

Hive技术原理

www.huawei.com





前言

- FusionInsight HD中Hive组件在社区版本Hive 基础上，加入了众多企业级定制化特性，如Colocation建表，列加密，语法增强等特性。整个产品在高可靠，高容错，可扩展性以及性能等各方面较社区有巨大提升。

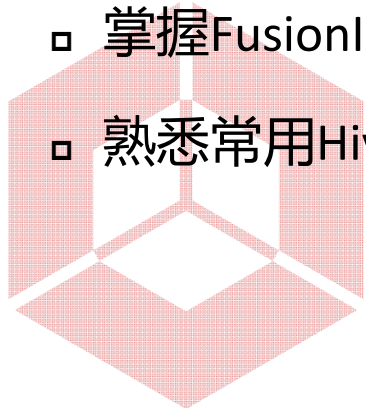


泰克教育
TECH EDUCATION



目标

- 学完本课程后，您将能够：
 - 掌握Hive应用场景与基本原理
 - 掌握FusionInsight中Hive增强特性
 - 熟悉常用Hive SQL语句

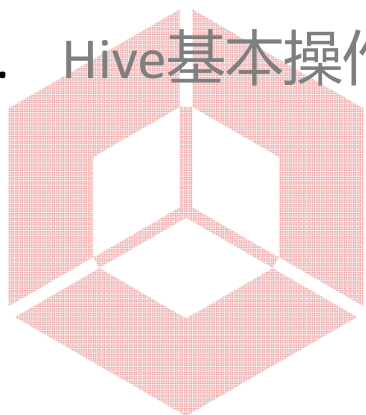


泰克教育
TECH EDUCATION



目录

1. Hive概述
2. Hive功能与架构
3. Hive基本操作



泰克教育
TECH EDUCATION

什么是Hive

- Hive是基于Hadoop的**数据仓库软件**，可以查询和管理PB级别的分布式数据。
- 它提供了如下功能：
 - 灵活方便的ETL(extract/transform/load)
 - 多种文件格式的元数据服务（自研的DBServer）
 - 直接访问HDFS文件以及HBase
 - 本身不具备计算能力，支持MapReduce，Tez，Spark等多种计算引擎

数据仓库和数据库



举个栗子

数据而设计，数据仓库是为分析

现在要准备一个舞台剧，首先我们要去市场上买东西，比如服装道具、灯光音响等等，我们**买东西的市场就是数据仓库**。我们把需要的东西买过来之后，节目组需要准备一个库房存放这些东西，**这个库房就是数据库**。

市场的商品有大量的冗余，并且分类存放，我们从中分析计算，挑选出自己需要的商品。而剧组仓库的东西没有冗余，需要用多少就放进来多少，不用的就清理出去，使我们处理业务的时候拿取方便。

同样的道理，在数据库中应当尽量避免数据冗余，一般采用符合范式的规则来设计表；而数据仓库在使用时有意引入冗余，采用反范式的方式来设计表。我们用数据仓库来处理数据，将结果存入数据库中，供业务使用。

数据仓库和数据库

数据而设计，数据仓库是为分析



举个栗子

我们要买服装道具就去服装市场，要买灯光音响就去电器百货市场，市场的设计就是面向主题。那什么是面向事务呢？我们把需要的东西买过来之后，存在我们自己的库房里面，这个库房就是为舞台剧存取物资准备的，东西按人分放，那么库房的设计就是面向事务。

