



# MongoDB

## 高级应用场景及技术特点

---

锦木信息  
唐峰

# 哪些场景适合MongoDB?

RDBMS, MongoDB?

NoSQL, MongoDB?

# 核心特点

## Document Model

最佳的数据管理方式



## Run Anywhere

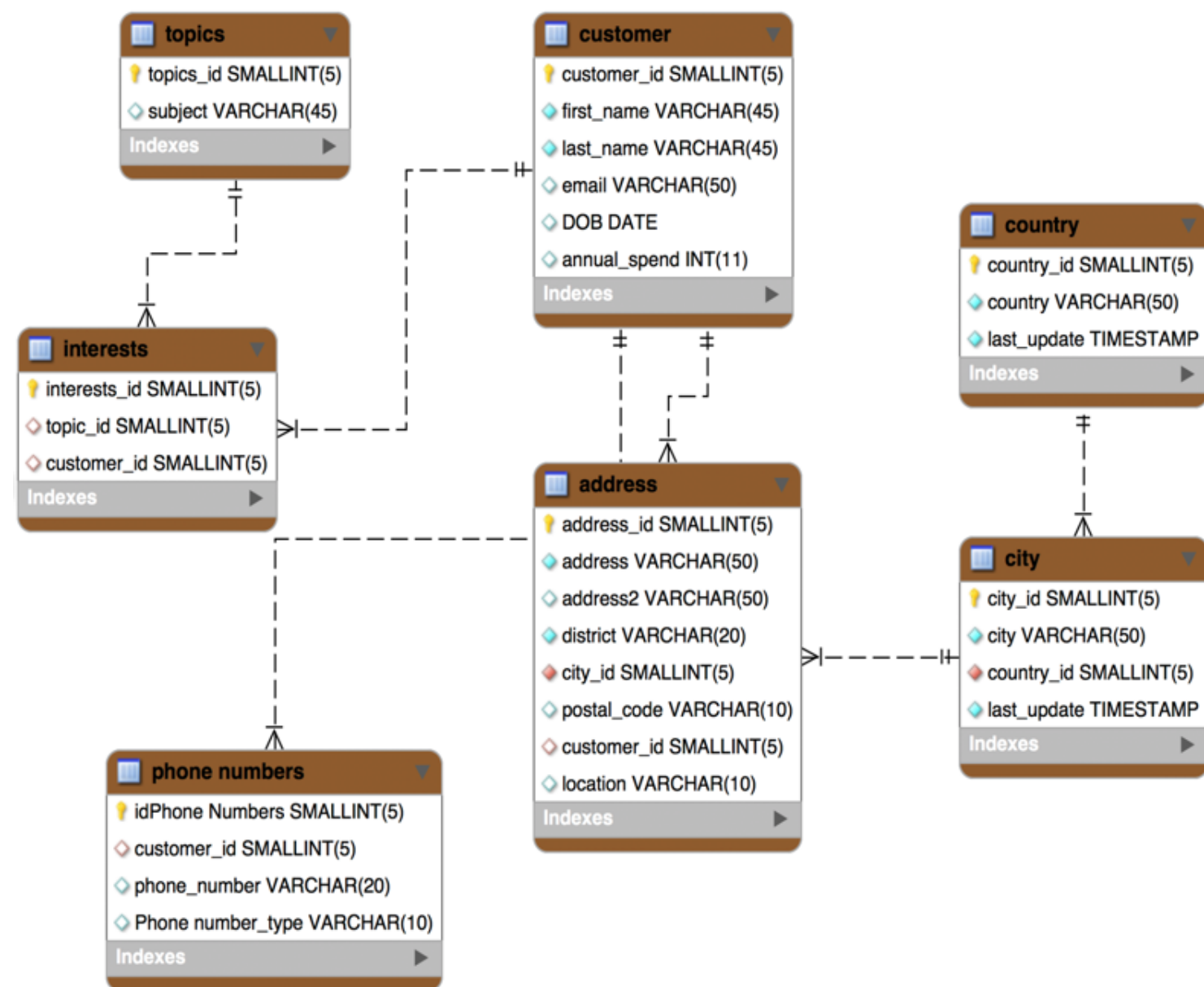
自由得运行于任意平台

## Distributed System

智能得将数据放在需要的地方

# 文档模型

模型易用



```
{
  _id : 123456 ,
  first_name : "Mark" ,
  last_name : "Smith",
  city : "San Francisco" ,
  phones: [
    {
      number : "1-212-777-1212",
      dnc : true,
      type : "home"
    },
    {
      number : "1-212-777-1213",
      type : "cell"
    }
  ]
}
```

