

MongoDB

单一视图的建设

葛 烨

锦木信息

锦木信息

MongoDB partner, 提供MongoDB订阅、咨询、技术支持服务。

主要客户

Shanghai Jinmu Information
Technology Co., Ltd.



Jinmu is a MongoDB reseller and services partner based on Shanghai. They provide NoSQL database, big data software, and consulting services solutions. Jinmu supports the full project lifecycle from initial design to production support. Industry focuses include financial services, transportation and retail. Their mission is to provide top-tier data solutions to customers.



什么是单一视图?

为什么要做单一视图?

- 数据统一
- 数据分析
- 用户画像
- 更多

单一视图适合的领域

- 互联网
- 客户/产品数据单一视图
- ECIF 企业客户
- 单一化数据源平台
- 主数据管理

- 要素

- 源系统
- 消费系统
- 处理 workflow
- 变更数据捕获(CDC)

为什么要选择MongoDB?

- 本身即为文档型数据库
- 传统RDBMS 漫长的开发周期
- 高效的数据迁移、存储及修改速度
- 未来数据的扩展
- (change streams)实时数据变更的获取

Case Studies (mongodb.com/customers)

Presentations (mongodb.com/presentations)

Free Online Training (university.mongodb.com)

Webinars and Events (mongodb.com/events)

Documentation (docs.mongodb.com)

MongoDB Enterprise Download (mongodb.com/download)

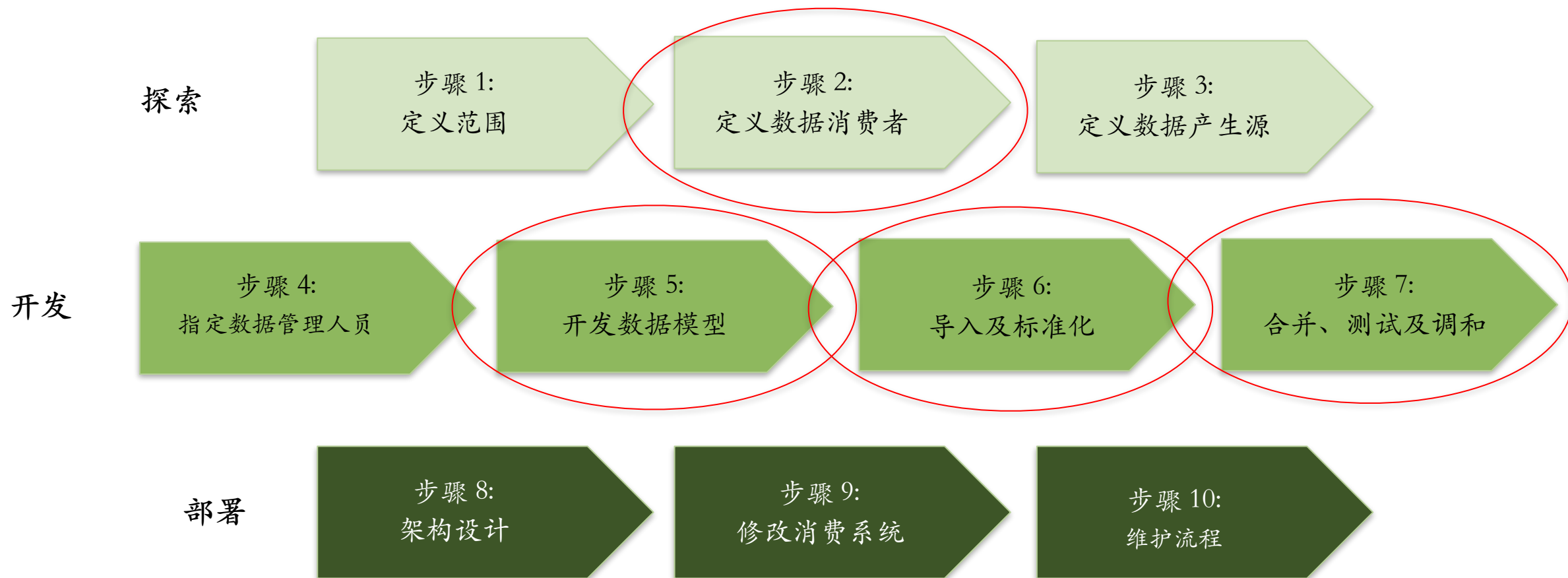
MongoDB Atlas database as a service for MongoDB