同样的道理，数据库的表设计往往是针对某一个应用进行设计的。但是符合应用的表未必符合分析，所以就要依照分析需求，分析维度以及分析指标重新设计数据仓库的表结构。

数据仓库和数据库

1、设计目的不同：数据库是为捕获数据而设计，数据仓库是为分析数据而设计。

2、面向主题&面向事务



举个栗子

3、集成的

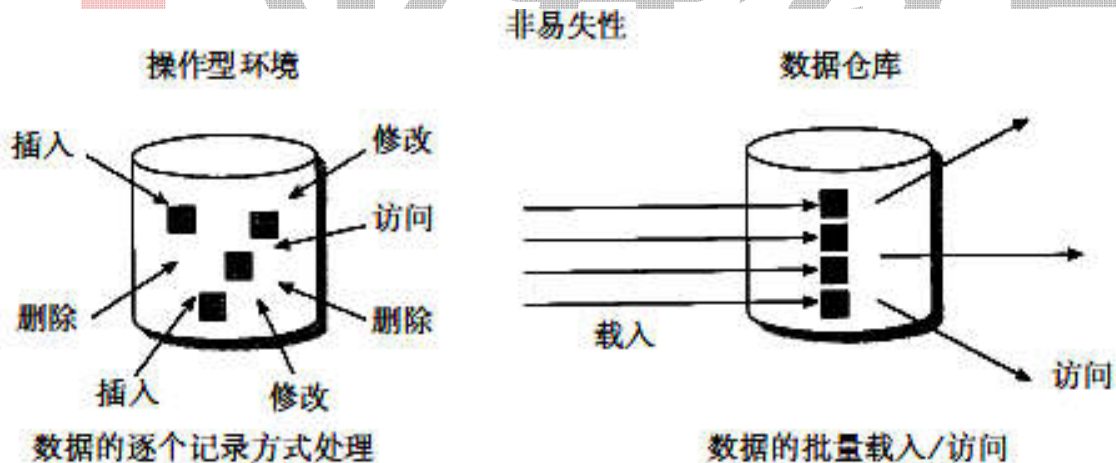
我们以借用“阿里”的业务场景为例。阿里的业务类型很多：天猫、淘宝、支付宝、蚂蚁花呗等等。每一个业务都需要单独的数据库存储相应的数据，数据库之间相互独立，并且往往是异构的。

当数据量累积到一定程度，我们想要做数据综合分析，比如用户多方画像，则需要使用数据仓库抽取、收集来自不同数据库的数据，对其进行清理归类以及系统的加工整合，消除源数据中的不一致，供具体分析使用。

分析的结果又可以存入各自业务对应的数据库中。

数据仓库和数据库

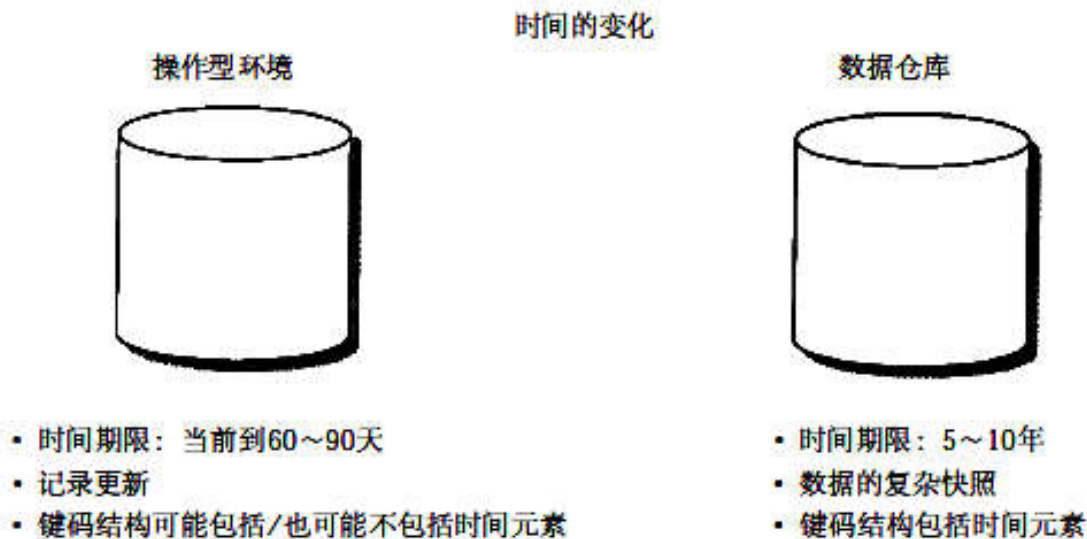
- 1、操作型数据库中的数据是实时更新数据库需要及数据仓库是为数据仓库的数据主要供企业决策分析之用，所涉及的数据操作主要是数据查询，一旦某个数据进入数据仓库以后，一般情况下将被长期保留，也就是数据仓库
- 2、面向主题&面向事务中有大量的查询操作，但修改和删除操作很少，通常只需要定期加载刷新。
- 3、集成的
- 4、相对的



数据仓库和数据库

1、操作型数据库主要数据,当前捕捉数据而设计的数据数据库是其中的数据通常包含历史信息,系统记录了企业从过去某一时点(如开始应用数据仓库的时点)到目前的各个阶段的信息,通过这些信息,可以对企业的发展历程和未来趋势做出定量分析和预测。

- 3、集成的
- 4、相对稳
- 5、反映月



Hive与传统数据仓库比较

	Hive	传统数据仓库
存储	HDFS，理论上可无限拓展。	集群存储，存在容量上限 ，而且伴随容量的增长，计算速度急剧下降。只能适应于数据量比较小的商业应用，对于超大规模数据无能为力。
执行引擎	有MR/Tez/Spark多种引擎。	可以选择更加高效的算法来执行查询，也可以进行更多的优化措施来提高速度。
使用方式	HQL（类似SQL）。	SQL。
灵活性	元数据存储独立于数据存储之外，解耦合元数据和数据。	低，数据用途单一。
分析速度	计算依赖于集群规模，易拓展，在大数据量情况下，远远快于普通数据仓库。	在数据容量较小时非常快速，数据量较大时，急剧下降。