{ JSON }

# 举例

金融行业资产管理案例

## 1. 基础客户信息

① 客户名称 ② 客户地址 ③ 组织机构代码 ④ 营业执照号 ⑤ 社会统一信用代码

## 2. 客户行业信息

矿业行业：



- ① 煤炭生产许可证
- ② 矿井储量 (万吨)
- ③ 矿井位置 (经纬度)
- ④ 百万吨死亡率(%)

地产行业：



- ① 房地产资质证书
- ② 年开工面积 (平米)
- ③ 年销售面积(平米)

电力行业：



- ① 装机容量 (兆瓦)
- ② 年发电量 (兆瓦)

# 关系模型


传统关系型设计模式

## 传统方案1：宽表

客户ID	名称	地址	组织机构代码	营业执照	矿井储量	开工面积	.....	发电量
100001	多多的电力	XXXX	XXXX	XXXX				XXXX
100002	大大的地产	XXXX	XXXX	XXXX		XXXX		

## 传统方案2：主从表

客户ID	名称	地址
100001	深深的煤业	XXXX



客户ID	METRIC_NAME	METRIC_VALUE
100001	生产许可证	mtsc017
100001	矿井储量	2000000
100001	矿井位置	138.2031 -124.4904
100002	百万吨死亡率	2%

# 文档模型

文档类型设计模式

{ JSON }

## 文档1

```
{
  name: "深深的矿业",
  orgId: "70107890X",
  industryInfo: {
    type: "采矿"
    coalCertificate: "mtsc017",
    location: [ 38.2031, -120.4904 ],
    mineReserves: 2000000,
    deathsByMT: 0.02
  }
}
```

## 文档2

```
{
  name: "大大的地产",
  orgId: "565425429",
  industryInfo: {
    type: "地产",
    estateCertificate: "fdczz0000001",
    areaPerYear : 897600.00,
    salesAreaPerYear : 300000.00
  }
}
```

储量小于2000万吨:

```
db.collection.find( { industryInfo. mineReserves : { $lt: 2000000 } } )
```

# 事务支持

4.0版本支持多文档事务

```
with client.start_session() as s:  
    s.start_transaction()  
    try:  
        collection.insert_one(doc1, session=s)  
        collection.insert_one(doc2, session=s)  
        s.commit_transaction()  
    except Exception:  
        s.abort_transaction()
```

