ELK 实战

自我介绍 - 霍金光

- 2006/09 – 2010/06 大连理工大学 软件学院（日语强化）
- 2009/07 – 2012/03 梦创信息（大连）有限公司
- 2012/04 – 2015/06 Works Applications（上海、东京）
- 2015/07 – 2017/04 创业公司（上海）
- 2017/05 – 今 LINE China 大连

软件的生命周期?

软件的生命周期

- 问题定义
- 可行性研究
- 需求分析
- 开发阶段 - 概要设计，详细设计，实现，测试
- 维护

软件的生命周期

- 问题定义
- 可行性研究
- 需求分析
- 开发阶段 - 概要设计，详细设计，实现，测试
- **维护**

日志

- 收集－能够采集多种来源的日志数据
- 传输－能够稳定的把日志数据传输到中央系统
- 存储－如何存储日志数据
- 分析－可以支持 UI 分析
- 警告－能够提供错误报告，监控机制

如何查看日志？

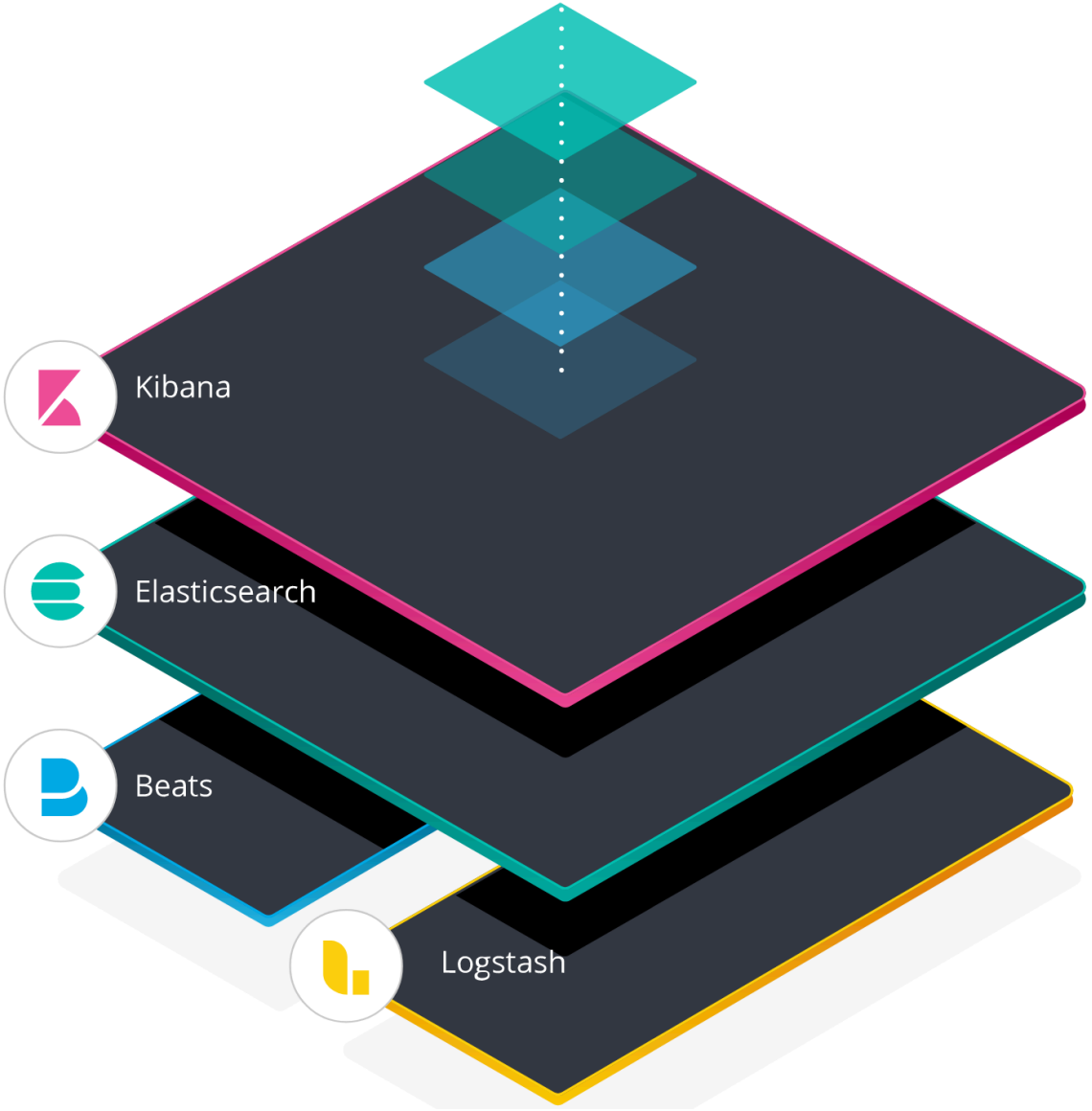
- 开发环境查看IDE控制台
- 发布到客户的服务器上呢?
- Splunk（商业收费）
- Scribe（Facebook）
- 现在公司使用的?

