Fluid 加速云上大数据查询框架设计

2023.02.16

目录

背景技术介绍

Fluid介绍

云原生环境下的弹性数据抽象与加速平台

Data Abstraction
for heterogeneous data sources

异构数据抽象

Data Acceleration
with autoscaling and portable cache runtimes

自动数据加速

Data-aware Scheduling
to improve data affinity for apps intelligently

应用数据调度

背景技术介绍

Fluid介绍

01

提供云平台数据抽象的原生支持

数据密集型应用所需基础支撑能力功能化，实现数据高效访问并降低多维成本

02

基于容器调度管理的智能数据集编排

通过数据集缓存引擎与Kubernetes容器调度和扩缩容能力的相互配合，实现数据集可迁移性

03

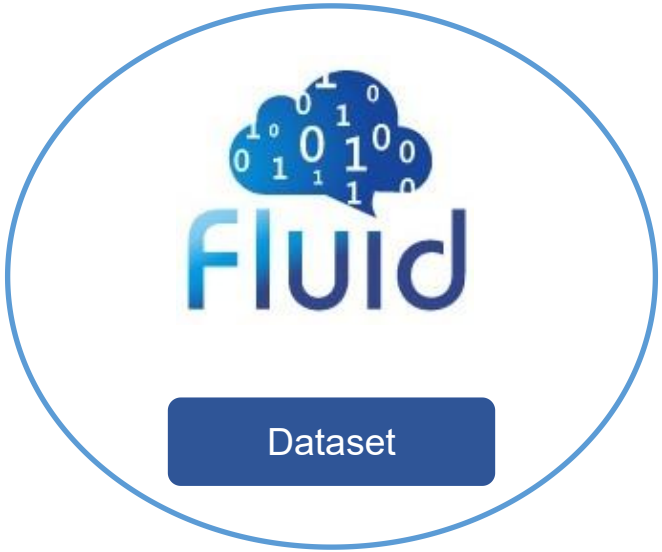
面向云上数据本地化的应用调度

Kubernetes调度器通过与缓存引擎交互获得节点的数据缓存信息，将使用该数据的应用以透明的方式调度到包含数据缓存的节点，最大化缓存本地性的优势

目录

需求分析

问题



…

Fluid 目前将通用数据集抽象为 Dataset CRD，为了支持云原生环境下大数据表查询的弹性加速，Fluid 社区设计一种云原生数据表抽象的概念 —— DataTable