## 和关系数据库的事务操作一样

- 语法基本一样
- 任何应用都可以使用
- 一个表或多个表中的多个文档都适用

## ACID 保证

- 要么全成功，要么全失败

## 版本要求

- MongoDB 4.0, 支持复制集，已经发布
- MongoDB 4.2: 支持分片，预计19年发布

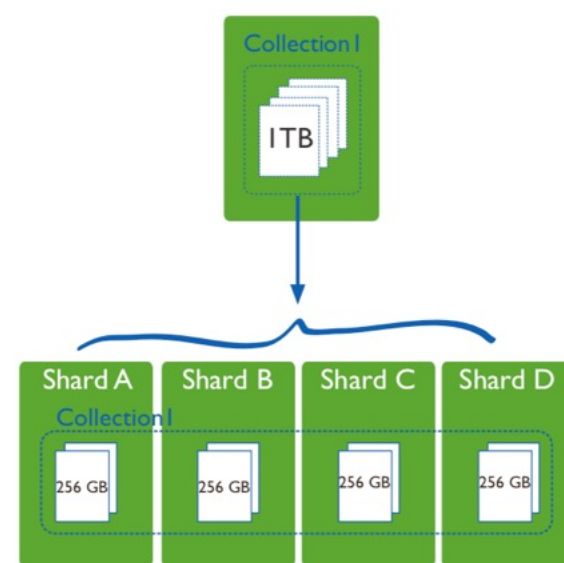


# 文档模型

性能强大、功能丰富

## 性能强大

文档模型、自动分片  