如果让你去设计呢？

该如何收集日志?

ELK 简介

- 所属公司 Elastic
- 官网 elastic.co
- 产品全部开源
- 部分插件如 X-Pack 需付费
- 按节点（Node）数收费



ELK 简介

- Elasticsearch 开源分布式搜索引擎， 提供搜集、分析、存储数据三大功能。它的特点有：分布式， 配置简单， 自动发现， 索引自动分片， 索引副本机制， restful风格接口，自动搜索负载等。
- Logstash （Beats）主要是用来日志的搜集、分析、过滤日志的工具， 支持大量的数据获取方式。一般工作方式为C/S架构， Client端安装在需要收集日志的主机上， server端负责将收到的各节点日志进行过滤、修改等操作在一并发往elasticsearch上去。
- Kibana 是一个开源和免费的工具， Kibana可以为 Logstash 和 ElasticSearch 提供日志分析友好的 Web 界面， 可以帮助汇总、分析和搜索重要数据日志。

ELK 简介 课程安排

- 重点介绍如何使用 ES 原理
- 使用 ES 写入、搜索数据
- 使用 kibana 查询数据
- 简单介绍如何使用 Filebeat 同步 log 到 ES

Elasticsearch

ES

Elasticsearch 特点

- 分布式（可扩展性 - 横向扩展 => 用过的分布式系统？）
- 配置简单
- 自动发现
- 索引自动分片， 索引副本机制
- 自动容错、高可用、易扩展。
- restful风格接口
- 自动搜索负载
- 不支持： 不支持频繁更新、关联查询、事务

Elasticsearch 最大特点？

Elasticsearch 最大特点

快！

Elasticsearch 实例

- 携程运维使用 ES 实例1 （参见网站）
  1. 集群：94个。最小三个节点， 最大：360+节点。 节点： 700+。
  2. 每日增量：1600亿条。 峰值：300W/s。
  3. 总数据量： 2.5万亿， PB数量级。
- 携程业务使用 ES 实例2
  1. 业务场景： 3集群， 每集群6个节点。 单个索引： 最大1000W-2000W。
  2. 关注：ES基础框架， 帮业务部分实现写入、查询、DSL调优。 查询： 3000-4000/s。