(mongodb.com/cloud)

MongoDB Stitch backend as a service ([mongodb.com/](https://mongodb.com/cloud/stitch)

[cloud/stitch](https://mongodb.com/cloud/stitch))

十步方法论



定义范围

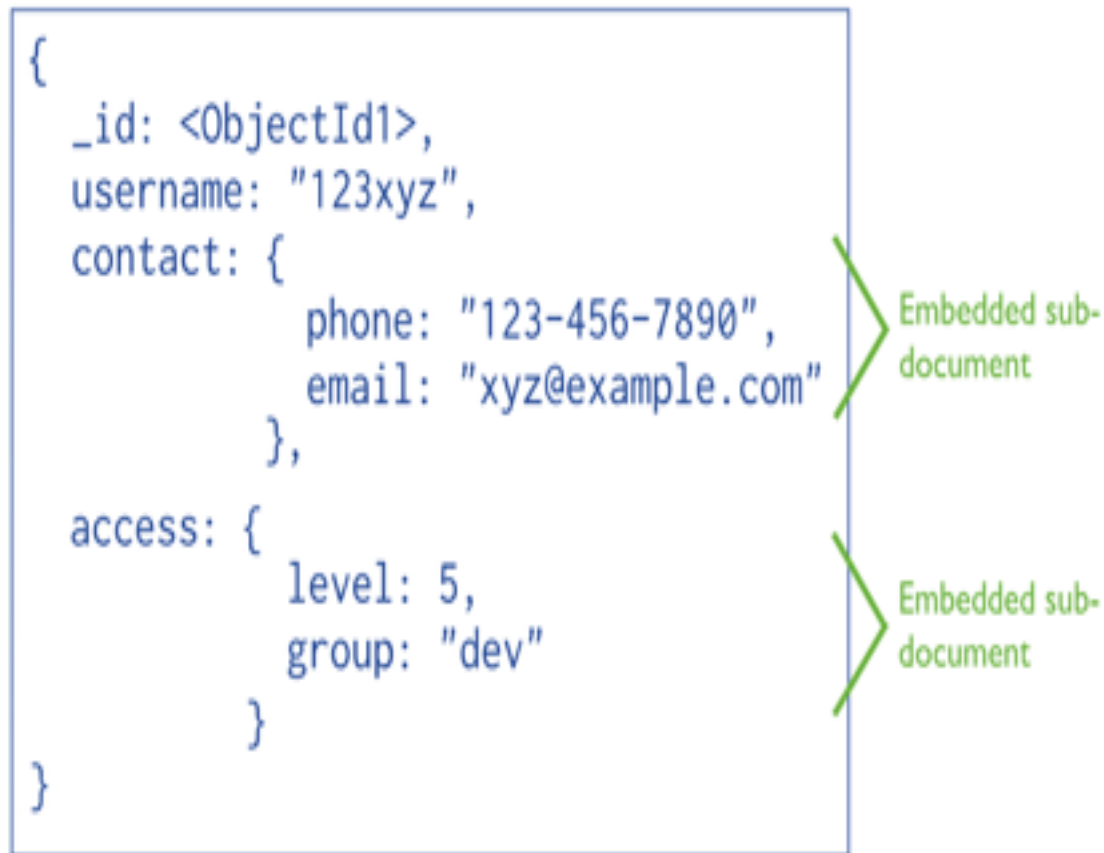
- 功能范围：初期的统一视图的范围尽量集中
- 业务目标：需要清晰且正确的实际合理的业务需求引导程序逻辑
- 参与部门：涉及到协同部门的沟通