PB级数据库支持



## 功能丰富

地理位置查询



图查询

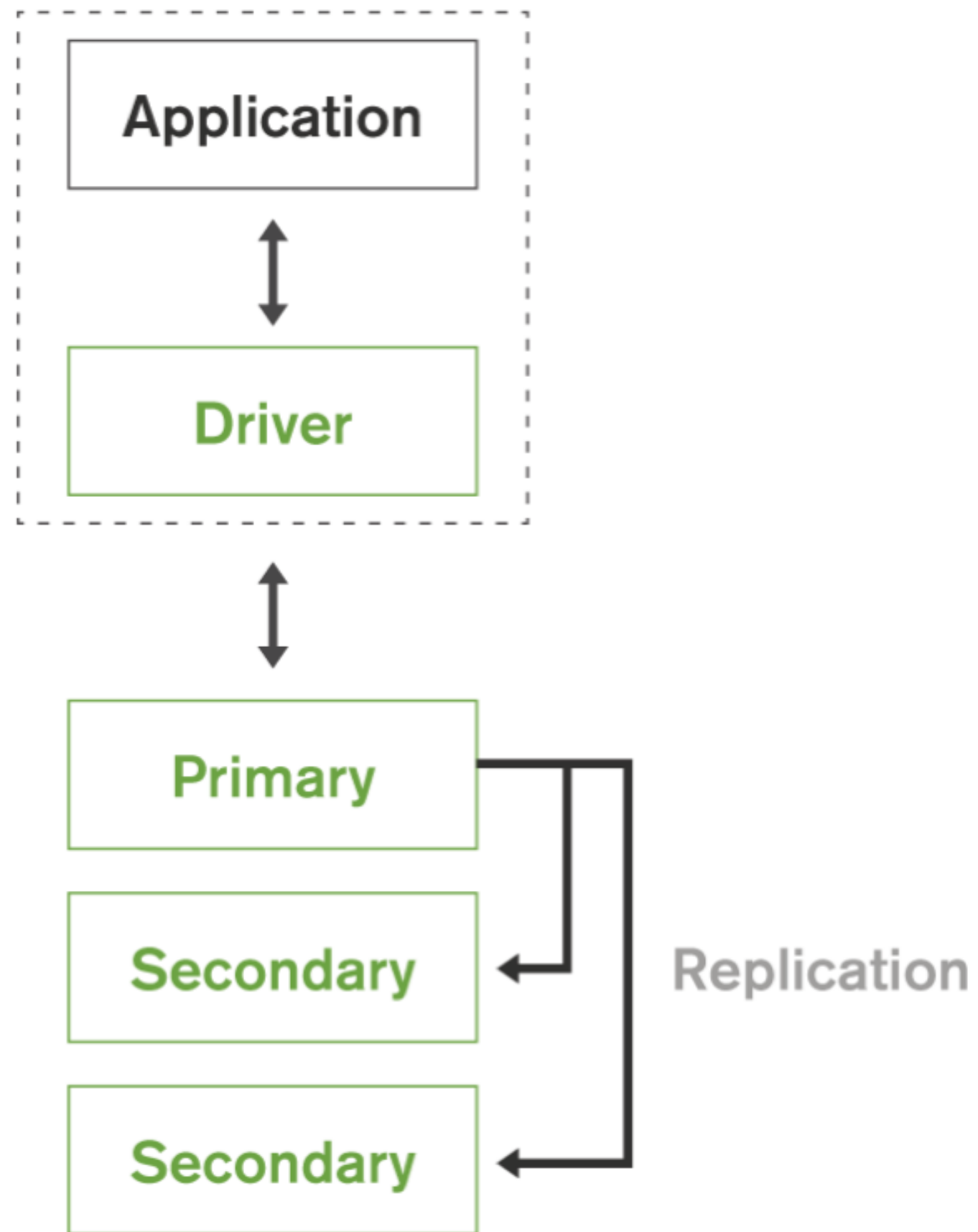


列表查询



# 分布式扩展

复制集 & 分片



复制集支持2-50个节点

自动恢复

关键能力：

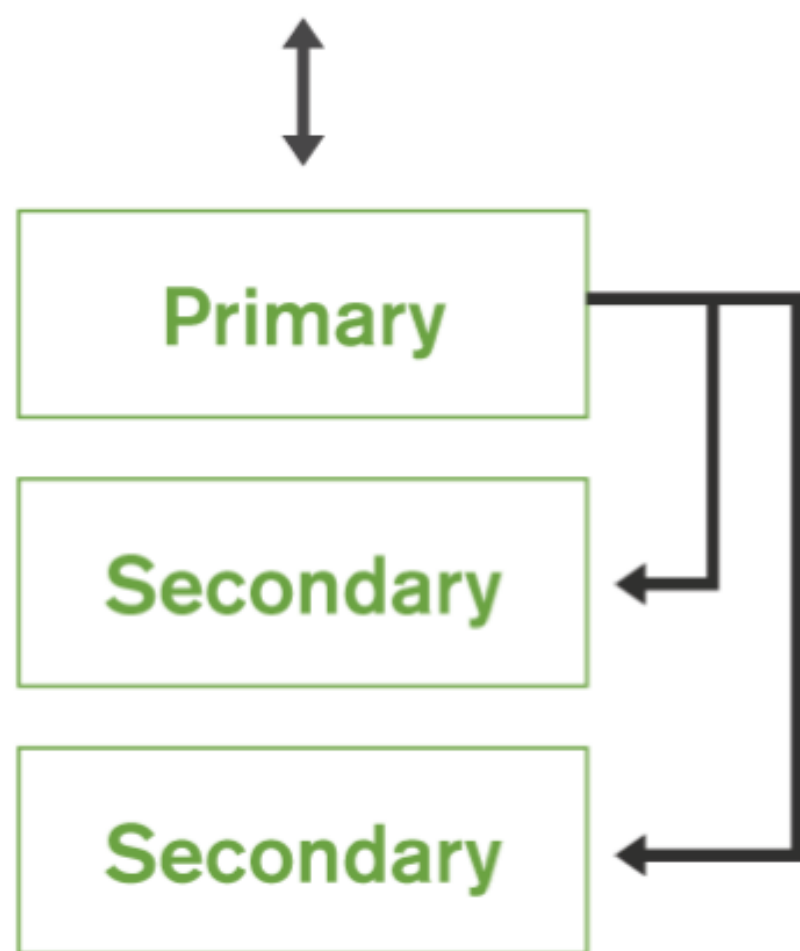
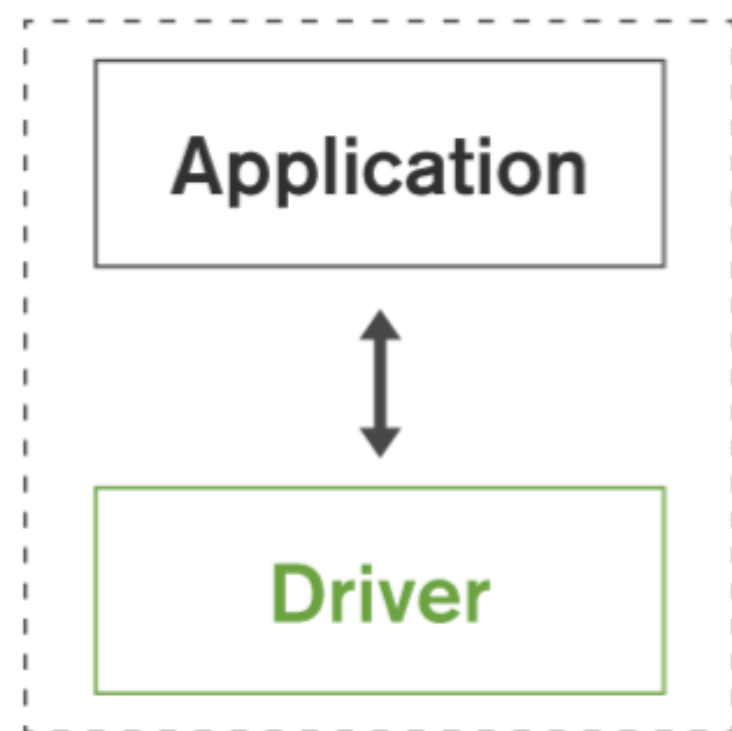
- 高可用
- 容灾
- 维护升级