Elasticsearch 实例

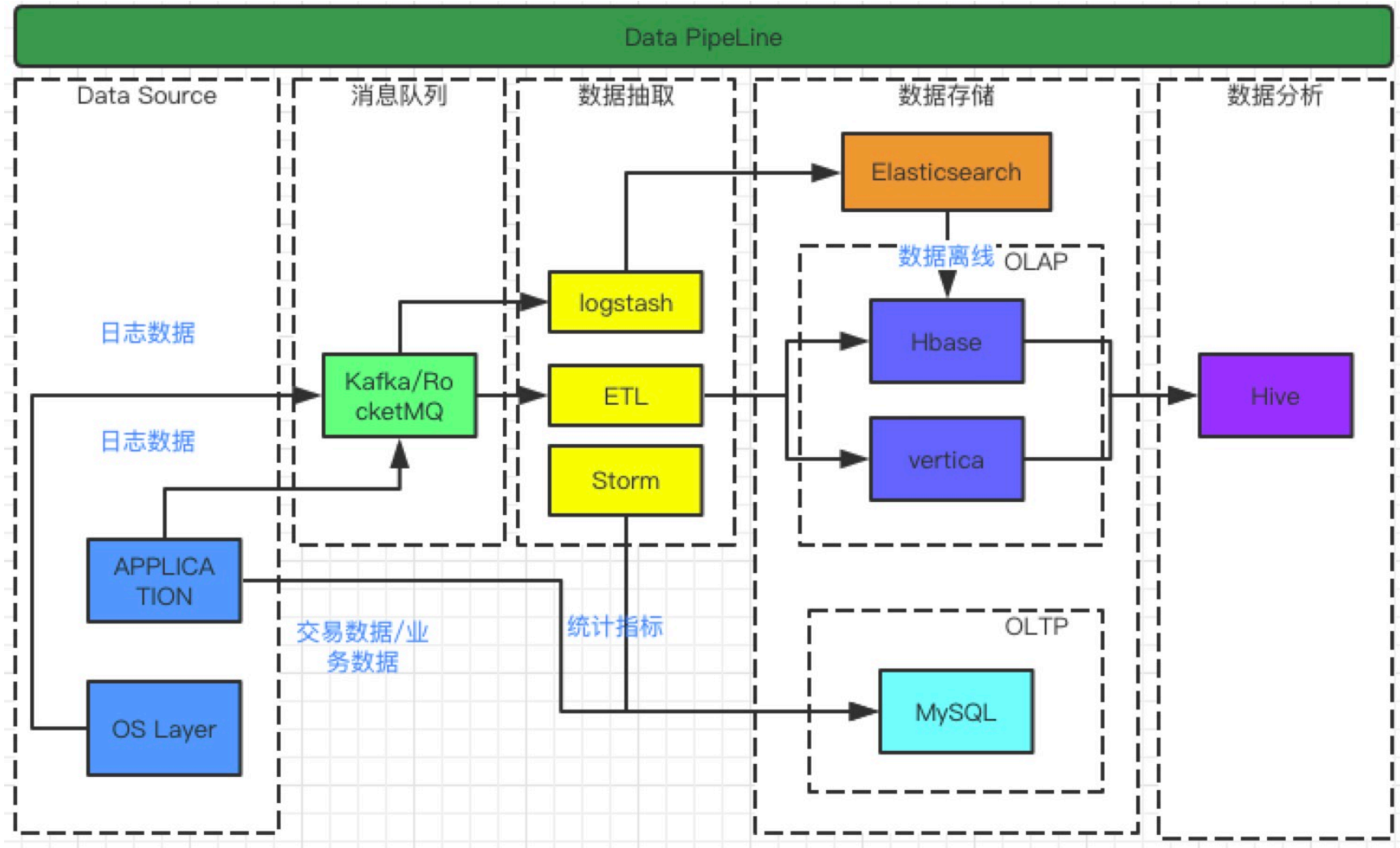
- 2013年初，GitHub抛弃了Solr，采取ElasticSearch 来做PB级的搜索。GitHub使用 ES 搜索20TB的数据，包括13亿文件和1300亿行代码。
- 维基百科：启动以 ES 为基础的核心搜索架构。
- 百度：百度目前广泛使用 ES 作为文本数据分析，采集百度所有服务器上的各类指标数据及用户自定义数据，通过对各种数据进行多维分析展示，辅助定位分析实例异常或业务层面异常。目前覆盖百度内部20多个业务线（包括casio、云分析、网盟、预测、文库、直达号、钱包、风控等），单集群最大100台机器，200个ES节点，每天导入30TB+数据。
- 淘宝等电商网站，新闻网站，OA办公系统等。