定义数据的消费

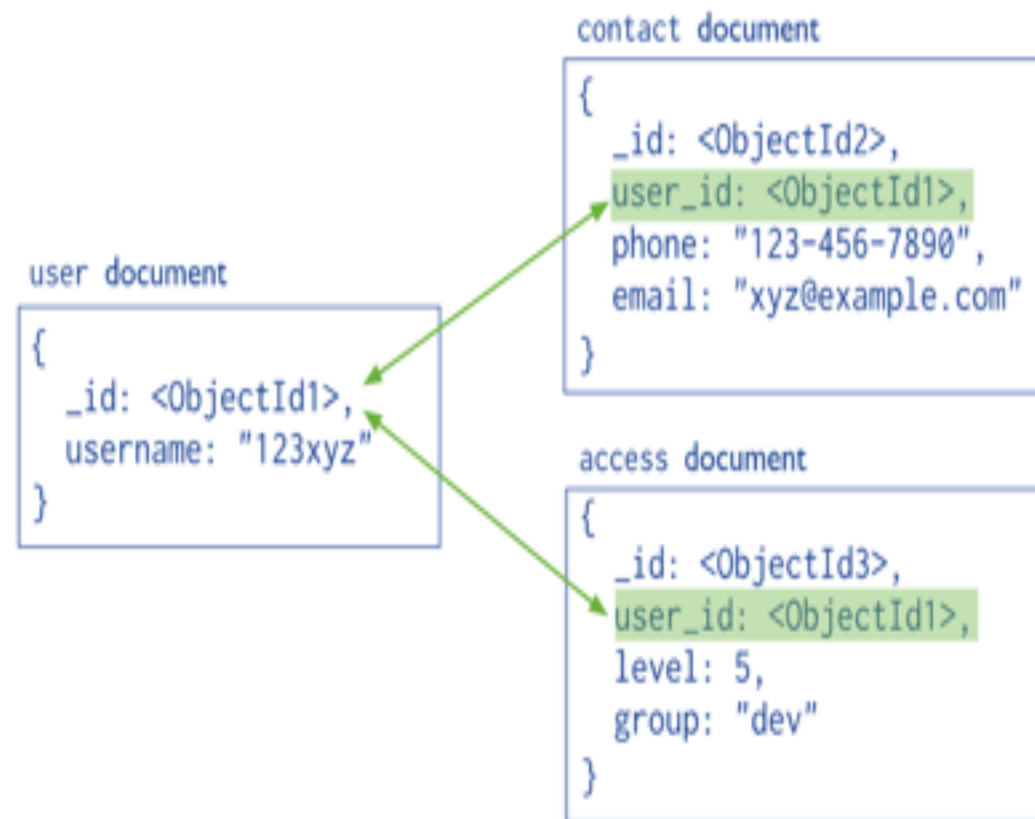
- 业务流程
- 具体的数据属性
- 数据的展现方式（用户具体的查询规则）

设计数据模型

内嵌



引用



影响建模的重要因素

- 标准化的数据格式
- 客户文档的公共属性
- 数据未来的扩展
- 查询方式

标准化数据格式

```
source_id: A_14  
f_name: James  
l_name: Bond  
dob: 07/14/1968  
eMail:007@spook.com
```