目录

分析

作业特点：

- 深度学习场景：要求高吞吐
- 大数据查询场景：要求低延迟

缓存集群运行多长时间？

- 深度学习场景：作业完成后用户可自行关闭缓存集群
- 大数据查询场景：缓存集群需长期运行

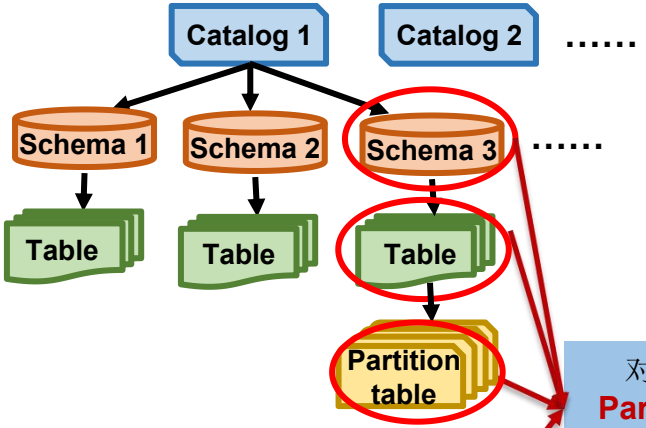
数据访问特征：

- 深度学习场景：对数据集依次访问
- 大数据查询场景：访问粒度多样化（列、单表、多表、分区表等）

缓存集群缓存什么数据？

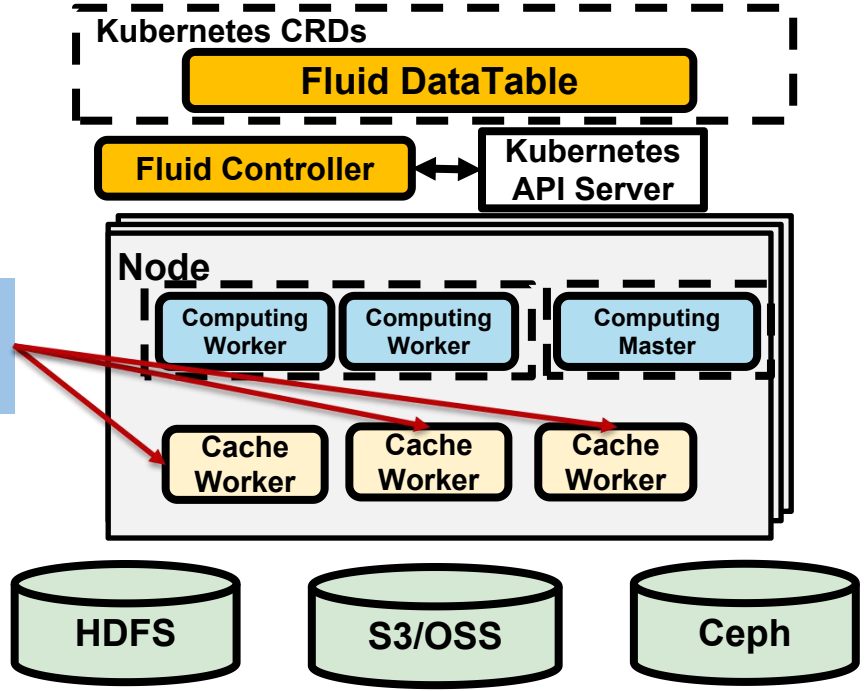
- 深度学习场景：缓存整个数据集
- 大数据查询场景：将查询所需数据来进行缓存

分析



id	name	age
001	Lucy	20
002	Mike	21
003	Lily	19

对 **Schema、Table、PartitionTable、Column** 进行多粒度抽象

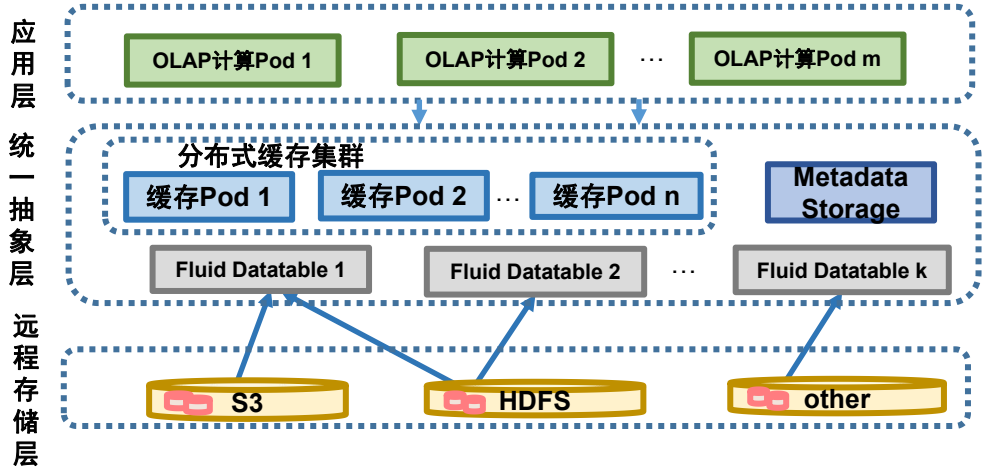


方案设计

DataTable - CRD

- 以用户或查询所需数据抽象数据
- 存储用户或查询所需数据表的元信息
  - 元数据存储位置(metastoreURL)，数据库名(schema)、表名(table)、分区表名(partitionTable)等
- 由于查询延迟要求高，需要部署一个长期运行的分布式缓存系统以减少集群冷启动时间。
- 多DataTable实例共享该分布式缓存系统，由分布式缓存提供数据访问加速能力

```yaml
apiVersion: data.fluid.io/v1alpha1
kind: DataTable
metadata:
  name: xxxx
spec:
  url: xxx.xxx.xxx.xxx:xxx   // hive hiveserver2 的url
  schemas:
    - schemaName: xx         // 可以理解为数据库名
