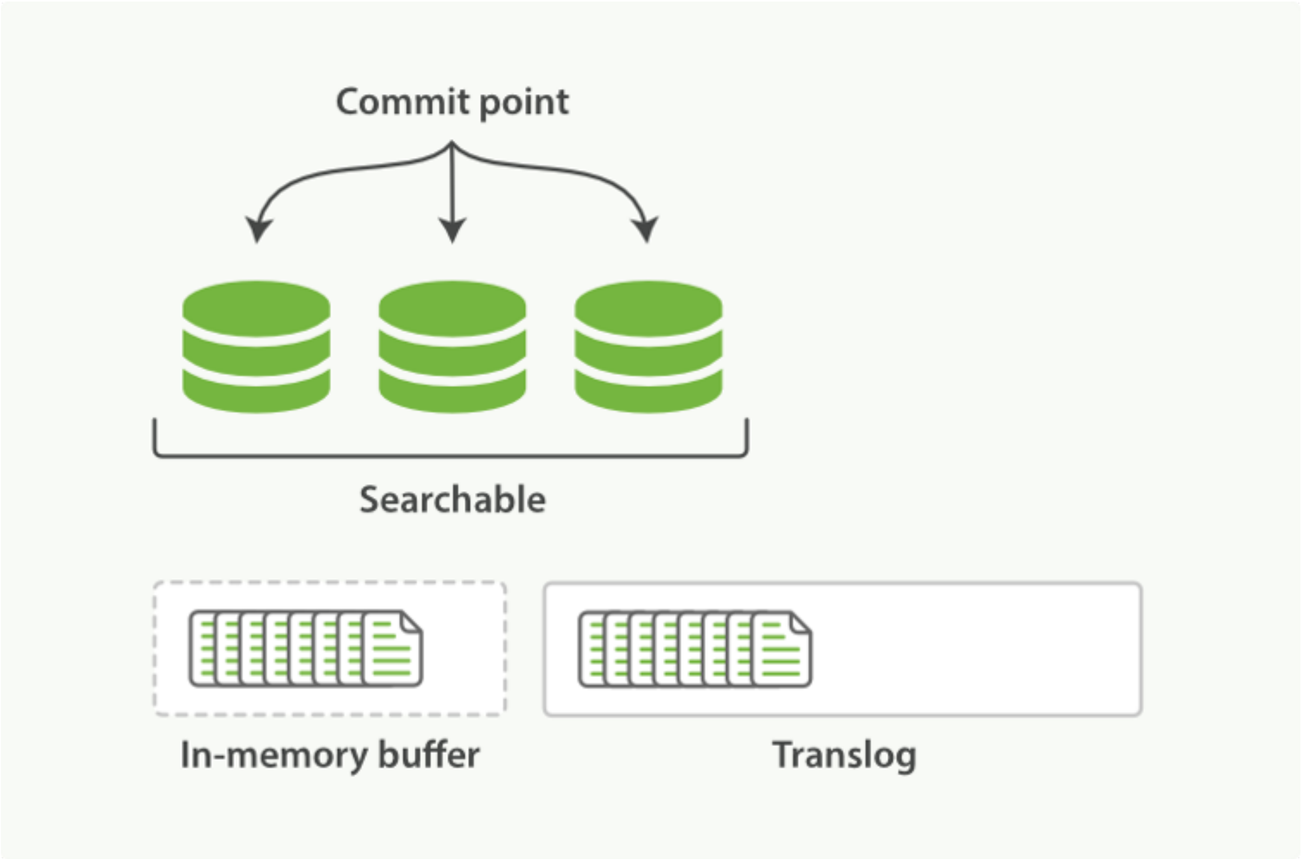
磁盘做filesync，将内核文件系统的数据写入磁盘。

新的segment被打开，内存缓存被清除。



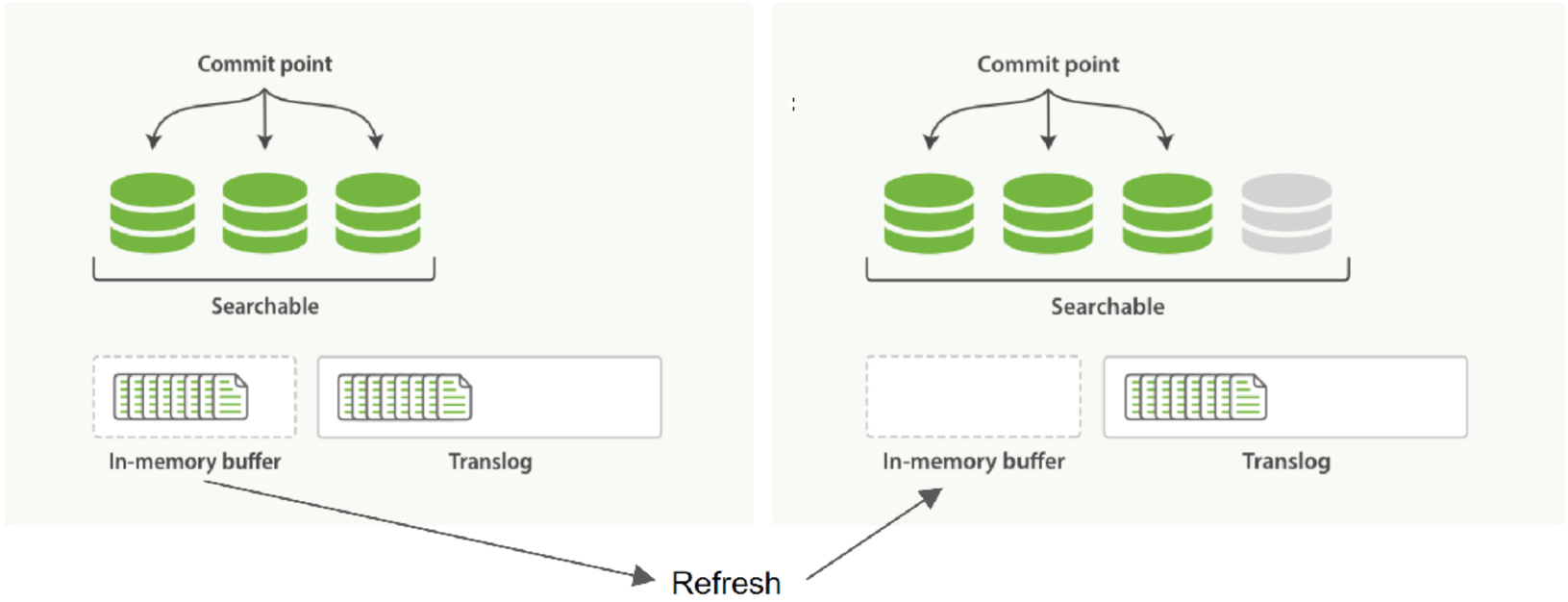
Near Real Time：

写缓存的同时，在写translog。



Near Real Time：

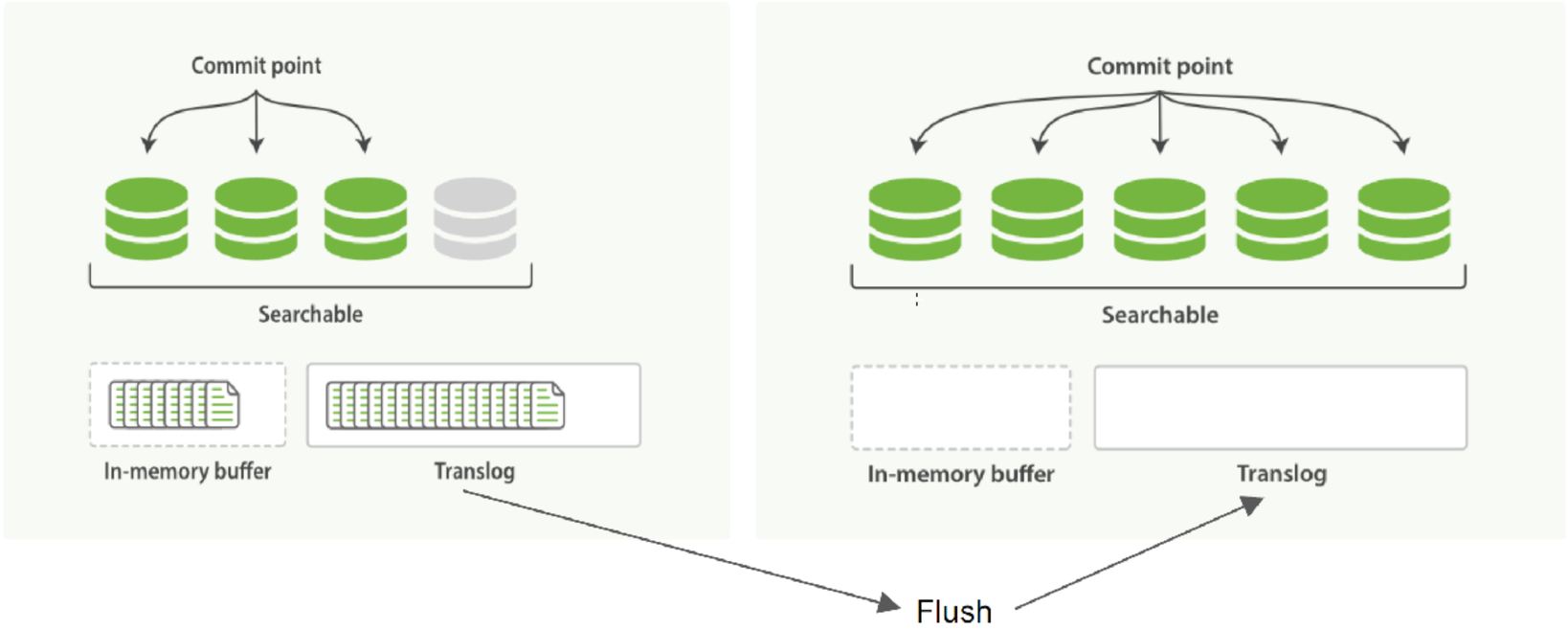
refresh发生的时候，translog日志文件依然保持原样。



https://www.elastic.co/guide/en/elasticsearch/guide/2.x/near-real-time.html

Near Real Time：

Flush过程



https://www.elastic.co/guide/en/elasticsearch/reference/2.4/index-modules-translog.html

基于Elastic Stack分析Vue2.0中文站日志数据

版本的选择

2.x => 2.x
5.x => 5.x

Elasticsearch 2.4.4
Logstash 2.4.1
Kibana 4.6.4
Filebeat 5.2.2

简单的RESTful API:

test/

```
{
    "mappings": {
      "user": {
        "properties": {
          "username": {
            "type": "string",
            "index": "not_analyzed"
          },
          "userage": {
            "type": "string",
            "index": "not_analyzed"
          },
          "sex": {
            "type": "string",
            "index": "not_analyzed"
          }
        }
      }
    }
}
```

简单的RESTful API:

test/user

```
{
    "username": "Tom",
    "userage": 18,
    “sex": "boy"
}
```

简单的RESTful API:

test/user/_search

```
{
  "query": {
    "term": {
      "userage": 18
    }
  }
}
```