sourceA

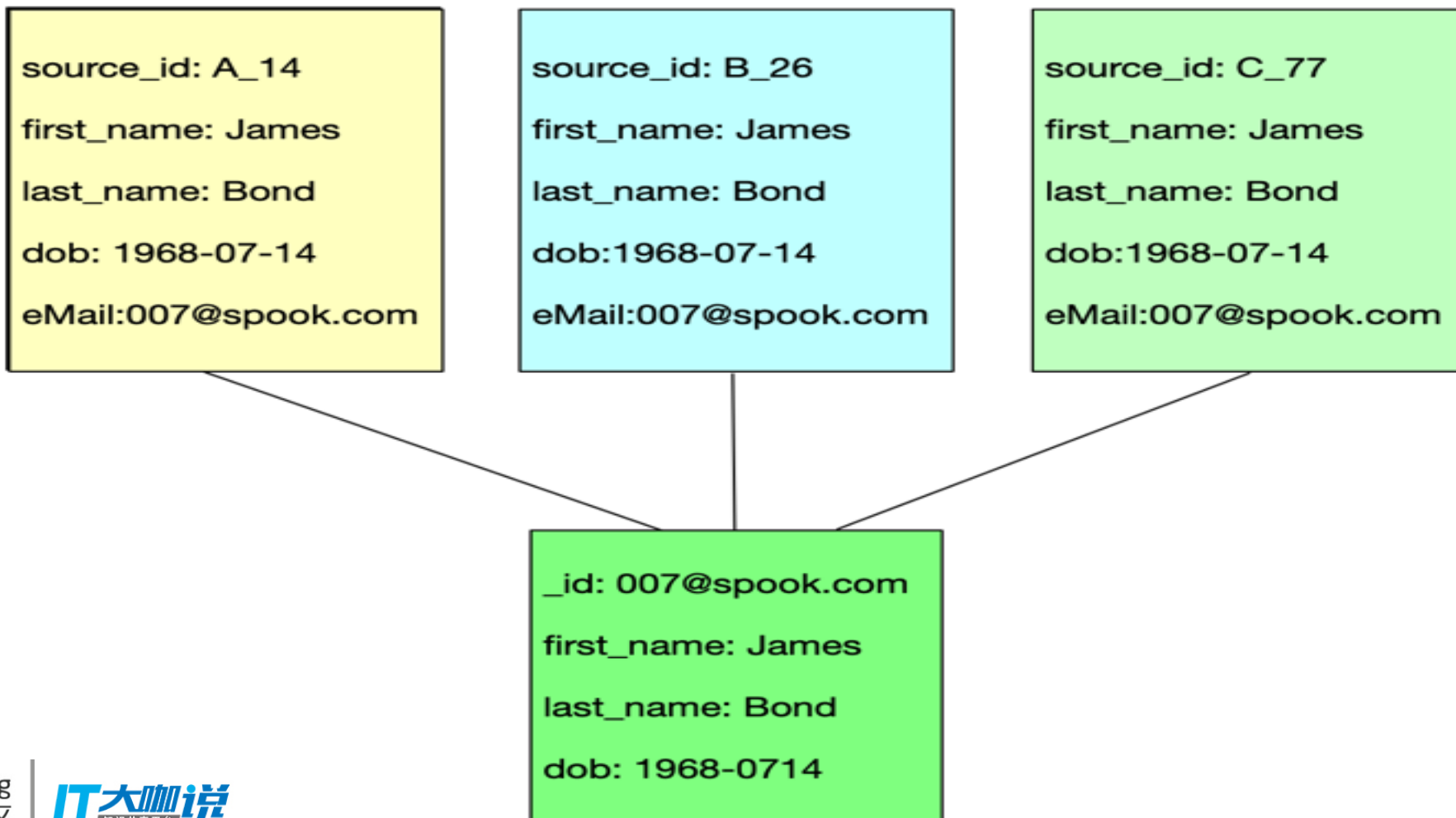
```
source_id: B_26  
name: James Bond  
bdate:July 14,68  
Email:007@spook.com
```

sourceB

```
source_id: C_77  
first: James  
last: Bond  
born:1968-07-14  
email:007@spook.com
```