任务隔离：联机与分析作业隔离

支持不同存储引擎（WireTiger、In-memory）

# 分布式扩展

复制集 & 分片



Replication



Horizontally Scalable



分片策略：范围、哈希、标签

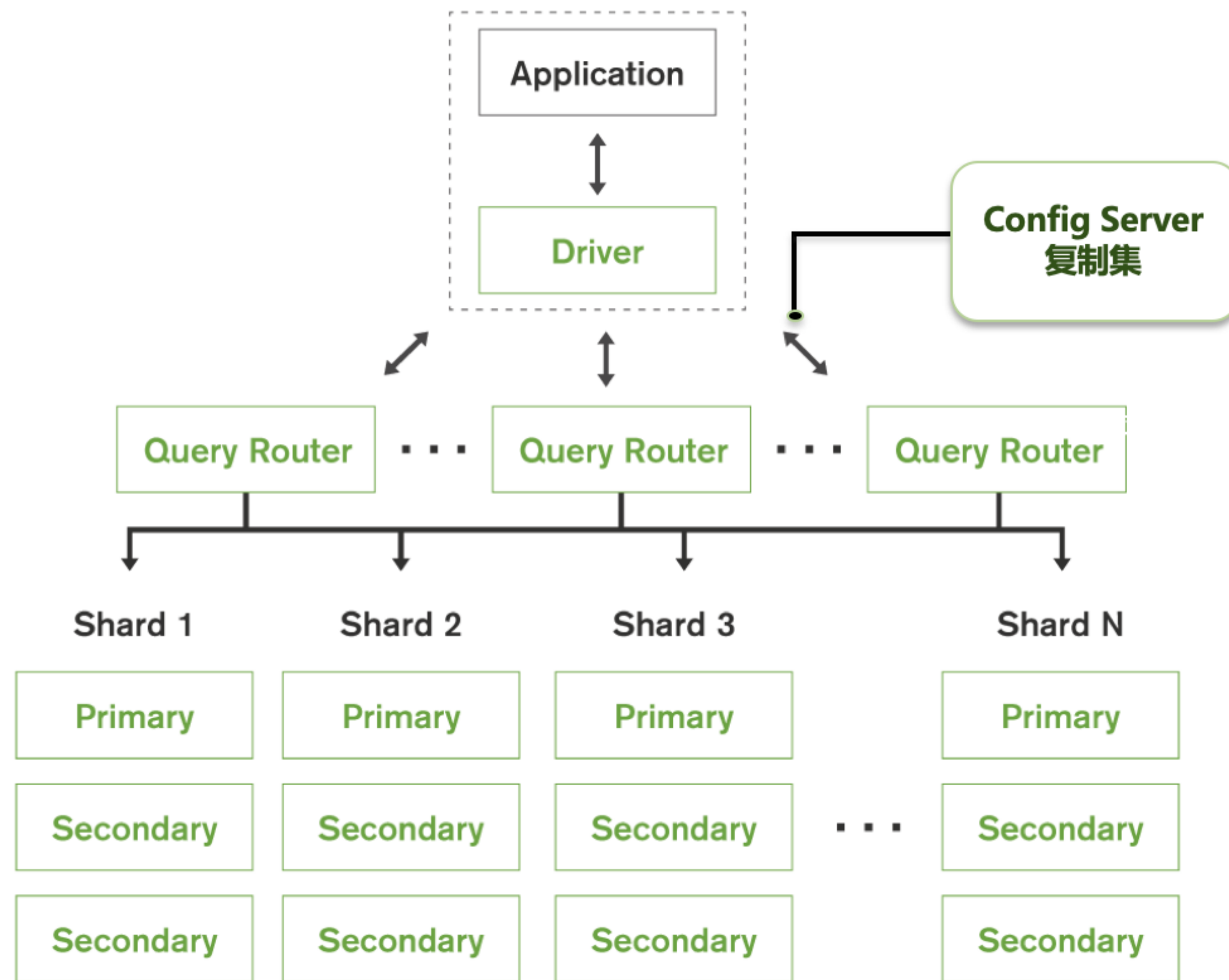
弹性扩展与收缩

数据自动均衡

应用完全透明

# 分布式扩展

复制集 & 分片



# 任意部署

本地、云端、移动端

桌面



本地服务器



私有云



混合云



公有云



数据库云服务



从桌面到云中均可部署运行



# 核心特点



## 最佳的数据管理方式

- 文档模型易用
- 模型变更零代价
- 性能表现优异
- 查询功能丰富

## 智能的将数据放在需要的地方

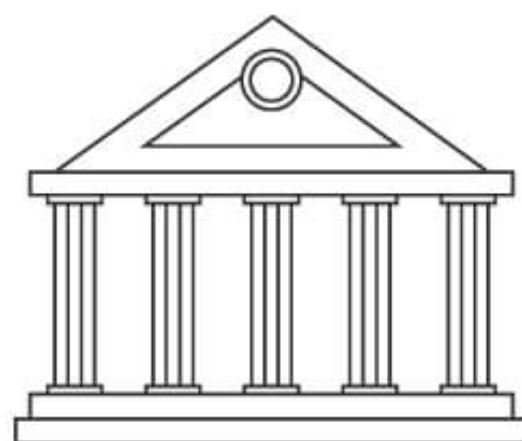
- 高可用复制集
- 分片弹性扩展
- 一库多用
- 可插拔引擎
- 就近数据读写