Hive与传统数据仓库比较

	Hive	传统数据仓库
索引	低效，目前还不完善。	高效。
易用性	需要自行开发应用模型，灵活度较高，但是易用性较低。	集成一整套成熟的的报表解决方案，可以较为方便的进行数据的分析。
可靠性	数据存储在HDFS，可靠性高，容错性高。	可靠性较低，一次查询失败需要重新开始。 数据容错依赖于硬件Raid。
依赖环境	依赖硬件较低，可适应一般的普通机器。	依赖于高性能的商业服务器。
价格	开源产品。	商用比较昂贵。

数据仓库和数据库

综上所述：

由于数据库和数据仓库**设计目的**完全不一样，两者的**性能**有所差异，**用法**也是有很大差别的，这一点主要体现在**表的设计**和**操作类型**上。

但是数据库和数据仓库也并不是完全独立的两个东西，数据仓库可以借助于某些数据库来实现，这也是为什么有人说**数据仓库不是一个实体，而是一种方案**的主要原因。



东亮教育
TECH EDUCATION

Hive简介

- Hive是基于Hadoop的数据仓库软件，可以查询和管理PB级别的分布式数据。
- Hive特性：
 - 灵活方便的ETL(extract/transform/load)。
 - 支持MapReduce，Tez，Spark等多种计算引擎。
 - 可直接访问HDFS文件以及HBase。
 - 易用易编程。