sourceC

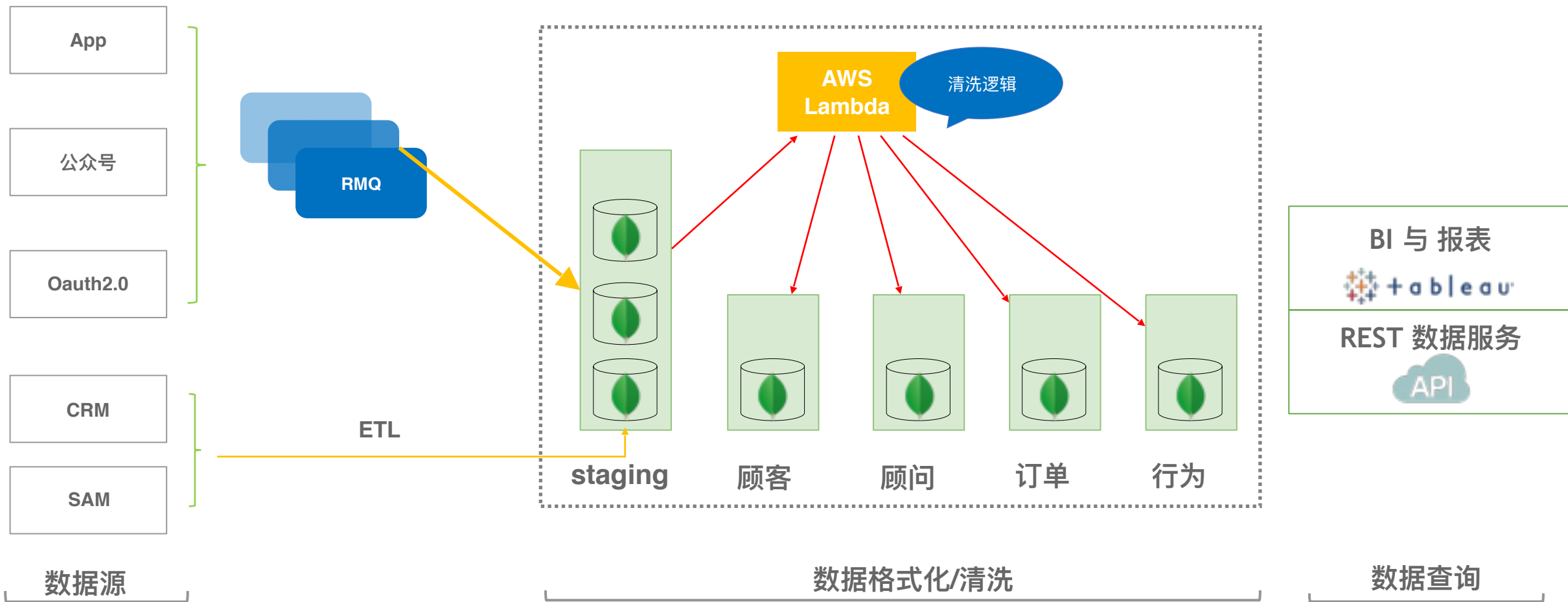
标准化的数据格式



数据的同步

- 历史数据的导入
ETL 工具 (kettle、infomatica)
- 源渠道增量数据的同步
更新的频率 (实时、定时)

实际的客户用例



数据变更订阅组建——changeStreams

New in version 3.6.

Change streams allow applications to access **real-time data** changes without the complexity and risk of tailing the **oplog**. Applications can use change streams to subscribe to all data changes on a single collection, a database, or an entire deployment **and immediately react to them**. Because change streams use the aggregation framework, applications can also filter for **specific changes** or **transform** the notifications at will.

ChangeStreams

```
{
  _id : { <BSON Object> },
  "operationType" : "<operation>",
  "fullDocument" : { <document> },
  "ns" : {
    "db" : "<database>",
    "coll" : "<collection>"
  },
  "to" : {
    "db" : "<database>",
    "coll" : "<collection>"
  },
  "documentKey" : { "_id" : <ObjectId> },
  "updateDescription" : {
    "updatedFields" : { <document> },
    "removedFields" : [ "<field>", ... ]
  }
  "clusterTime" : <Timestamp>,
  "txnNumber" : <NumberLong>,
  "lsid" : {
    "id" : <UUID>,
    "uid" : <BinData>
  }
}
```

Change Event

insert Event

update Event

replace Event

delete Event

drop Event

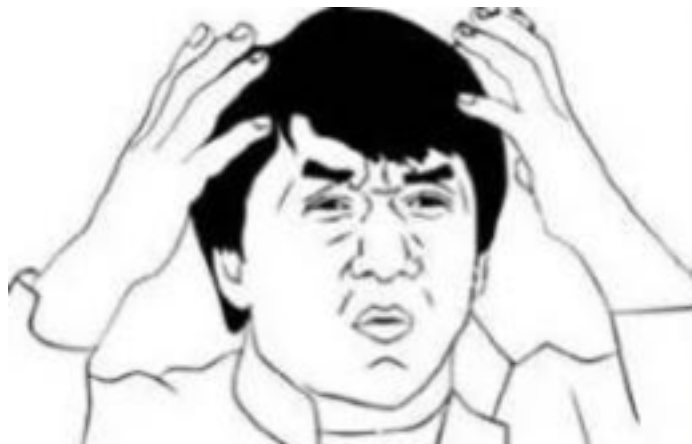
rename Event

dropDatabase Event

invalidate Event

Changestreams 的使用

- 适用的数据库架构单实例? 复制集? 分片?
- 如何捕获变更的数据?



