

Architect

SACC

2022 中国系统架构师大会

SYSTEM ARCHITECT CONFERENCE CHINA 2022

· 激发架构性能 点亮业务活力

云上会议 网络直播 | 2022年10月27-29日

IT168.com

ChinaUnix.net

ITPUB

# 实时流数据平台架构实践

光大银行

王磊

大数据团队负责人

# 个人介绍

## 王磊

- 光大银行资深架构师、大数据团队负责人
- 信通院大数据产品评测专家评委
- 《分布式数据库30讲》专栏作家
- 金融数士公众号作者

目前主要负责光大银行大数据基础平台、实时数据平台等系统建设及数据技术产品研发工作



# 实时数据处理体系的演进过程

2017-2021

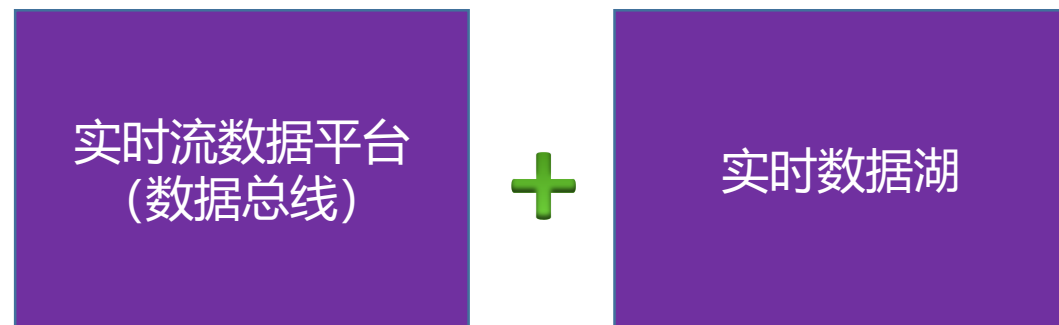


操作对象：消息

处理模式：单笔+微批

定位：数据平台、资源平台

2022



操作对象：流数据

处理模式：单笔

定位：数据平台

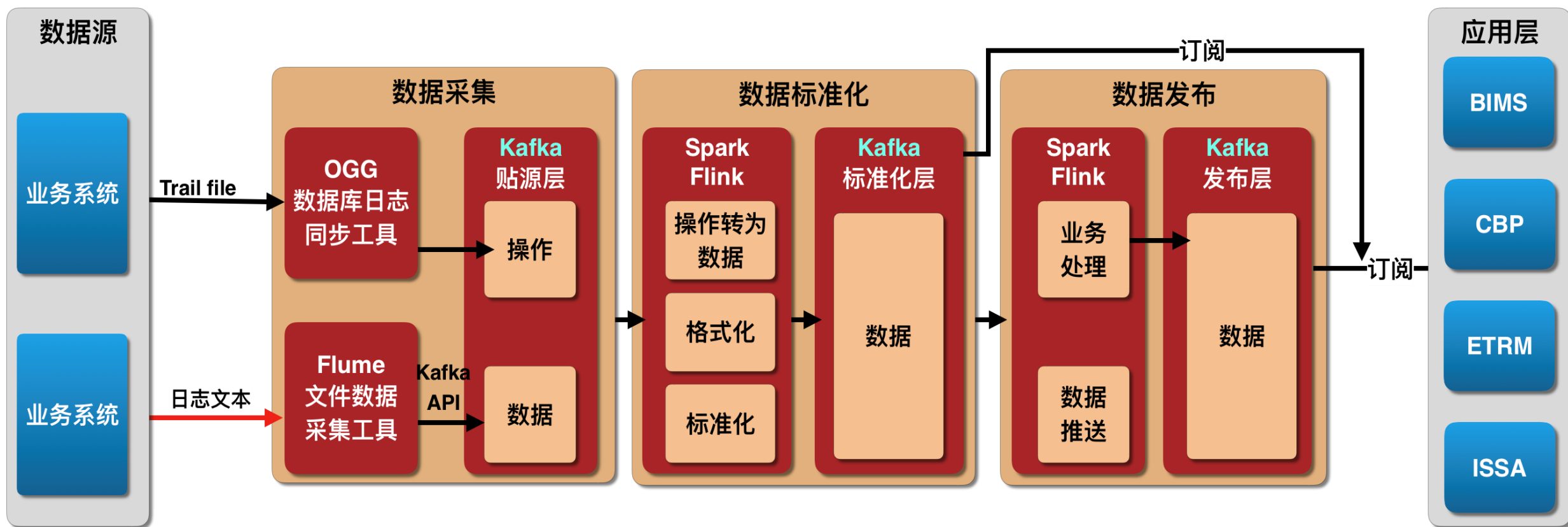
操作对象：数据表

处理模式：微批

定位：数据平台



# 准实时数据平台



# 准实时数据平台的特点

1. 以“消息”为传输实体
2. 向应用暴露原生接口
3. 数据多层次存储
4. 分布式消息队列为核心 (Kafka)
5. 内置流计算引擎 (Flink)