Hive的优点

Hive的优点

高可靠、高容错

- 1.HiveServer采用集群模式
- 2.双MetaStore
- 3.超时重试机制

1

类SQL

- 1.类似SQL语法
- 2.内置大量函数

2

可扩展

- 1.自定义存储格式
- 2.自定义函数

3

多接口

- 1.Beeline
- 2.JDBC
- 3.Thrift
- 4.Python
- 5.ODBC

4

Hive缺点

Hive的缺点

延迟较高

- 1.默认MR为执行引擎。
- 2.MR延迟较高。

1

不支持物化视图

- 1.Hive支持普通视图，不支持物化视图。
- 2.Hive不能在视图上更新、插入、删除数据。

2

不适用OLTP

- 1.暂不支持列级别的数据添加、更新、删除操作。

3

暂不支持存储过程

- 1.当前版本不支持存储过程，只能通过UDF来实现一些逻辑处理。

4

Hive的应用场景

数据挖掘

- 用户行为分析
- 兴趣分区
- 区域展示

非实时分析

- 日志分析
- 文本分析

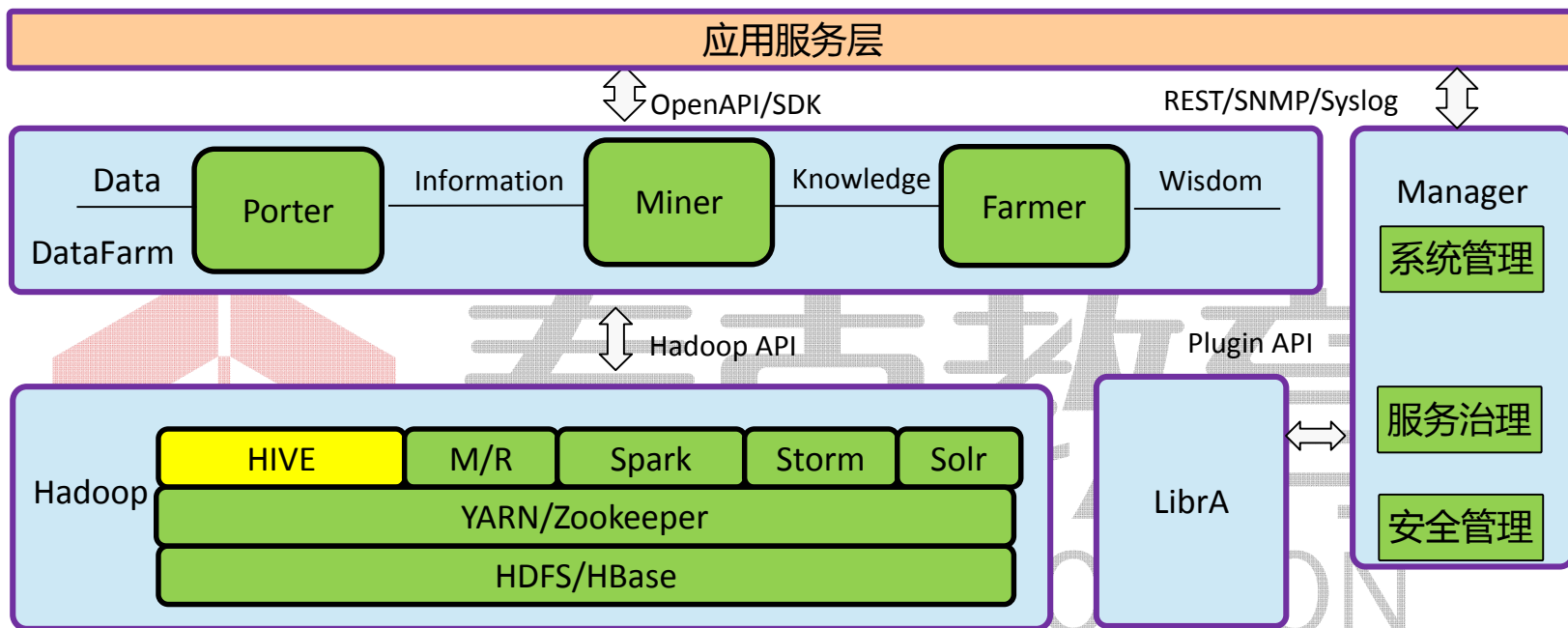
数据汇总

- 每天/每周用户点击数
- 流量统计

数据仓库

- 数据抽取
- 数据加载
- 数据转换

Hive在FusionInsight中的位置

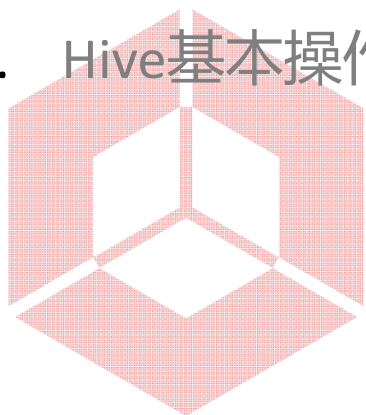


Hive 是一种数据仓库处理工具，使用类SQL的HiveQL语言实现数据查询功能，所有Hive的数据都存储在HDFS中。



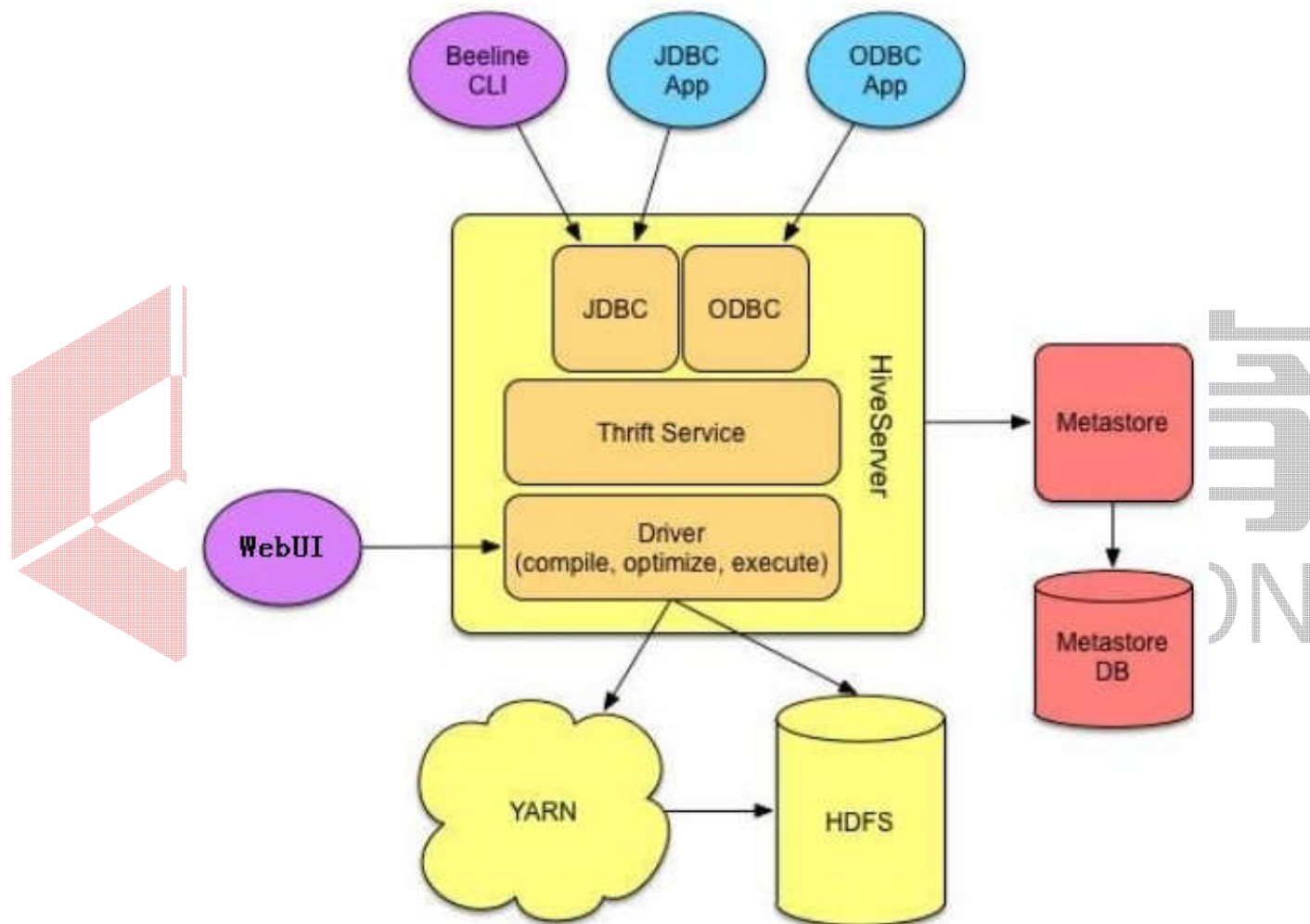