问题：如何提高

MySQL / Oracle 吞吐量？

携程使用 ES ：300W/S 写入， 3000 - 4000 次/S 查询（排序）

Elasticsearch 实例



Elastic 中文社区（来源）

Elasticsearch 节点角色划分

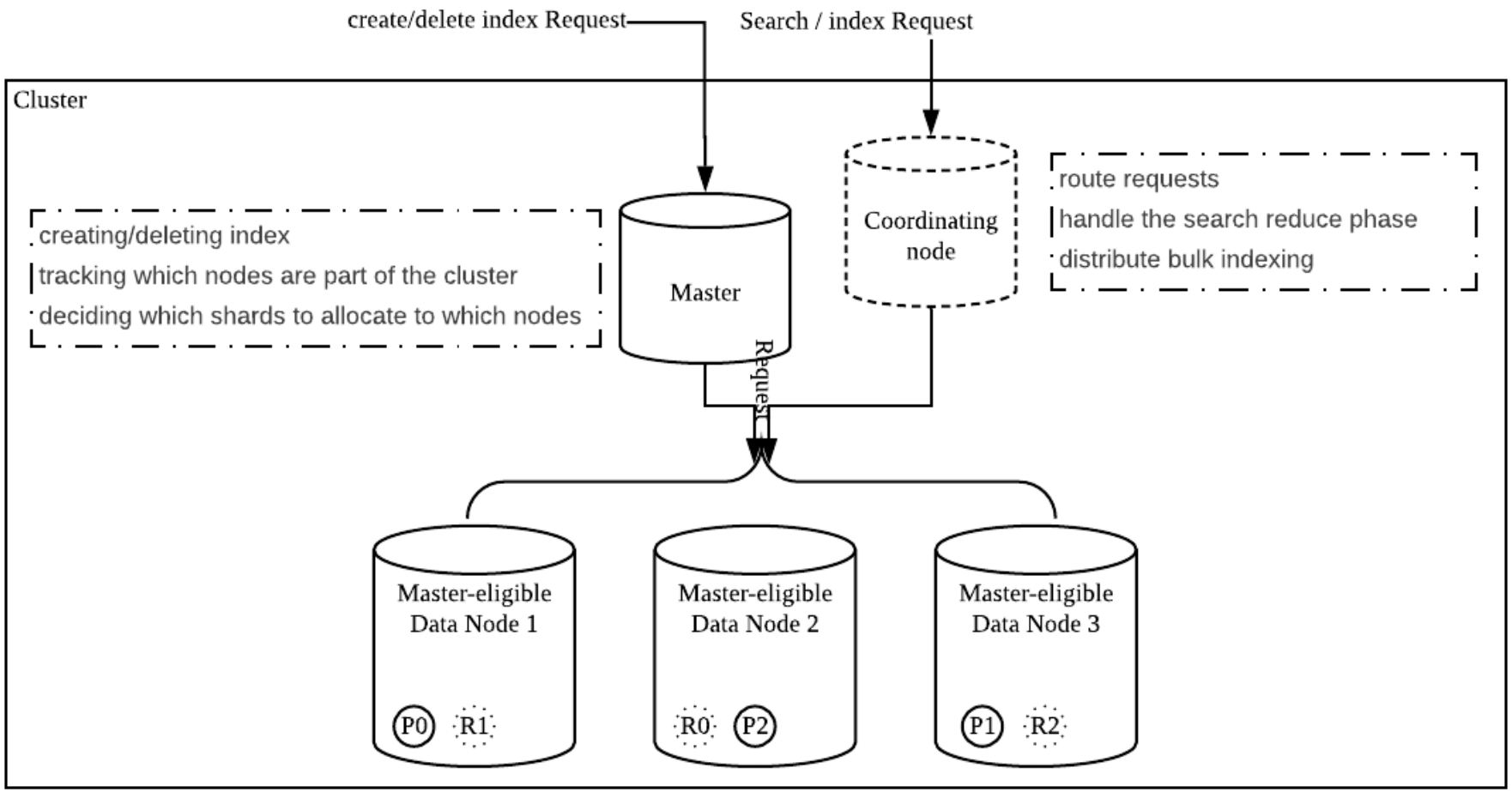
- master：该节点不和应用创建连接，主要用于元数据(metadata)的处理，比如索引的新增、删除、分片分配等，master节点不占用io和cpu，内存使用量一般
- client：该节点和检索应用创建连接、接受检索请求，但其本身不负责存储数据，可当成负载均衡节点，client节点不占用io、cpu、内存
- data：该节点和索引应用创建连接、接受索引请求，该节点真正存储数据，es集群的性能取决于该节点个数（每个节点最优配置情况下），data节点会占用大量的cpu、io、内存