# 准实时数据平台的问题

## 存在问题

## 原有解决办法

## 不足之处

### 1. 数据管理能力不足

1.1 消息不等同于数据

1.2 部分字段变更操作

1.1 定制化数据转换程序, avro->json

1.2 平台内置存储全量数据基线

1.1 开发成本高, 耦合流计算资源

### 2. 数据结构上下游耦合紧密

上游变动引发的改造成本高风险大

### 3. 原生接口变动, 波及所有应用

应用被动跟随Kafka版本变动

### 4. 数据平台和资源平台叠加

造成系统定位不清晰

# 实时流数据平台的新办法

## 存在问题

### 1. 数据管理能力不足

1.1 消息不等同于数据

1.2 部分字段变更操作

### 2. 数据结构上下游耦合紧密

上游变动引发的改造成本高风险大

### 3. 原生接口变动，波及所有应用

应用被动跟随Kafka版本变动

### 4. 数据平台和资源平台叠加

造成系统定位不清晰

## 解决办法

### 1. 消息管理转变为数据管理

1.1 **schema registry** 统一管理，将逻辑与算力分离

1.2 转移到实时数据湖解决

### 2. 平台作为数据属主，上下游解耦

下游系统个性化订阅（**schema registry**），避免无效变更

### 3. 提供SDK，封装接口

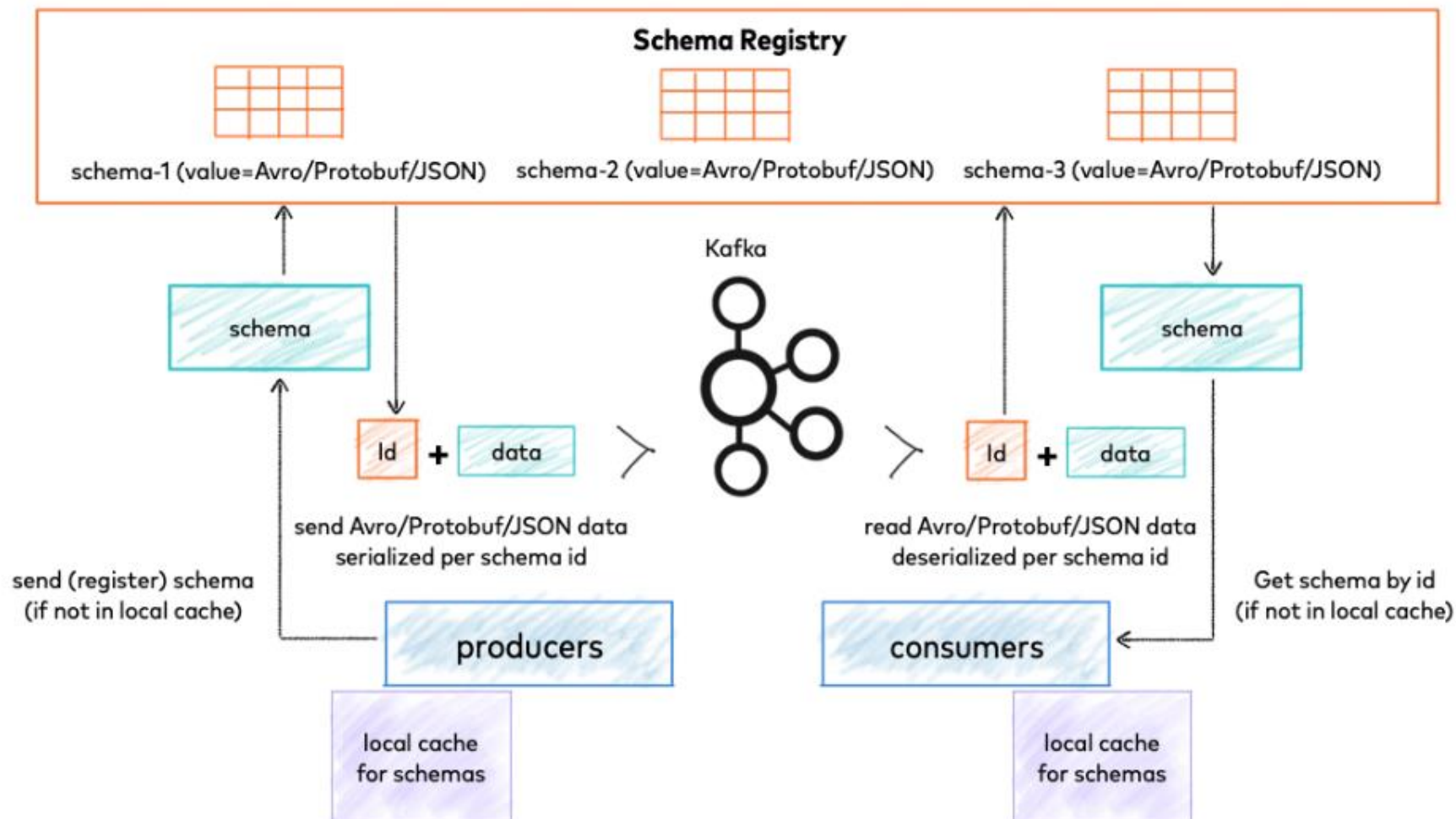
屏蔽内部设计复杂性，高于组件的平台边界

### 4. 剥离资源平台属性

定位清晰，功能聚焦



# Kafka Schema Registry



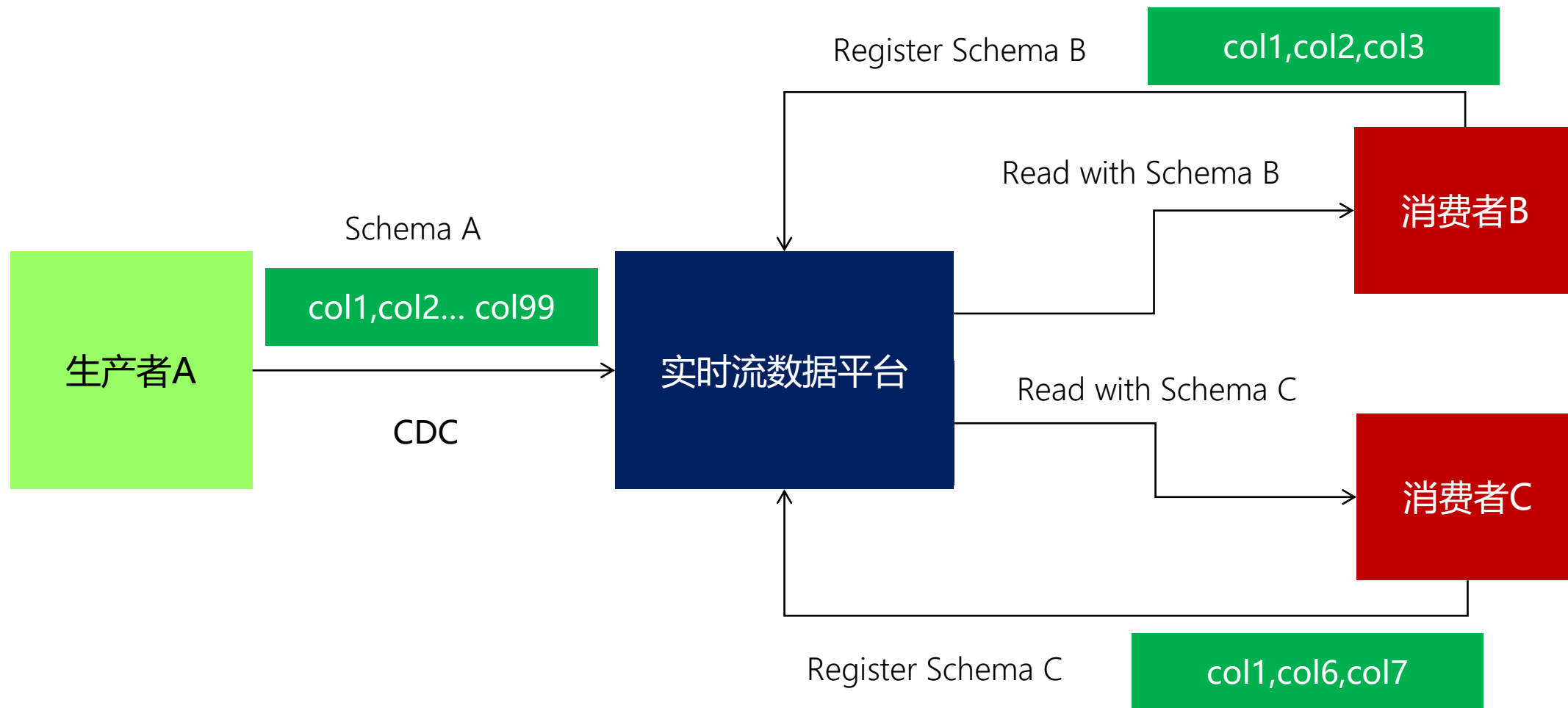
# Schema Registry 特性

- Schema Registry 是一个独立的服务器进程，可以运行在Kafka之外的机器上。
- Schema Registry 设计为分布式，单主架构，ZooKeeper/Kafka 协调主选举，只有master节点能进行数据写入。
- Schema Registry的工作是维护所有Schema的 “数据库”。该“数据库”保存在内部 Kafka topic中，并缓存在 Schema Registry 本地中以实现低延迟访问。
- 多个Schema Registry集群可服务于同一个kafka集群。
- Schema Registry为每个注册的模式分配全局唯一ID。分配的ID保证单调递增且唯一，但不一定是连续的，基于写入 Kafka 存储的最后一个 ID
- 在master重新选举期间，只有在新master节点同步旧master存储中的所有记录后才会进行批量分配Schema ID