      tables:
        - tableName: xx      // 数据表名
          partitionColumn:
            -  c_comment: AUTOMOBILE  // 分区表 （这里是以分区表为单位缓存）
    - schemaName: xxx
      tables:
        - tableName: xxxx  // 以整张表为单位缓存
    - schemaName: xxx      // 以整个库为单位缓存
```



Controller 设计

- **环境搭建阶段：**
  - K8S集群创建好对应的CRD以及其Controller
  - 集群管理员创建全局共享的缓存集群

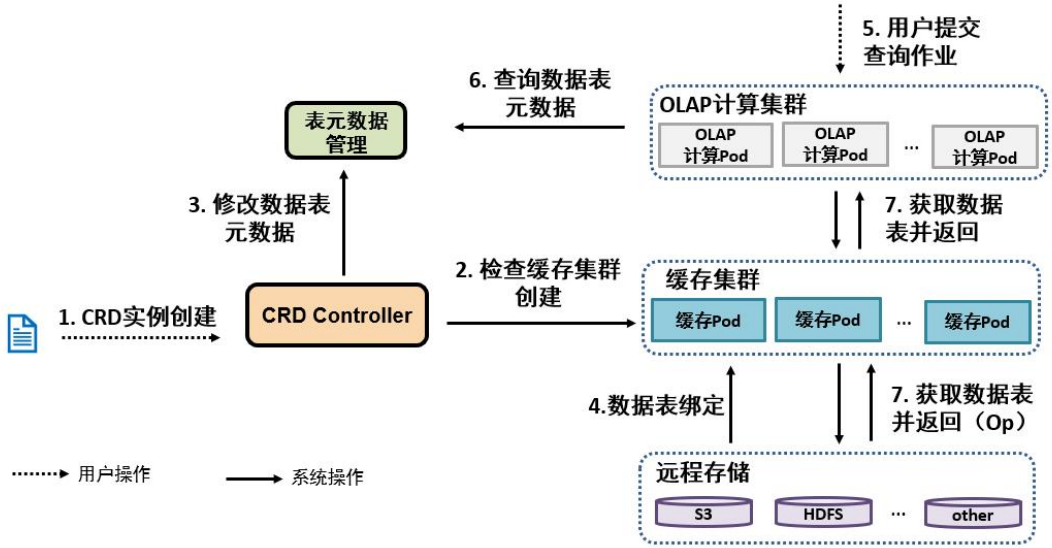
- **查询执行阶段：**
  - 用户创建CRD实例
  - Controller检查缓存集群创建、修改元数据管理模块中表位置信息（即将对应数据表进行挂载）
  - 用户执行查询

- **查询结束阶段：**
  - 用户删除CRD实例
  - Controller判断是否存在其他用户使用该表，如没有恢复原有表位置信息

方案设计

运行流程

1. 用户创建 DataTable CRD 实例；
2. DataTable Controller 检查缓存系统的状态；
3. 修改 DataTable 中指定表数据的位置源信息；
4. 缓存系统挂载提交的 DataTable 中指定表数据；
5. 用户提交查询作业；
6. 查询执行引擎执行作业，缓存命中则从缓存中直接读取，否则从远程存储系统中读取，并存入缓存系统。



目录

后续规划

- 在 Fluid 中实现 DataTable CRD 以及 Controller 代码
- 对接大数据查询系统
- 按照实时访问情况动态调整数据副本
- ……

谢 谢！ Q&A