简单的RESTful API:
test/user/_search

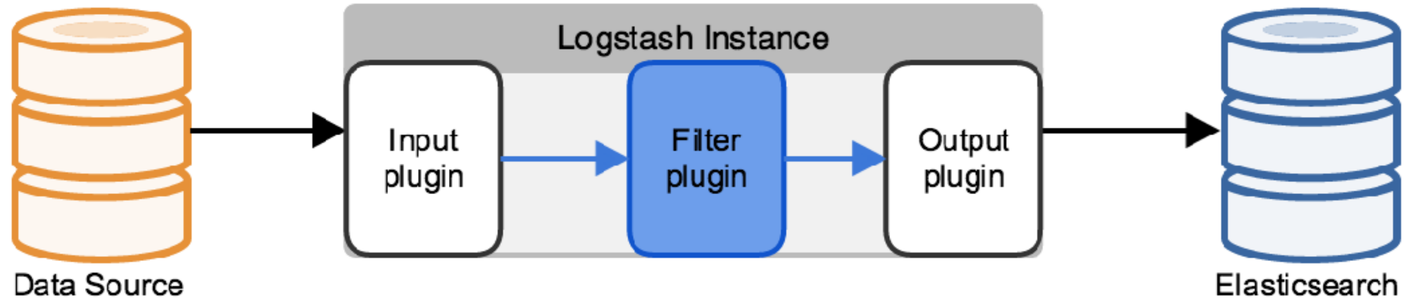
```
{
  "query": {
    "bool": {
      "must": [
        {
          "term": {
            "userage": 18
          }
        },
        {
          "term": {
            "sex": "boy"
          }
        }
      ]
    }
  }
}
```

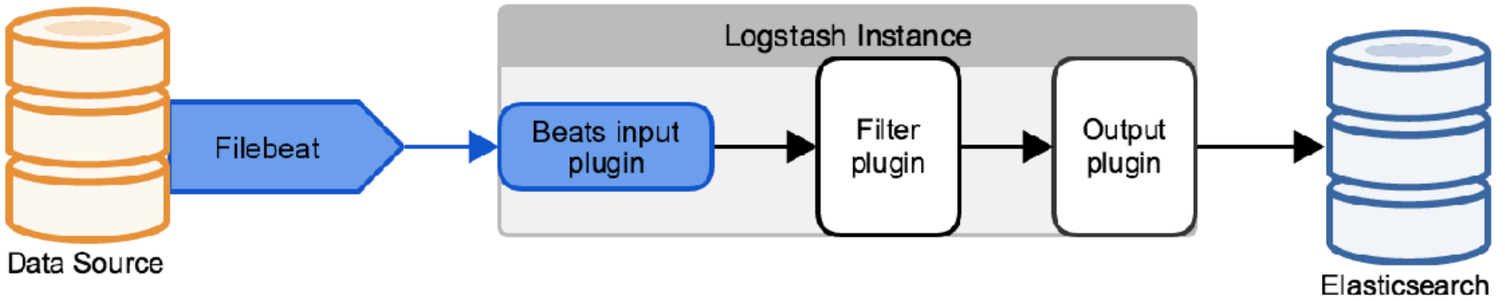
简单的RESTful API:

test/user/_search

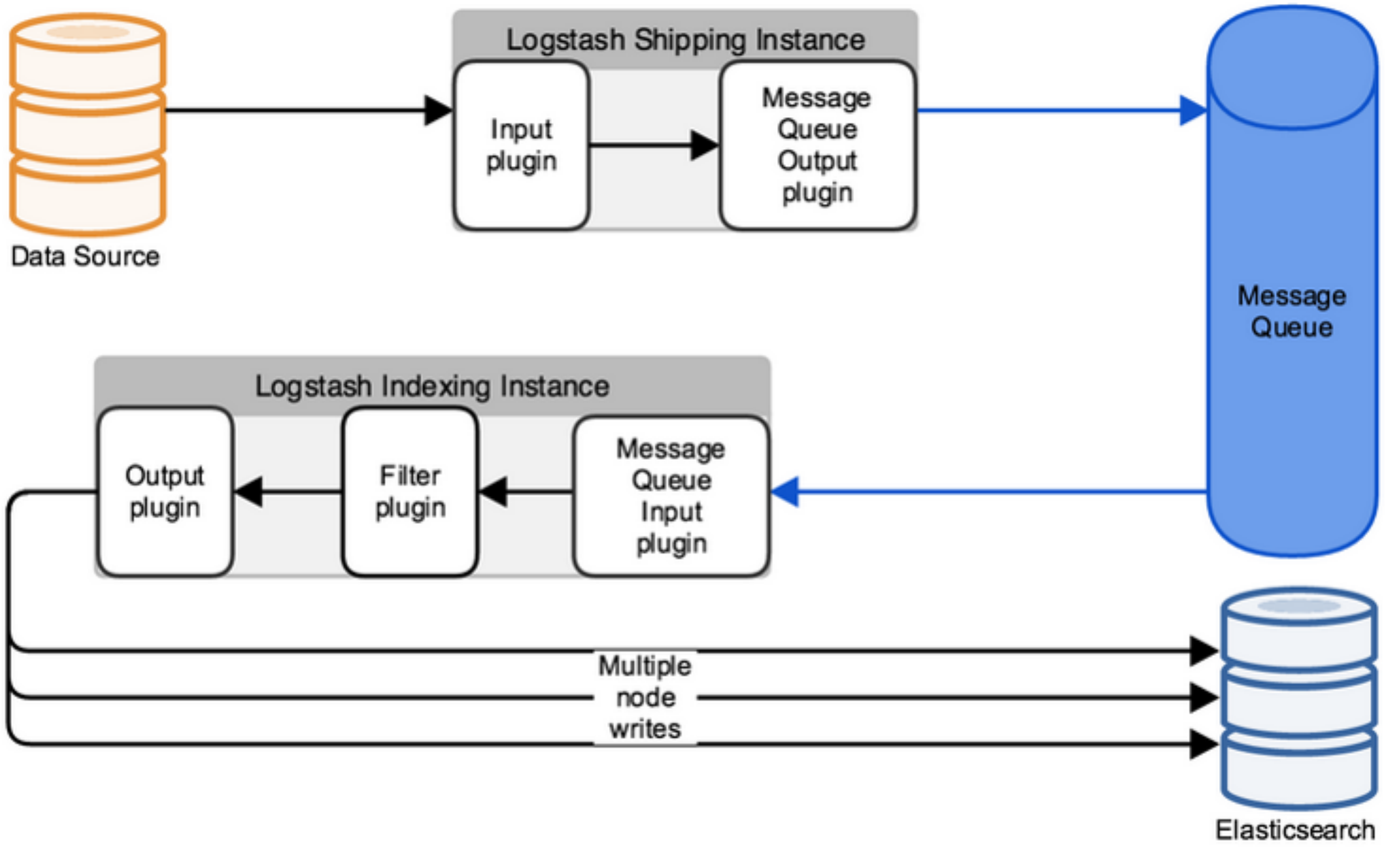
```
{
    "query": {
        "query_string" : {
            "query" : "18"
        }
    }
}
```