# 模式演化 *schema evolution*

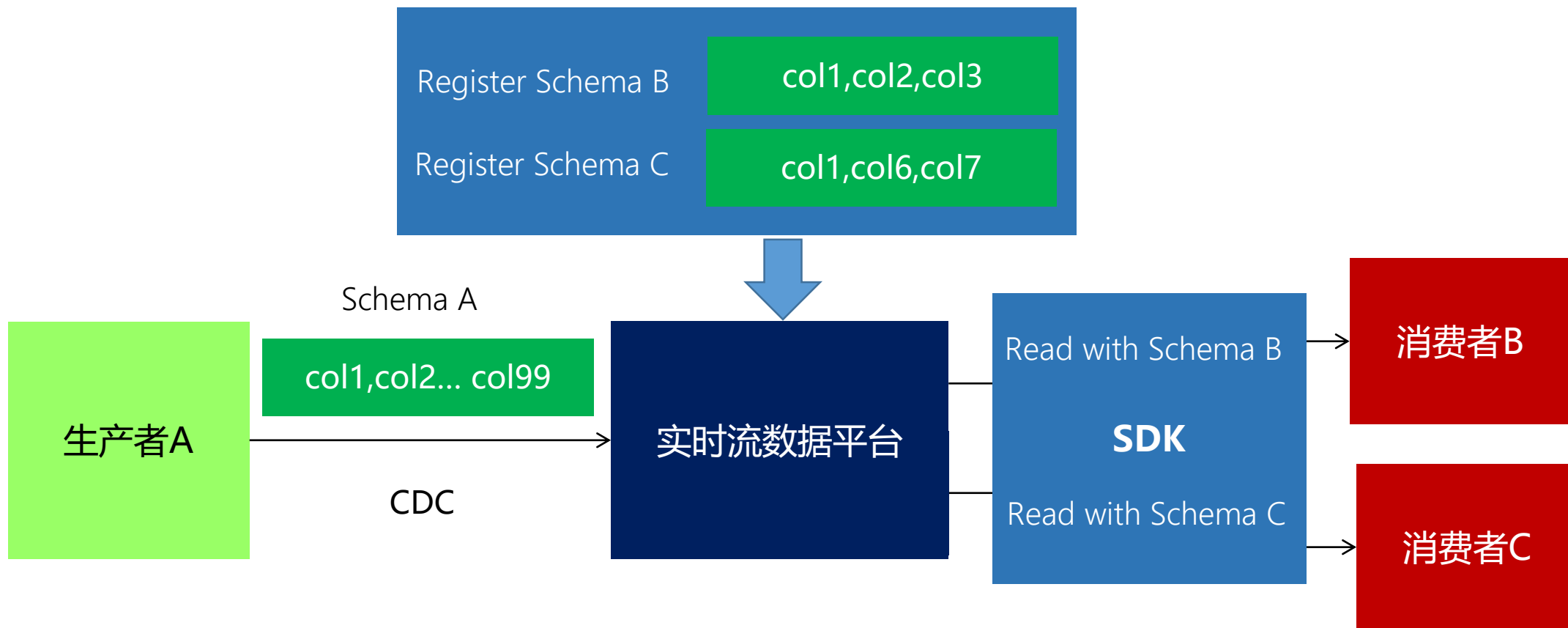
兼容类型	允许更改	描述	示例
BACKWARD	添加带默认值字段 删除字段	使用新Schema的消费者可以读取使用最后一个Schema生成的数据	如果Topic的三个Schema按X-2、X-1和X的顺序更改，则该兼容性确保使用新Schema X的消费者可以处理使用Schema X或X-1的生产者写入的数据
BACKWARD_TRANSITIVE	添加带默认值字段 删除字段	使用新Schema的消费者能够处理所有注册Schema写入的数据	该兼容性确保使用新Schema X的消费者可以处理使用Schema X、X-1或X-2的生产者写入的数据
FORWARD	添加字段 删除带默认值字段	可以使用次新Schema读取使用新Schema生成的数据	兼容性可确保使用新SchemaX的生产者写入的数据可以由使用SchemaX或X-1的消费者处理
FORWARD_TRANSITIVE	添加字段 删除带默认值字段	使用所有已注册的Schema读取使用新Schema生成的数据	兼容性可确保使用新SchemaX的生产者写入的数据可以由使用SchemaX、X-1或X-2的消费者处理。
FULL	添加带默认值字段 删除带默认值字段	既向后兼容又向前兼容，旧数据可以用新Schema读取，新数据也可以用次新Schema读取	该兼容性确保使用新SchemaX的消费者可以处理使用SchemaX或X-1的生产者写入的数据；生产者使用新SchemaX写入的数据可以由消费者使用SchemaX或X-1处理
FULL_TRANSITIVE	添加带默认值字段 删除带默认值字段	既向后兼容又向前兼容，旧数据可以用新Schema读取，新数据也可以用最后一个Schema读取	该兼容性确保使用新SchemaX的消费者可以处理使用SchemaX、X-1或X-2，并且生产者使用新SchemaX写入的数据可以由消费者使用SchemaX、X-1或X-2。
