

# 是时候改变你数仓的增量同步方案了

MLSQL Stack 系列

# 分享内容

**Stage1:**

**传统增量同步方案**

**Stage2:**

**基于Delta Lake新方案**

**Stage3:**

**新方案组件内部设计**

# 传统增量同步方案

## MySQL Binlog sync



# 优点

基本无

# 缺点

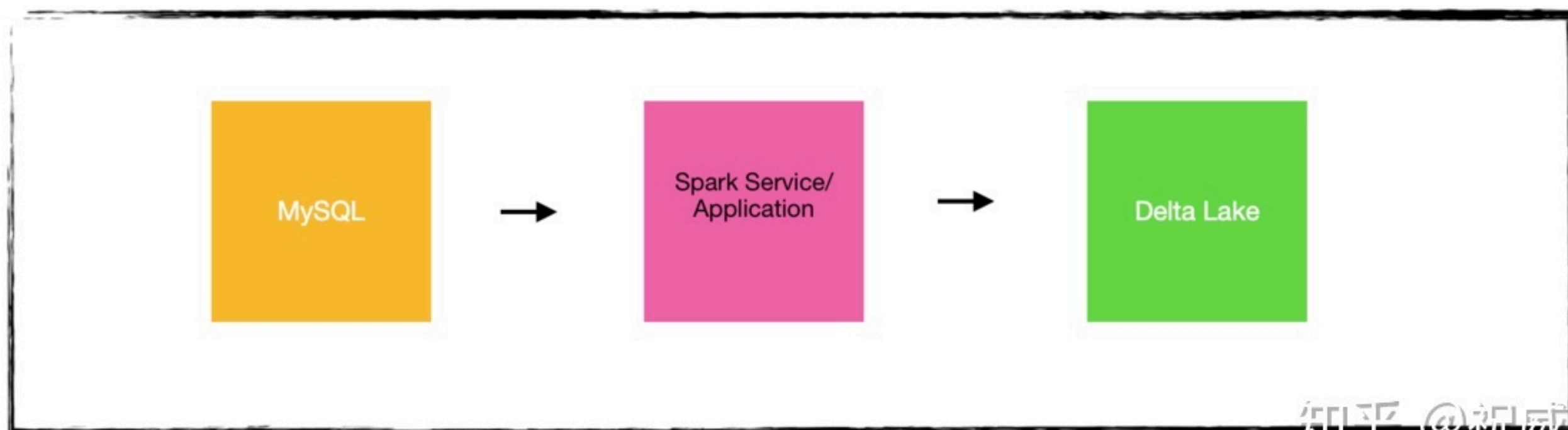
- 组件繁多（构建成本高）
- 每个组件都需要一个专家（人力成本高）
- 环节多，还涉及到数据的倒入导出（延时高）
- 传说中的Lambda架构？

**Lambda架构**

**用架构的复杂弥补存储不给力**

# 一个最直观的架构

MySQL Binlog sync



知乎 @祝威廉



**上面就是新方案**

# 代码直观感受

```
val spark: SparkSession = ???

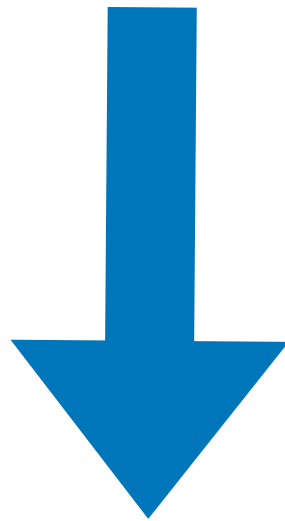
val df = spark.readStream.
  format("org.apache.spark.sql.mssql.sources.MLSQLBinLogDataSource")
  option("host", "127.0.0.1").
  option("port", "3306").
  option("userName", "xxxxx").
  option("password", "xxxxx").
  option("databaseNamePattern", "mssql_console").
  option("tableNamePattern", "script_file").
  option("binlogIndex", "4").
  option("binlogFileOffset", "4").
  load()

df.writeStream.
  format("org.apache.spark.sql.delta.sources.MLSQLDeltaDataSource")
  option("__path__", "/tmp/sync/tables").
  option("mode", "Append").
  option("idCols", "id").
  option("duration", "5").
  option("syncType", "binlog").
  checkpointLocation("/tmp/cpl-binlog2")
  .mode(OutputMode.Append).save("{db}/{table}")
```