Elasticsearch 基本概念

- 节点（Node）和集群（Cluster）

集群是一个或多个节点（服务器）的集合， 这些节点共同保存整个数据，并在所有节点上提供联合索引和搜索功能。一个集群由一个唯一集群ID确定，并指定一个集群名（默认为“elasticsearch”）。该集群名非常重要，因为节点可以通过这个集群名加入集群，一个节点只能是集群的一部分

Elasticsearch 节点角色划分



本机安装 ES 单点

本机启动 Kibana

配置文件 kibana.yml

访问地址 localhost:5601

本机安装 ES 集群

模拟三节点

本机安装 ES 集群

- 配置文件 elasticsearch.yml
- 因为三个节点，需要拷贝三份配置文件，并修改各个文件当中的 node.name/data/logs/port 等信息
- cluster.name: dlsp
- node.name: node1
- path.data: D://data/node1
- path.logs: D://logs/node1

本机安装 ES 集群

- 配置文件 elasticsearch.yml

- network.host: 127.0.0.1
- http.port: 9201
- transport.tcp.port: 9202
- discovery.zen.ping.unicast.hosts: ["localhost:9202", "localhost:9302", “localhost:9402"]
- discovery.zen.minimum_master_nodes: 2

本机安装 ES 集群

- Linux 上的启动脚本

```
#!/bin/sh

LOG_DIR=/var/logs/es1
ES_PATH_CONF=/conf/es1
ES_PATH=/espath

export ES_NETWORK_HOST=127.0.0.1
export ES_PATH_CONF

${ES_PATH}/elasticsearch/bin/elasticsearch -d -p ${LOG_DIR}/elasticsearch.pid
```

本机安装 ES 集群

- 确认 log 信息无误
- 查看 ps -ef | grep elasticsearch 进程存在
- 浏览器访问 localhost:9021 看是否可访问

ES 基本概念

Elasticsearch 基本概念

- Index（索引）— 数据库表

索引(index)类似于关系型数据库里的“数据库” — 它是我们存储和索引关联数据的地方。索引名称必须是全部小写，不能以下划线开头，不能包含逗号。

- Type（类型）

在索引中，我们可以定义一个或多个类型。类型是索引的逻辑类别/分区，其语义完全由开发者决定。通常，为具有一组公共字段的文档定义类型。

Deprecated from ES6.0，Elasticsearch 7.0 开始已经取消 type 概念。

Elasticsearch 基本概念

- Document（文档）— 数据库表中的一行数据

文档是可索引信息的基本单元， 以JSON表示。我们可以把文档理解为数据库文档中的行列数据。在索引/类型中， 您可以存储任意数量的文档。每个文档必备属性 _index, _type, _id, 检索时必须指定 _index， _type