NONE	所有更改	Schema兼容性检查被禁用	消费者需要和生产者同时使用同一种Schema

# 数据编排——消费者个性化订阅





# SDK封装



1. Register动作集中在平台管控，应用对schema registry机制无感
2. 平台封装接口，屏蔽原生组件变动带来的部分影响



# 实时流数据平台与实时数据湖的差异

## 实时流数据平台 (数据总线)

操作模式:

面向单条数据, 单笔操作/微批操作 (复杂度高)

时效性

毫秒级~秒级

技术组件

Kafka

存储特点:

日志模式, 顺序读写

支持计算引擎:

Flink为主

## 实时数据湖

操作模式:

面向数据集合, 微批操作

时效性

分钟级

技术组件

Hudi/Iceberg+HMS+HDFS

存储特点:

支持parquet等多种文件结构, 对复杂查询有优化

支持计算引擎:

Hive/Spark/Flink

# 方案总结

## 纯消息队列管理实时流数据存在的问题

1. 数据管理能力不足
2. 数据结构上下游耦合紧密
3. 原生接口变动，波及所有应用
4. 微批操作不友好，读写性能差
5. 数据平台和资源平台叠加（个性化问题）

## 我们的解决思路

1. 基于schema库实现真正的数据管理
2. 基于SDK屏蔽技术复杂性和原生接口变化
3. 将微批操作转移到实时数据湖方案

# We are hiring

## 大数据开发工程师 & Java后端工程师

Java技术栈为主，有组件研发经验尤佳  
熟悉Hadoop大数据生态技术

联系邮箱：zh-wanglei@cebbank.com



金融数士



THANKS

Architect