## 自由的运行于任意平台

- 服务器
- 云 (Atlas)
- 容器
- 移动端

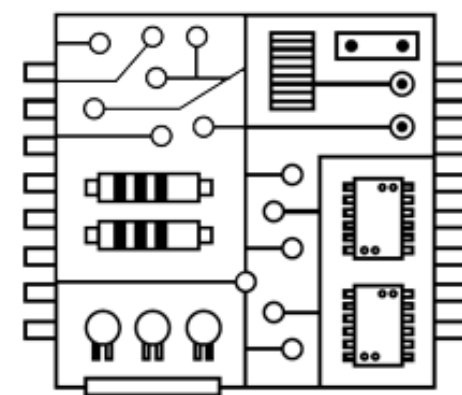
# 统一视图

Single View



金融

分析用户资产、信用、盈利、投资数据，控制风险，提供增值业务。



互联网

根据用户行为与浏览内容，实时推荐感兴趣的商品、内容。



零售

全渠道用户信息清洗合并，生成标签信息，个性化活动推送与服务。

# 统一视图

Single View

功能要求	MongoDB
<b>数据模型</b> <ul style="list-style-type: none"><li>- 数据来源于多个系统，数据格式差异大</li><li>- 随着业务发展，需要动态调整数据模式</li></ul>	<b>数据模型</b> <ul style="list-style-type: none"><li>- 文档模型支持丰富的数据结构（Attribute Pattern）</li><li>- 灵活模式无变更成本</li></ul>
<b>查询</b> <ul style="list-style-type: none"><li>- 查询条件、类型多变</li><li>- 数据算法分析&amp;快速可视化</li></ul>	<b>查询</b> <ul style="list-style-type: none"><li>- 丰富的查询、索引、聚合</li><li>- 数据可视化：MongoDB Charts &amp; BI Connector</li><li>- AI: Spark connector、R &amp; Python Drivers</li></ul>
<b>性能与安全</b> <ul style="list-style-type: none"><li>- 新数据源持续集成，数据量增长迅速</li><li>- 敏感数据安全与保护</li></ul>	<b>性能与安全</b> <ul style="list-style-type: none"><li>- 自动扩展分片架构</li><li>- 鉴权、权限、加密、审计</li><li>- 基于Zone的分区</li></ul>



# 物联网 & 时序数据

IoT & Time-series Data



## IoT

可穿戴设备、车联网、等传感器数据，包含了大量数据价值。



## 交易数据

行情数据、交易行为分析，提供相似K线，智能投顾等增值业务。



## 日志

基础架构及应用系统产生的日志分析，提升运维效率降低故障率。

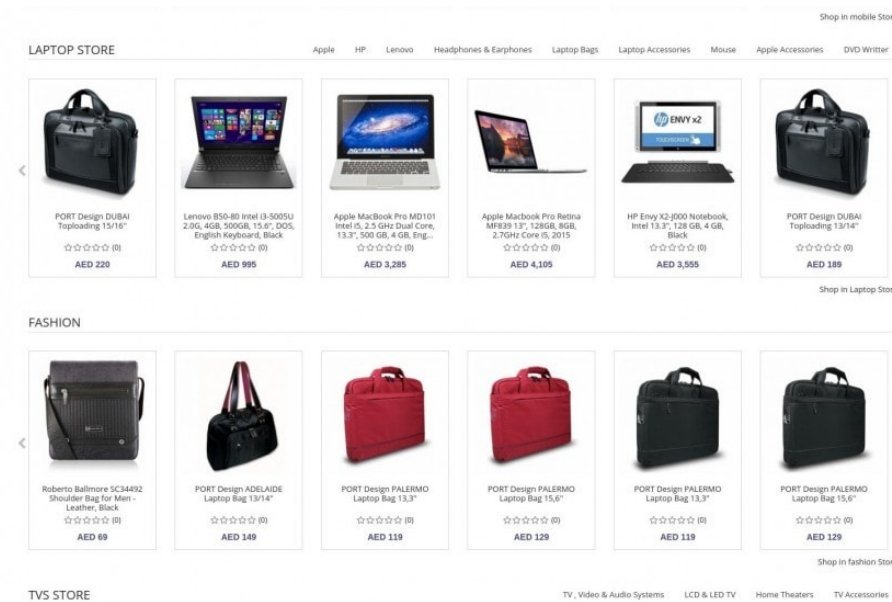
# 物联网 & 时序数据

IoT & Time-series Data

功能要求	MongoDB
<b>数据模型</b> <ul style="list-style-type: none"><li>- 时序类数据，大并发写入</li><li>- 数据结构随着软/硬件版本的更新发生变化</li></ul>	<b>数据模型</b> <ul style="list-style-type: none"><li>- 文档模型支持丰富的数据结构 (Bucket Pattern)</li><li>- 灵活模式无变更成本</li></ul>
<b>扩展与性能</b> <ul style="list-style-type: none"><li>- 数据量大，随设备全球分布</li><li>- 持续在线</li><li>- 读写实时性要求高</li></ul>	<b>扩展性</b> <ul style="list-style-type: none"><li>- 分布式架构auto-sharding</li><li>- rang hash zone的分片策略</li></ul>
<b>计算分析</b> <ul style="list-style-type: none"><li>- 实时写入与实时分析</li><li>- 大数据量计算分析</li></ul>	<b>计算分析</b> <ul style="list-style-type: none"><li>- 一库多用</li><li>- 丰富的查询、索引、聚合</li><li>- 数据可视化: MongoDB Charts &amp; BI Connector</li><li>- AI: Spark connector、R &amp; Python Drivers</li></ul>