配置MySQL信息

配置存储信息

(可能) N人N天



(可能) 1人几分钟

# 要解决的问题

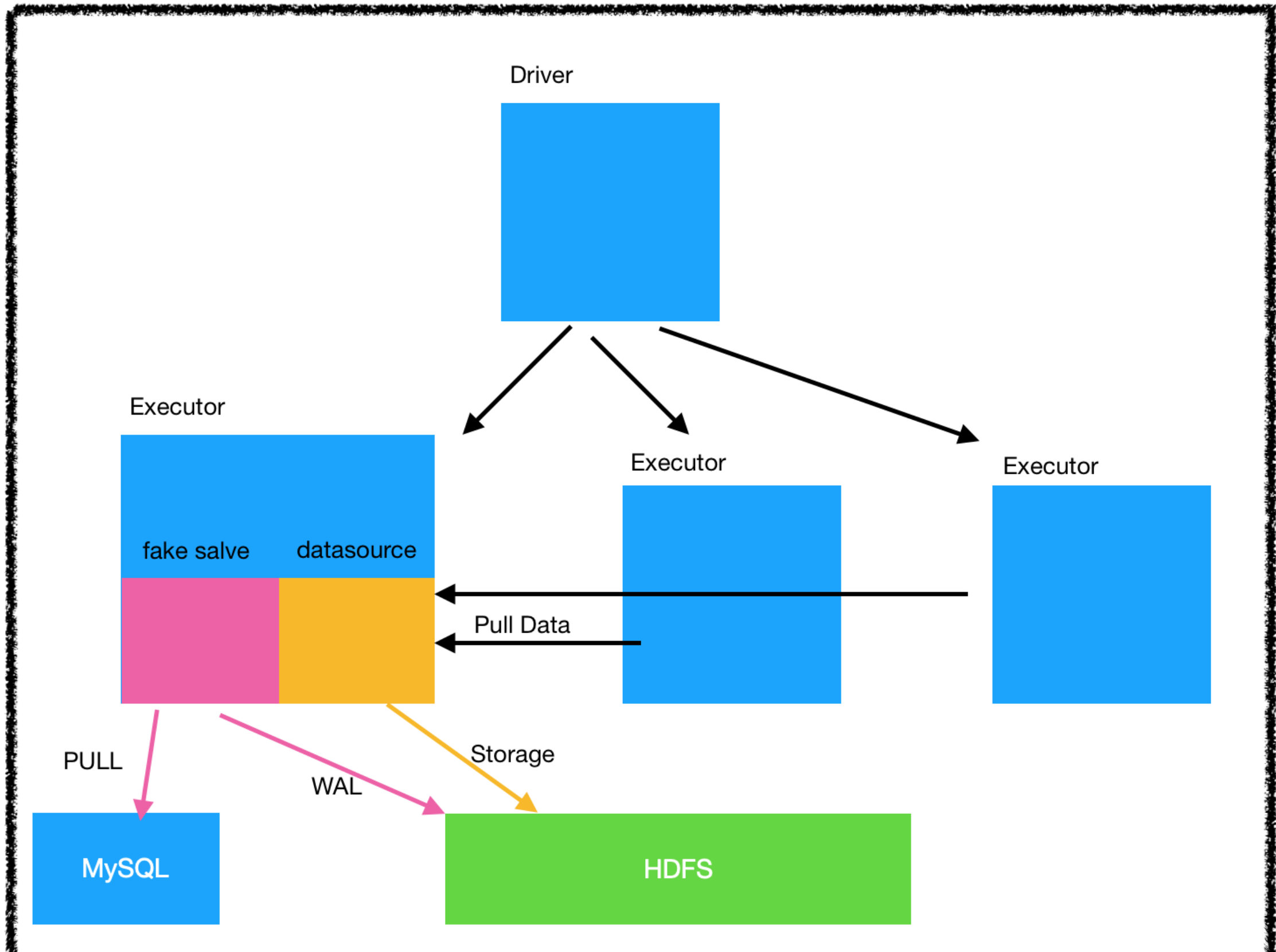
- Spark需要能够直接消费Binlog
- HDFS的存储需要支持Upsert/Partial Upsert
- 一段Replay 的逻辑

**我们通过MLSQL演示**

# Spark-Binlog技术细节

# Spark-Binlog 架构

Spark Service/Application



# Spark-Binlog流程

- 任意选择一个Exeuctor启动一个常驻的MySQL Slave
- 将消费到的Binlog转化为数据并且生成WAL写入HDFS
- 在Executor启动一个服务，提供对WAL的读取（满足Spark数据源需要回放的问题）
- 流式程序向该服务pull数据



# Delta-Plus 技术细节

# 要解决的问题

- 原生Delta更新放大的问题
- 文件数控制
- 不同类型的更新问题

# 原生Delta更新放大的问题

## 更新操作的秘密



祝威廉 1个月前

进入主题前的一句唠叨如果说，Delta带来的三大核心特性：流批共享upsert/delete/overwrite等操作版本回滚让我选最核心的特性的话，我会选择第二个。在大数据领域，我们也是一步步进化的，从最早的数据存储采用纯文本，到后面ORC/Parquet等面... [阅读全文](#) >

- 每次更新都需要发生全表扫描
- 每次更新都需要删除、新增大量文件
- 极端情况是重新生成整张表

# 解决办法

- 通过布隆过滤器，不需要join就能过滤掉不需要的文件
- 对数据根据主键组区间进行分区，保证每个文件里的数据都在自己的区间内

# 文件数控制

- Delta-Plus通过更新逻辑，可以确保文件数恒定，删多少文件，就新增多少文件

# 不同类型的更新问题

- Partial Merge
- Replace

# 它们都是开源项目

- spark-binlog: <https://github.com/allwefantasy/spark-binlog>
- delta-plus: <https://github.com/allwefantasy/delta-plus>

# 欢迎关注MLSQL

官网地址： <http://www.mlsql.tech>

知乎地址：

[https://zhuanlan.zhihu.com/c\\_205546986](https://zhuanlan.zhihu.com/c_205546986)