Get一个简单的change streams

```
watchCursor = db.getSiblingDB("data").sensors.watch()
```

```
while (!watchCursor.isExhausted()){  
    if (watchCursor.hasNext()){  
        printjson(watchCursor.next());  
    }  
}
```

定向获取想要的change streams

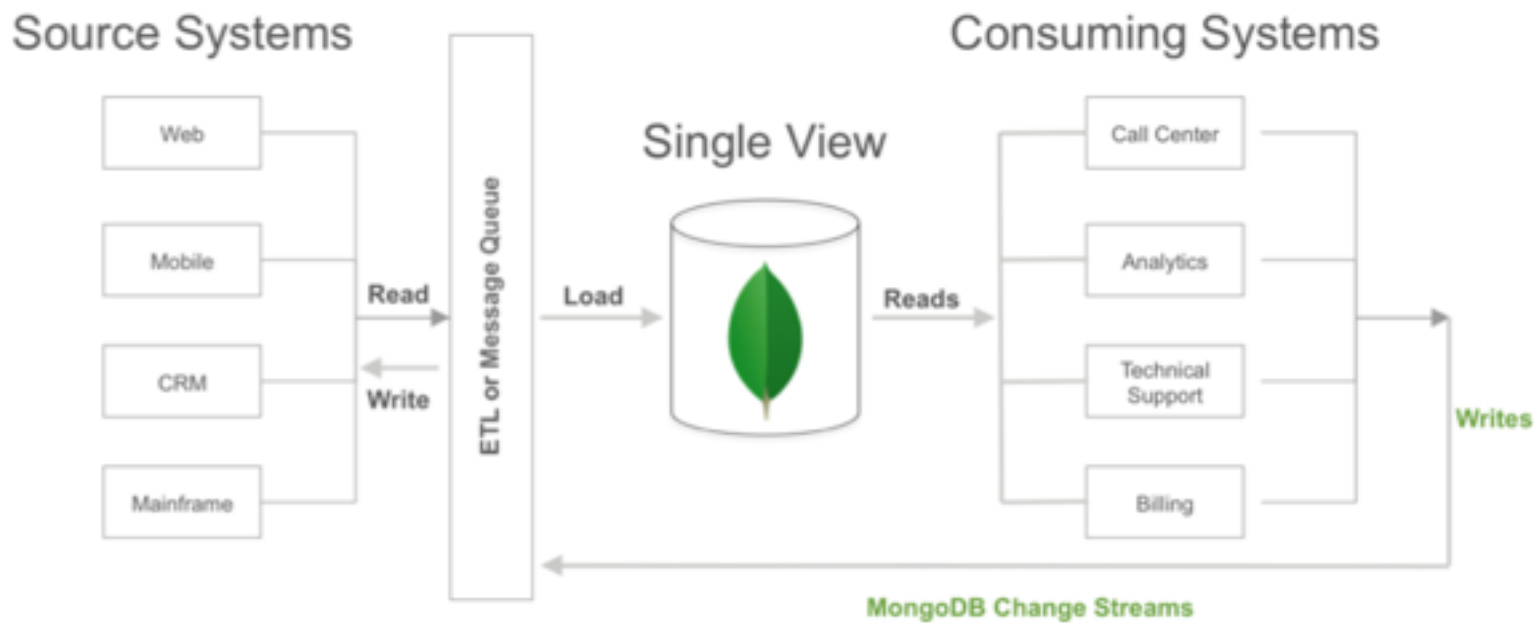
```
MongoCollection<Document> collection = db.getCollection("myTargetCollection");

// Create $match pipeline stage.
List<Bson> pipeline = singletonList(Aggregates.match(Filters.or(
    Document.parse("{'fullDocument.username': 'alice' }"),
    Filters.in("operationType", asList("delete")))));

// Create the change stream cursor, passing the pipeline to the
// collection.watch() method

MongoCursor<Document> cursor = collection.watch(pipeline).iterator();
```