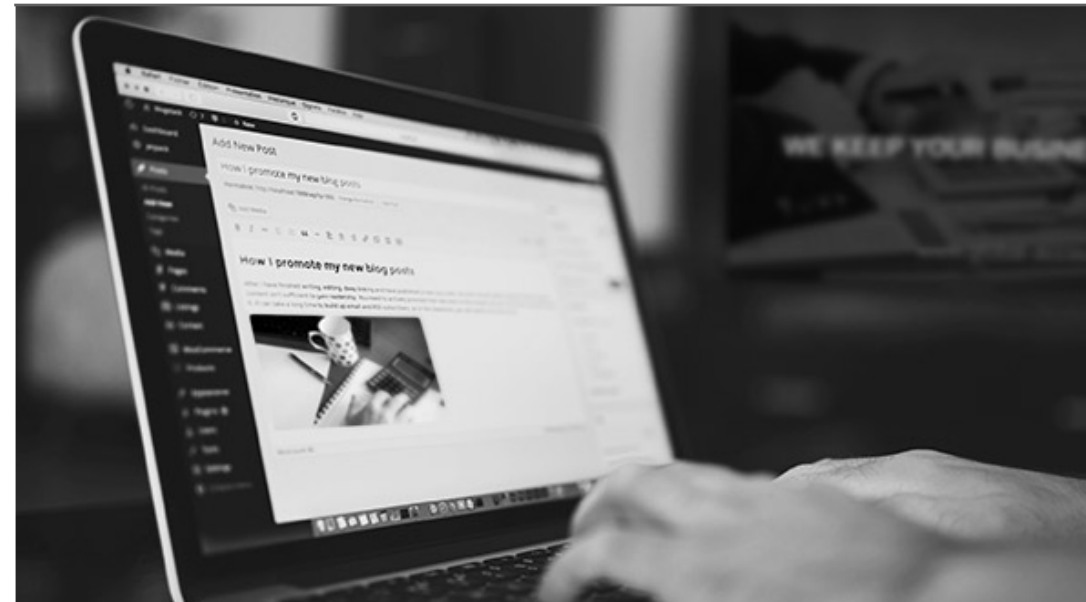
# 产品目录 & 内容管理

Product Catalog & Content Management



## 产品目录

SKUs、F/X标的，设备等。属性和metadata频繁更新。



## 内容管理

组织管理任何形式的数据：文本、视频、音频、图片等。



## 移动应用

移动/现代应用快速开发、快速上线、快速迭代。

# 产品目录 & 内容管理

Product Catalog & Content Management

功能要求	MongoDB
<b>数据模型</b> <ul style="list-style-type: none"><li>- 对象有复杂多变的属性，元数据</li><li>- 应用快速迭代，频繁更新数据模型</li></ul>	<b>数据模型</b> <ul style="list-style-type: none"><li>- 文档模型支持丰富的数据结构</li><li>- 灵活模式无变更成本，应用版本快速迭代上线</li><li>- GridFS支持存储二进制数据</li></ul>
<b>用户体验</b> <ul style="list-style-type: none"><li>- 用户查询请求条件组合多变</li><li>- 促销、秒杀等活动期间数据库性能不能降低</li><li>- 7*24在线</li></ul>	<b>用户体验</b> <ul style="list-style-type: none"><li>- 丰富的查询、索引、聚合</li><li>- 分布式架构auto-sharding，用户就近访问数据</li><li>- 一库多用：复制集同时满足在线交易与分析型需求</li></ul>
<b>实时分析</b> <ul style="list-style-type: none"><li>- 用户个性化推荐</li><li>- 实时统计销售指标与内容消费情况</li></ul>	<b>实时分析</b> <ul style="list-style-type: none"><li>- 聚合Aggregation Pipeline</li><li>- AI: Spark connector、R &amp; Python Drivers</li><li>- 数据可视化：MongoDB Charts &amp; BI Connector</li></ul>