Elastic Stack的演进

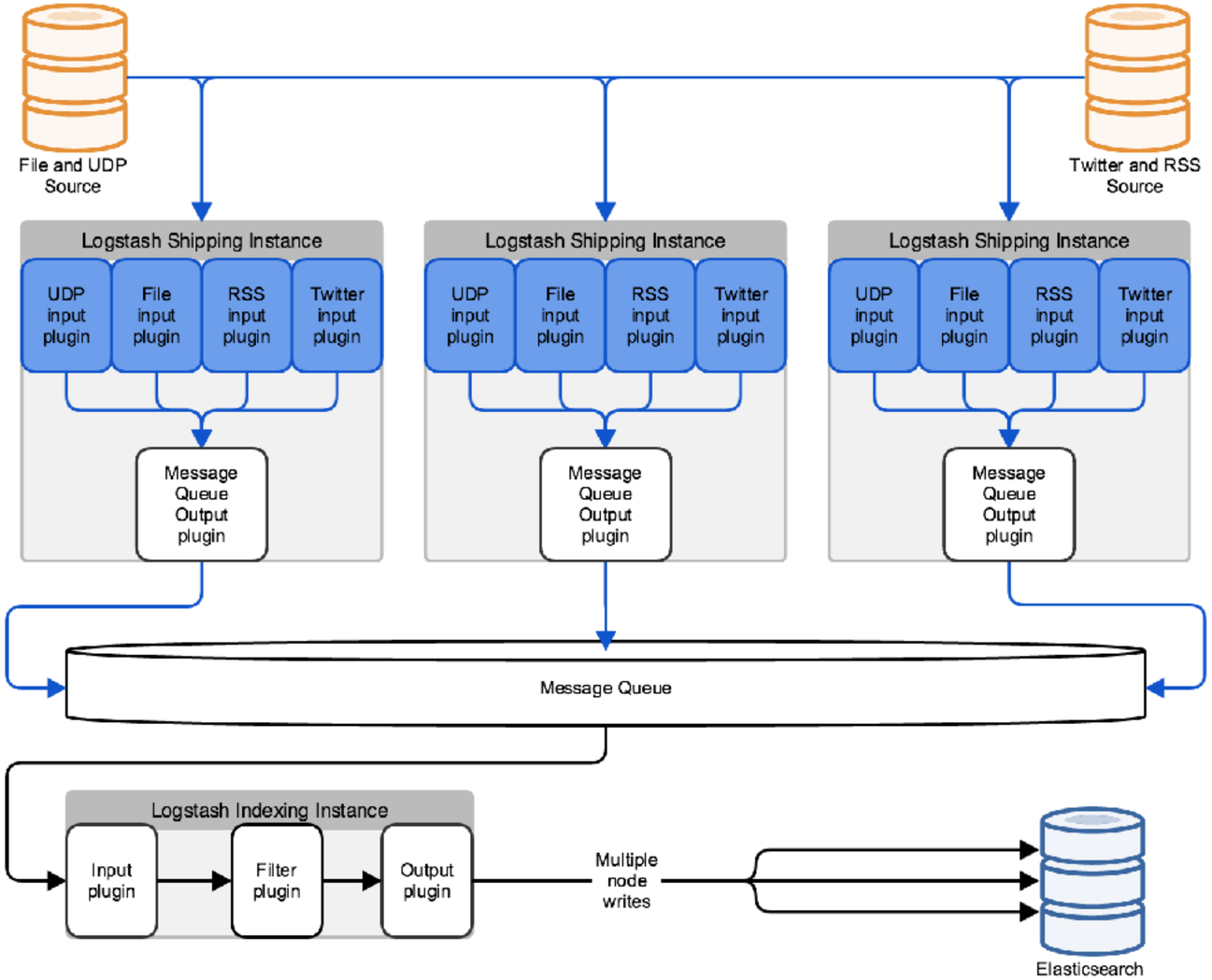




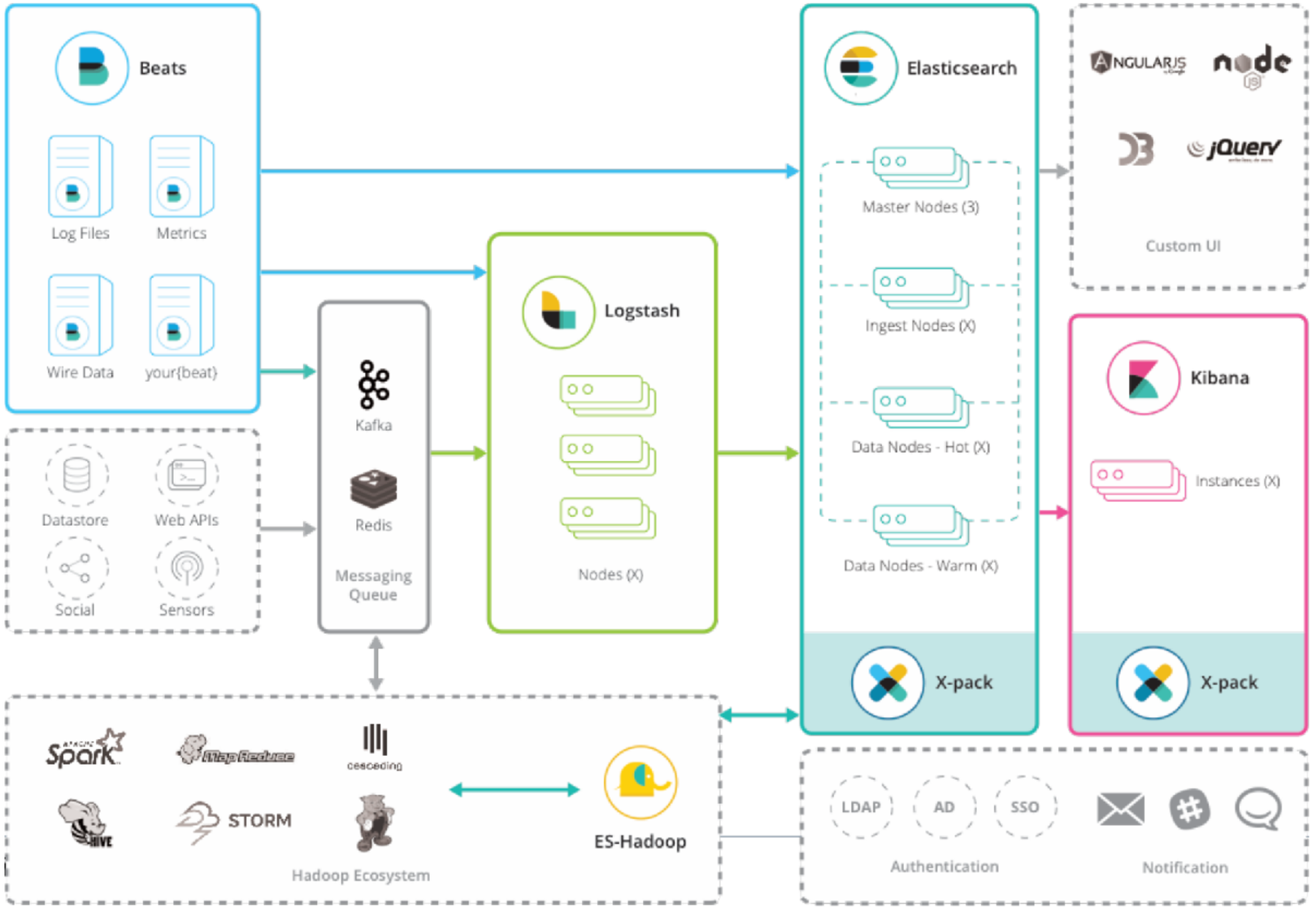
Elastic Stack的演进



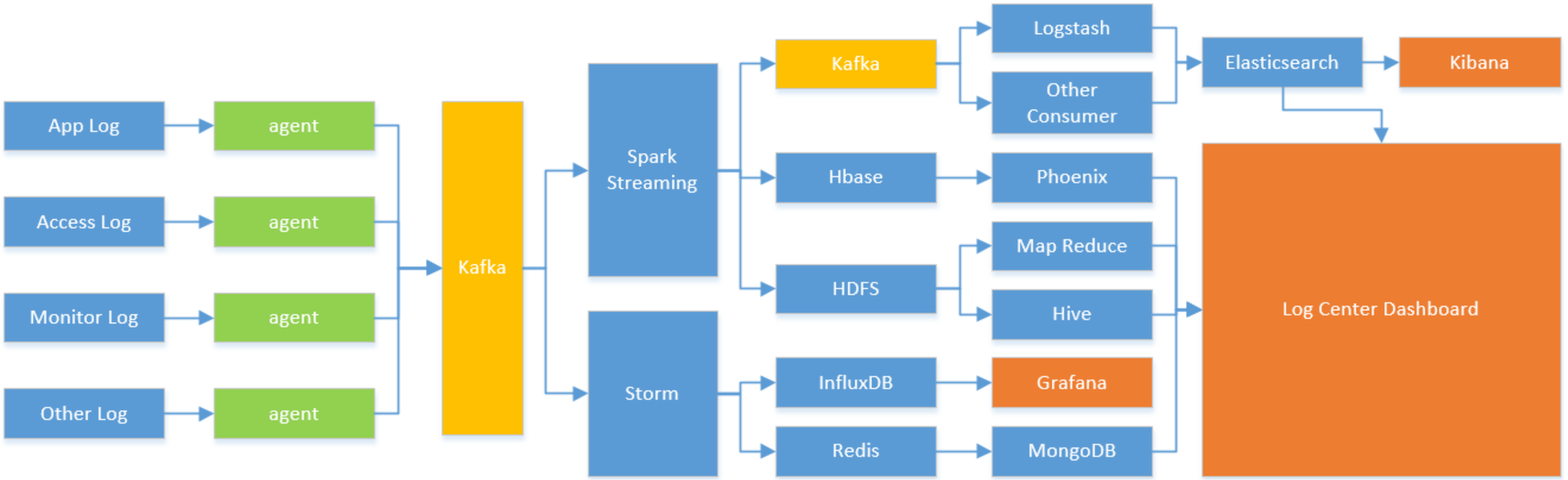
Elastic Stack的演进



Elastic Stack的演进



大日志中心架构



Agent: Kafka-java-producer、Kafka-go-producer、Flume、Filebeat

Flume => HDFS => Map Reduce => MongoDB
Kafka-java-producer、Flume => Kafka => Storm => MongoDB

一些建议：

1.开发 / 测试 / 运维友好

开发图形化界面，封装query。大栅栏（DSL）真的很难写。

不需要分词的字段使用not_analyzed。默认String类型会采用标准分词。

避免使用fielddata cache。缓存会消耗JVM堆内存。