新系统数据的回写



运维支持

- 调试、压测、生产性能问题解决
- 完整的监控体系
- 自动化部署与备份的解决方案

MongoDB企业版套件——OpsManager

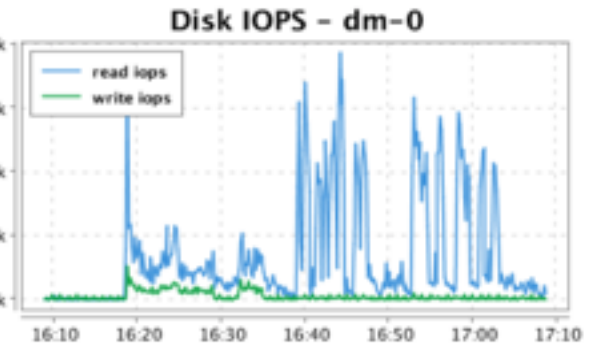
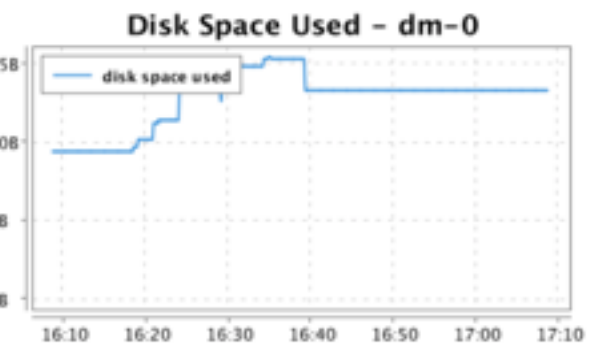
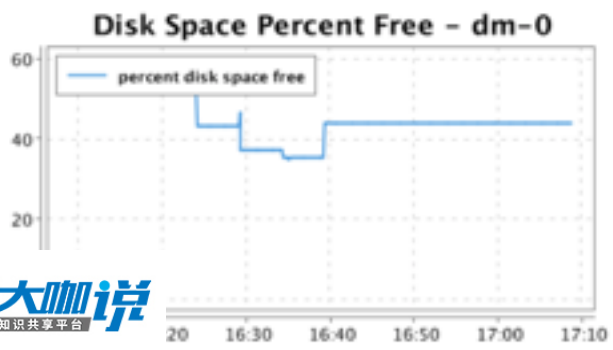
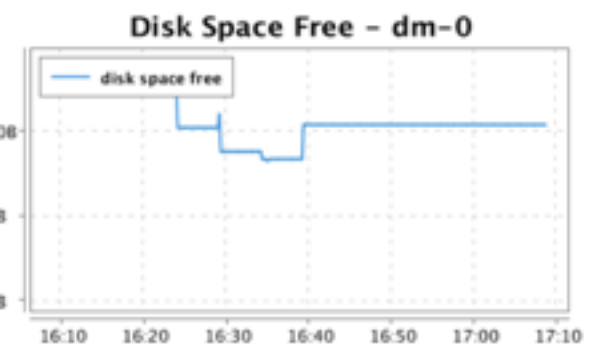
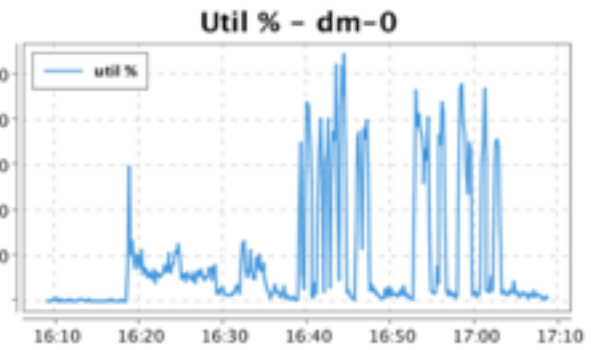
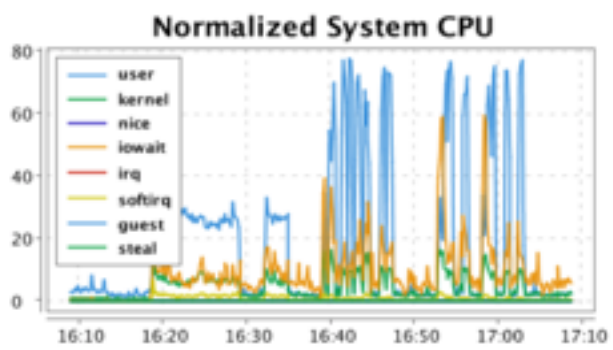
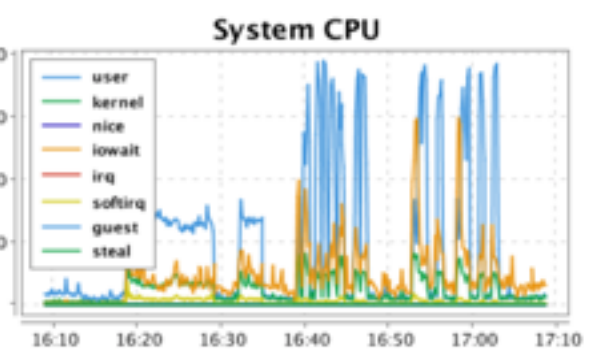
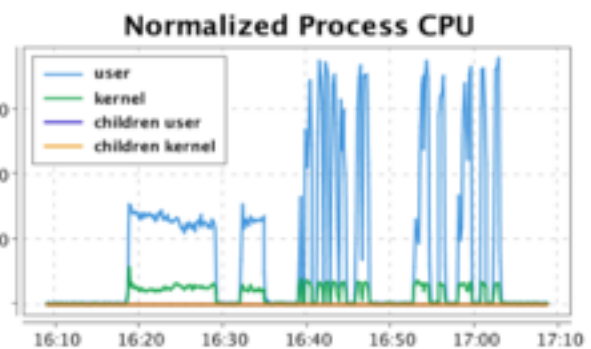
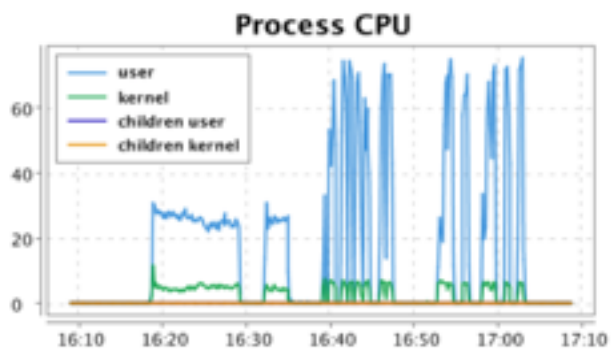
- 数据库、操作系统监控以及丰富的告警提示机制
- 自动化部署以及大型集群快速（无停机）维护
- 高效压缩及长周期的备份
- 智能化性能优化提示