目录

1. Hive概述
2. **Hive功能与架构**
3. Hive基本操作

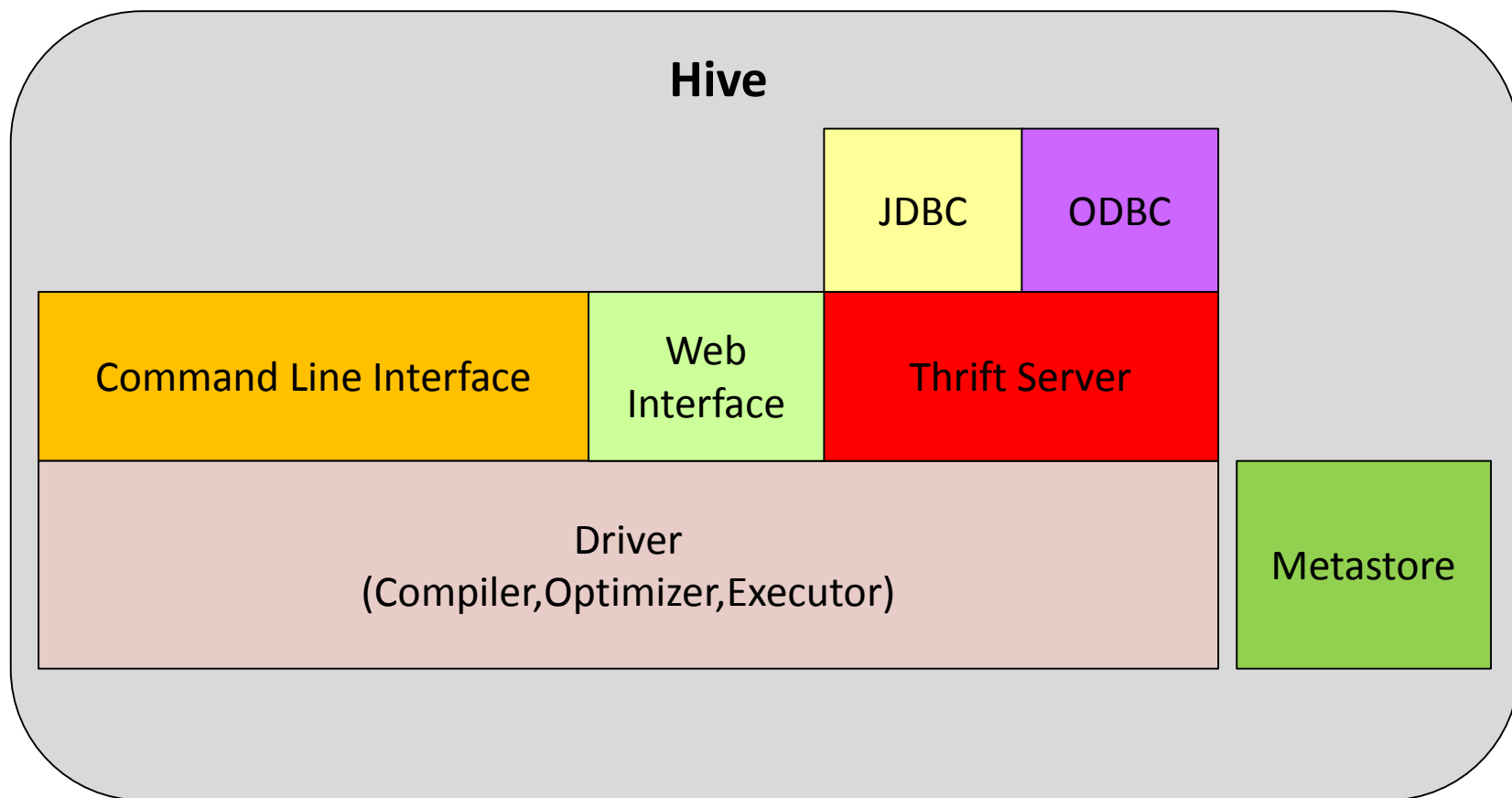


泰克教育
TECH EDUCATION

Hive的架构



Hive的架构



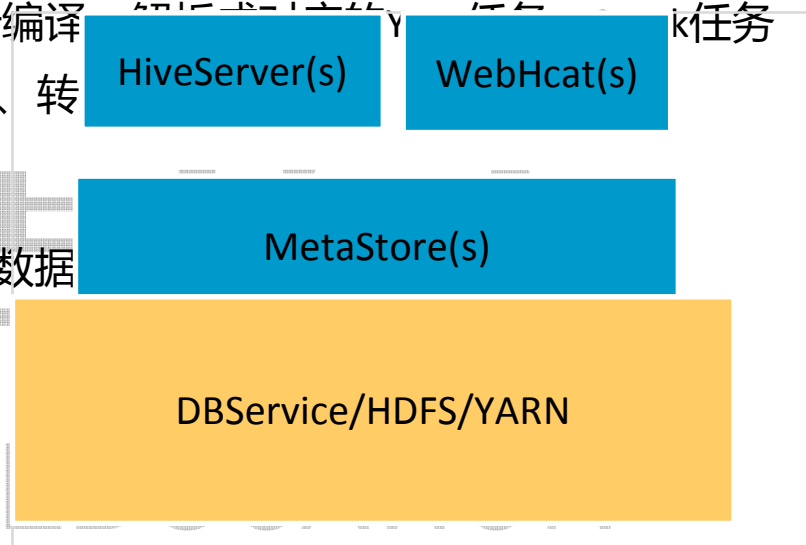
FusionInsight HD中Hive的架构

- Hive分为三个角色HiveServer、MetaStore、WebHcat。

- HiveServer 将用户提交的HQL语句进行编译或者HDFS操作，从而完成数据的提取、转
- MetaStore 提供元数据服务。
- WebHcat 对外提供基于https协议的元数据