Elasticsearch 基本概念

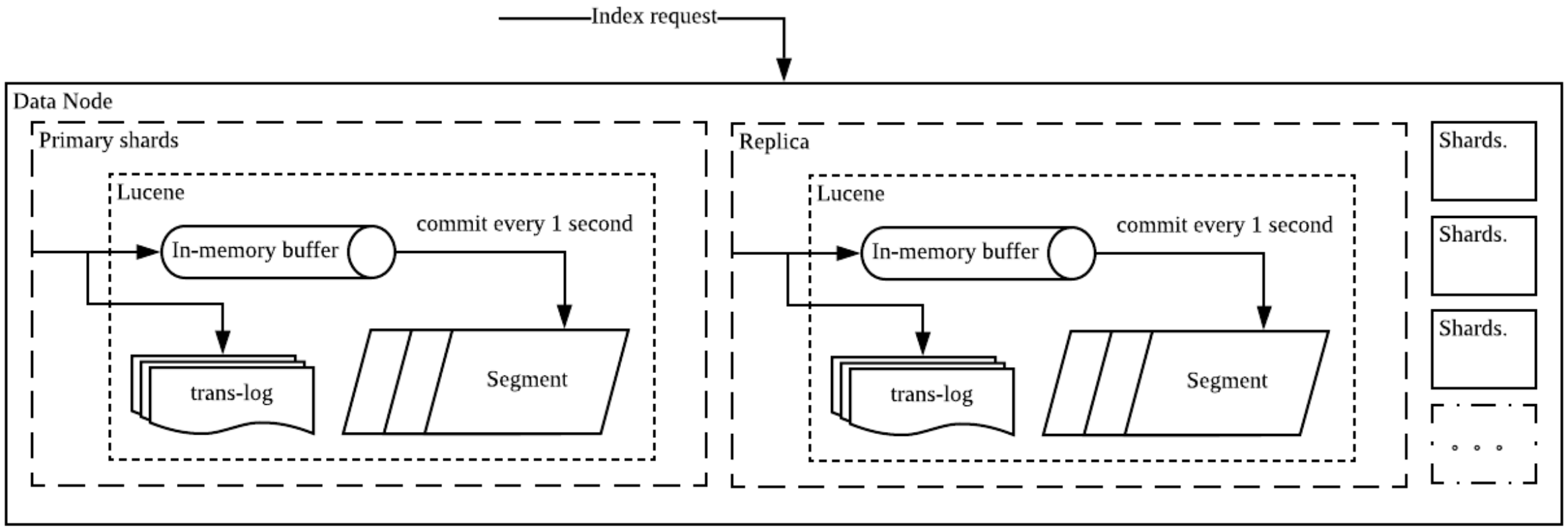
- Mapping（映射）— 数据库表中每一列的定义

模式映射（schema mapping，或简称映射）用于定义索引结构。 类型：text, keywrod, date, long, double, boolean, ip, object, nested, geo_point (地理坐标）等类型。

- Field（字段）— 数据库表的列

ElasticSearch里的最小单元 相当于数据的某一列， 类似于json里一个键。

Elasticsearch 节点角色划分



创建 ES Index

```
PUT twitter
{
    "settings" : {
        "number_of_shards" : 1,
        "number_of_replicas" : 2
    },
    "mappings" : {
        "_doc" : {
            "properties" : {
                "user" : { "type" : "text" },
                "post_date" : { "type" : "date" },
                "message" : { "type" : "text" },
            }
        }
    }
}
```

ES Doc

生成 ES Document

指定 ID

```
PUT twitter/_doc/1
{
    "user" : "kimchy",
    "post_date" : "2009-11-15T14:12:12",
    "message" : "trying out Elasticsearch"
}
```

自动生成 ID

```
POST twitter/_doc/
{
    "user" : "kimchy",
    "post_date" : "2009-11-15T14:12:12",
    "message" : "trying out Elasticsearch"
}
```

检索 ES Document

指定 ID

```
curl -X GET “localhost:9200/twitter/_doc/0"
GET twitter/_doc/0
```

```
GET /_search
{
    "query": {
        "match" : {
            "message" : "this is a test"
        }
    }
}
```

Kibana 查询语法

age:(>=10 AND <20)

bytes: >1000

count:[1 TO 5]

count:[10 TO *]

date:[2012-01-01 TO 2012-12-31]

date:{* TO 2012-01-01}

response:200

response:200 **and** (extension:php **or** extension:css)

response:200 **and** extension:php **or** extension:css

response:200 **and not** (extension:php **or** extension:css)

response:(200 **or** 404)

Java 操作 ES

Java 操作 ES

- pom.xml 中引入 ES 操作所需依赖

```
<!-- elasticsearch -->
<dependency>
  <groupId>org.elasticsearch.client</groupId>
  <artifactId>transport</artifactId>
  <version>6.8.0</version>
</dependency>
```

search-service 代码演示

ES 练习

- 当增加商品时，同时将商品信息更新到 ES 中
- 当更新商品信息时，同时更新信息到 ES 中
- 为 web 端提供全文检索商品的接口，并能够进行分页

Filebeats 本机部署

- 在每一台需要收集日志的服务器上都需要进行安装
- 修改配置信息
- - type.enabled : true
- paths:
- - /var/log/*.log
- #- c:\programdata\elasticsearch\logs*

谢谢！

Wechat: huohuo5234