Project 0

ops:30000

October 17, 2018 5:08PM CST

DISPLAYED AS: avg / sec



MongoDB企业版套件——OpsManager

智能的索引推荐

两个慢查询语句导致性能问题：

1.

```
db.POCCOLL.find({
  "fld1":
  {
    $gt: ISODate("2018-10-13T08:00:28.336Z")
  }
}).sort({
  "fld6": 1
})
```

2.

```
db.POCCOLL.find({
  "fld1":
  {
    $gt: ISODate("2018-10-13T08:00:28.336Z")
  },
  "fld0":
  {
    $gt: 200
  }
}).sort({
  "fld6": 1
})
```



MongoDB企业版套件——OpsManager

SAMPLE QUERIES IMPROVED BY THIS INDEX

FIND

fld1: Object
\$gt: Sat Oct 13 2018 8:00am
fld0: Object
\$gt: 200

SORT

fld6: 1

Mon Oct 22 2018 12:59pm 76616 ms 50021376 / 101 nScanned / nReturned

METRICS FOR THESE QUERIES

QUERY INEFFICIENCY SCORE	EXECUTION COUNT	AVERAGE EXECUTION TIME
495261	1	76616 MS

FIND

fld1: Object
\$gt: Sat Oct 13 2018 8:00am

SORT

fld6: 1

Mon Oct 22 2018 9:28am 60171 ms 50021376 / 101 nScanned / nReturned

METRICS FOR THESE QUERIES

QUERY INEFFICIENCY SCORE	EXECUTION COUNT	AVERAGE EXECUTION TIME
495261	1	60171 MS

Index Advice

An index for this query shape will improve the efficiency of read operations. To create an index, use `db.collection.createIndex()` in the mongodb shell or a [similar method from your driver](#).

```
db = db.getSiblingDB("POCDB")
db.getCollection("POCCOLL").createIndex({
  "fld6": 1
}, {background: true})
```

Need more help? [View documentation on indexes.](#)

Close

谢谢!