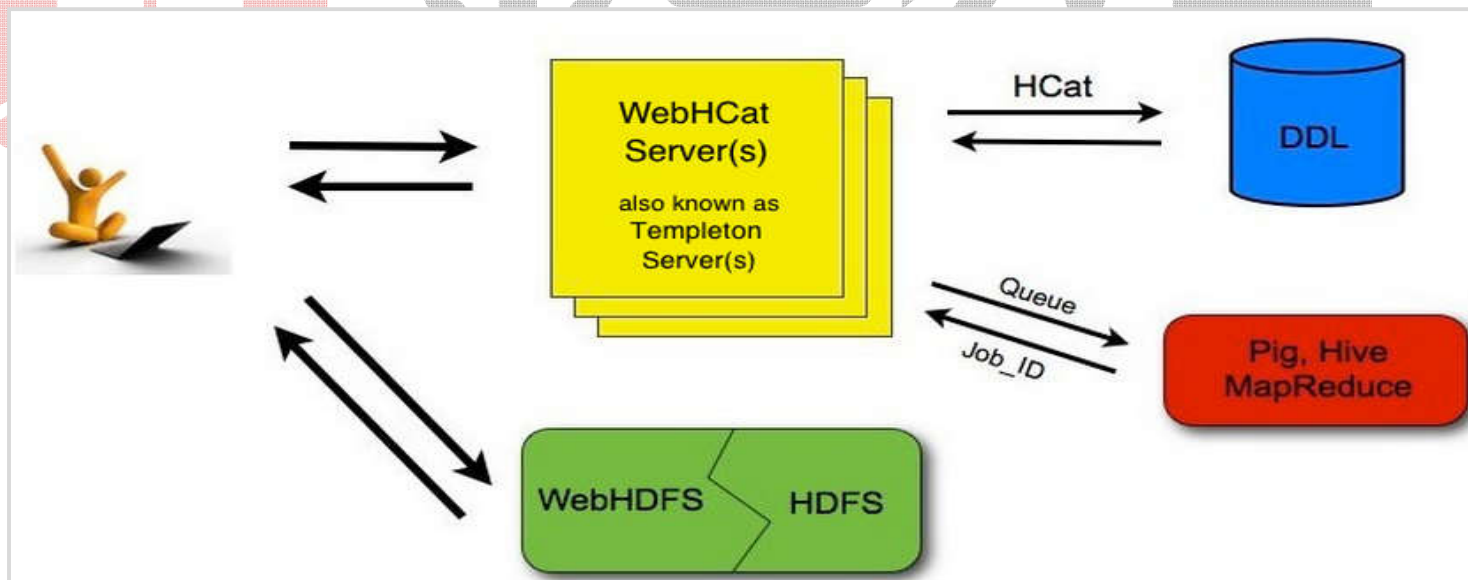


泰普
TECH

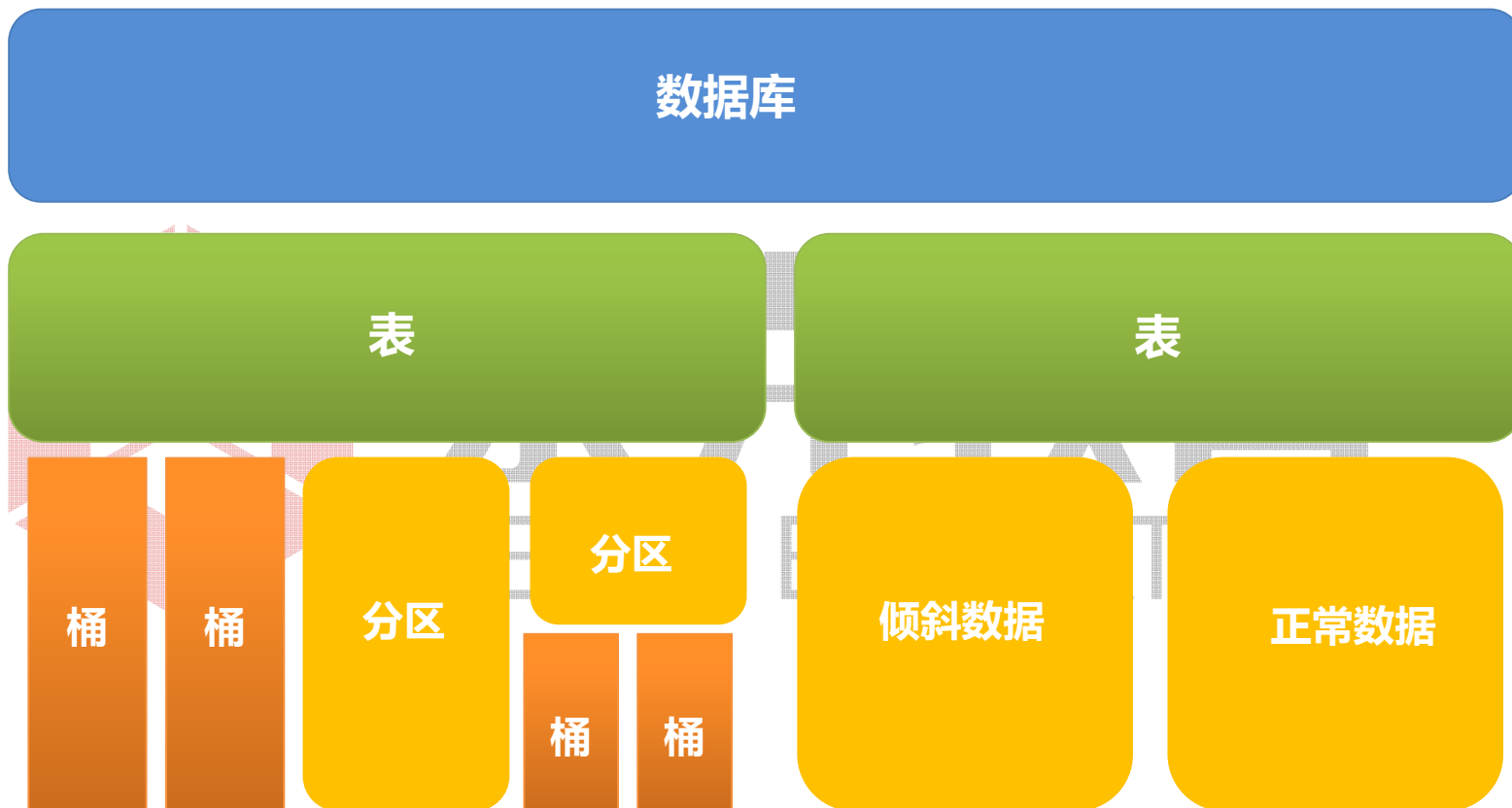


WebHCat架构

- WebHCat提供Rest接口，使用户能够通过安全的HTTPS协议执行以下操作：
 - 执行Hive DDL操作。
 - 运行Hive HQL任务。
 - 运行MapReduce任务。