# 主机下移 & 数据中台

## Mainframe Offload & Operational Data Layer



### 成本

大量查询压力转移到ODL上，更好的性能与扩展性，降低成本。



### 敏捷

快速响应业务变化，ODL数据平台提升开发效率3-5倍。

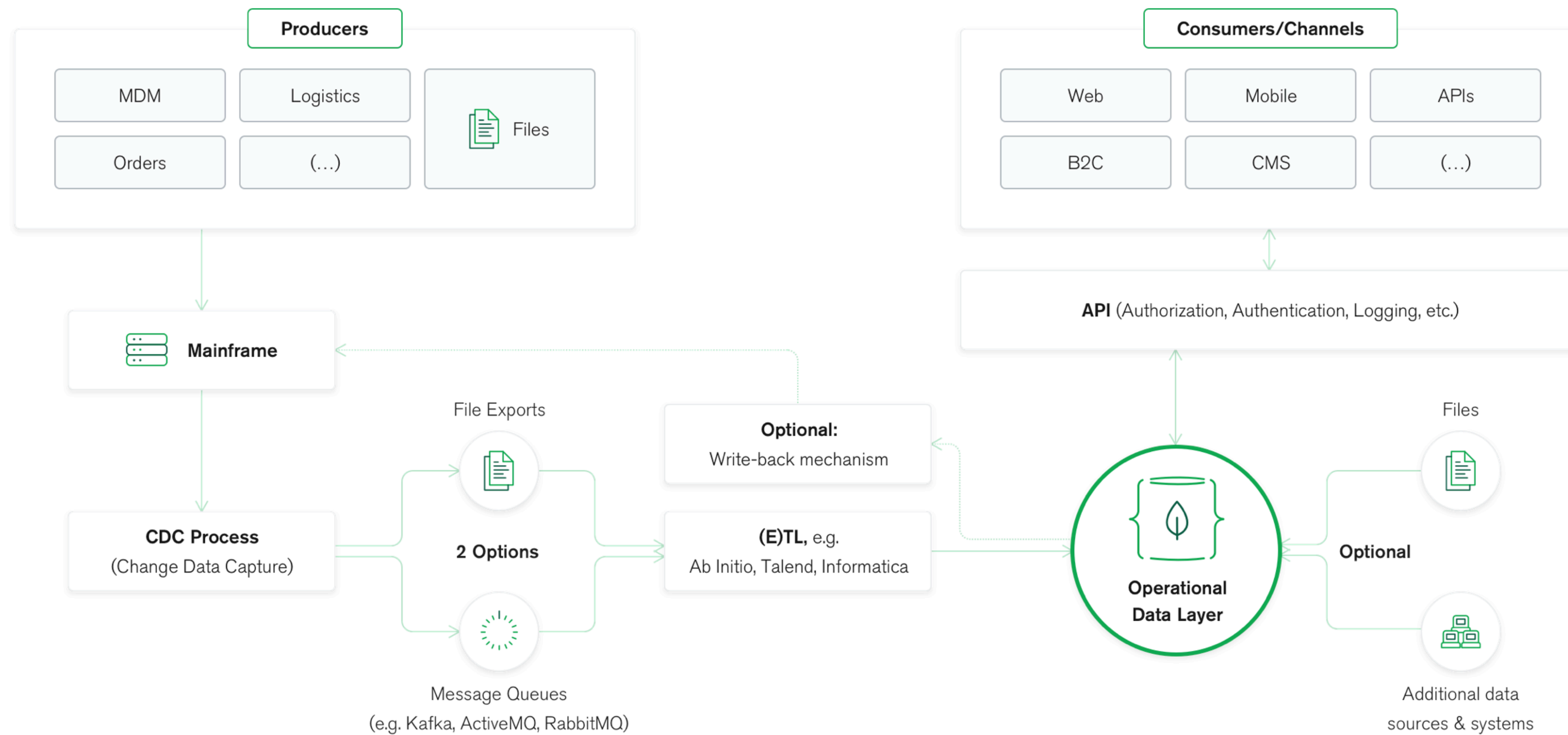


### 体验

7\*24，满足不断增长的用户和数据量增长。

# 主机下移 & 数据中台

Mainframe Offload



# 正在使用

用户	案例情况
 <p>中國銀行 BANK OF CHINA</p>	<p><b>用户交易</b></p> <ul style="list-style-type: none"><li>- 整合全行分散交易数据，实现客户层面一本帐</li><li>- 收支实时汇总交易金额，秒级响应</li><li>- 千亿级别数据实时处理</li></ul>
 <p>国泰君安证券 GUOTAI JUNAN SECURITIES</p>	<p><b>中台系统&amp;策略量化</b></p> <ul style="list-style-type: none"><li>- 用户全局数据、共享数据、标签、字典，处理三端一微30000/s的并发请求</li><li>- 产品资讯信息，模拟大盘数据</li><li>- 行情数据存放与分析，量化交易</li></ul>
 <p>中國東方航空 CHINA EASTERN</p>	<p><b>旅客数据</b></p> <ul style="list-style-type: none"><li>- 旅客多数据源的快速集成（航信、行程、行李、投诉、贵宾厅、网站、App等）</li><li>- Spark和MongoDB集成，大幅提升数据分析合处理速度</li><li>- 旅客敏感数据集中到MongoDB中，成为唯一用户数据来源</li></ul>

# 正在使用

用户	案例情况
	<p><b>机票缓存</b></p> <ul style="list-style-type: none"><li>- 机票价格缓存在MongoDB内存引擎中，快速响应票价请求</li><li>- 利用TTL索引，自动删除过期票价</li><li>- 分片架构应弹性对压力增长</li></ul>
	<p><b>网盘</b></p> <ul style="list-style-type: none"><li>- 存放网盘文件的metadata元数据</li><li>- 3PB数据量</li><li>- 分片架构应弹性对压力增长</li></ul>
	<p><b>智能捷创平台</b></p> <ul style="list-style-type: none"><li>- 保证普华永道全球所有用户能够协同工作</li><li>- 支持跨区读写文档、附件；读写高性能，低延迟</li><li>- 基于地理位置的分片集群</li><li>- 低代码、低花费、快速开发</li></ul>



谢谢