避免size的使用。

GET /_search?size=5&from=100005

每个shard需要返回top 100010给请求结点，合并后，排序后返回其中5条。

自己建index、shard、mapping 避免使用程序自动创建

避免使用Query_String

具体看业务需求，重写入轻查询。

例如，基于应用、IP，查看应用错误日志、HTTP状态码。

5.4版本会有官方的Elasticsearch-SQL支持

一些建议：

2.Kafka 0.8.1-0.8.2 难以逾越的鸿沟

Filebeat 0.8.2～0.10

Logstash 2,4以后不支持低版本

数据格式、权限管理、删除topic

High Level api、Low Level API

Consumers的时候 Partition数量必须大于Consumer数量

3.不要超过32G内存

单机多实例 64G 1实例 128G 2实例

CMS、G1

4.flush、refresh

index.refresh_interval: 30s（默认1s）

index.translog.flush_threshold_period:30m（默认30m）

一些建议：

5.迁移

刷snap快照、scroll&bulk

修改search操作的最大window数，默认为10000

{ "index" : { "max_result_window" : 100000000}}

迁移和数据大小没有绝对的关系，主要是看doc数。

Elasticsearch-dump

自己写Map

6.Elasticsearch集群master节点最好为奇数

防止“脑裂”。

当然最好不要为1，否则没法重启master节点

一些建议：

7.采集端

再造一个轮子，自己写agent。

监听Zookeeper，实现配置热加载。

multiline多行是个问题。一个异常10000多行。

Logstash性能问题，大量消耗CPU和内存，容易僵死。

Filebeat就是Logstash的Shipper。

Filebeat监控文件多了，会卡住。

8.对于容器服务的日志收集

Pod vs 宿主机 资源可控

宿主机起一个Pod专门收集日志

小服务一个Pod起一个Container收集日志

Thanks