Hive数据存储模型

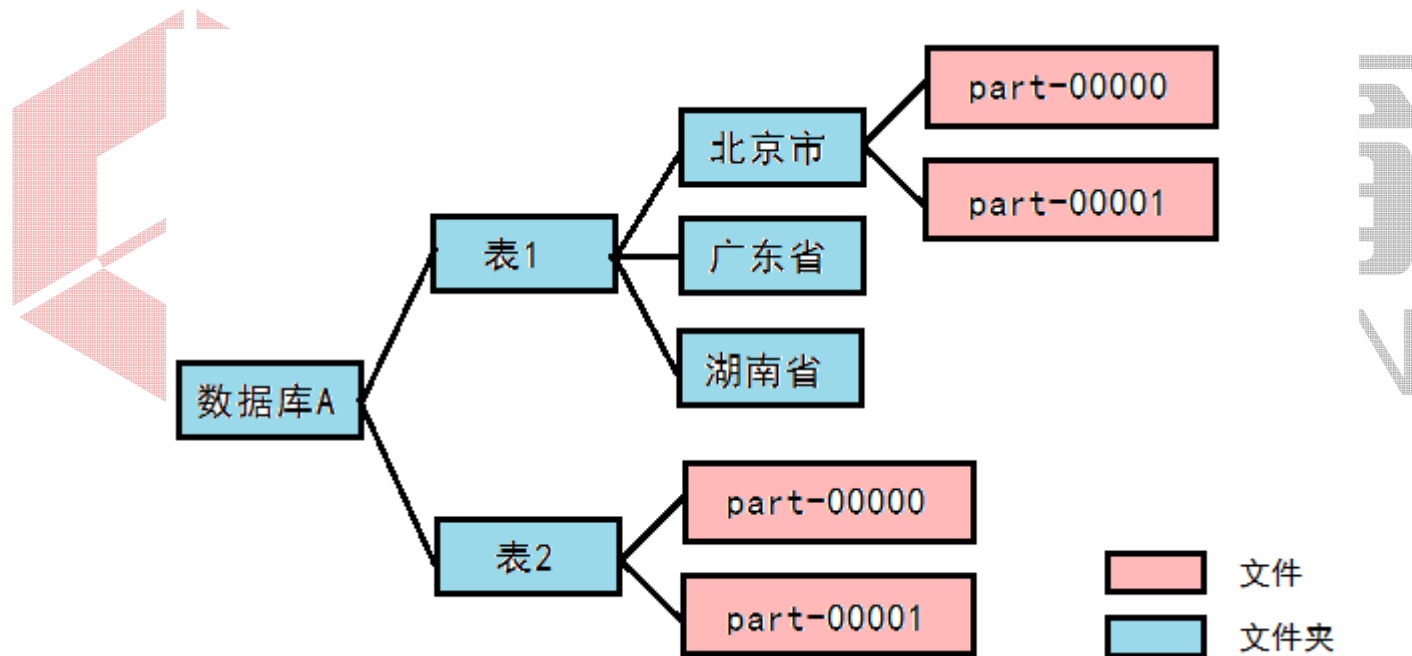


Hive数据存储模型 - 分区和桶

- 分区：数据表可以按照某个字段的值划分分区。
 - 每个分区是一个目录。
 - 分区数量不固定。
 - 分区下可再有分区或者桶。
- 桶：数据可以根据桶的方式将不同数据放入不同的桶中。
 - 每个桶是一个文件。
 - 建表时指定桶个数，桶内可排序。
 - 数据按照某个字段的值Hash后放入某个桶中。

Hive数据存储模型 - 分区和分桶

在Hive中，一个Partition对应于表下的一个目录，所有的Partition的数据都存储在对应的目录中；Buckets对指定列计算hash，根据hash值切分数据，目的是为了并行，每个Bucket对应一个文件。



Hive数据存储模型-托管表和外部表

- Hive 默认创建托管表，由Hive来管理数据，意味着Hive会将数据移动到数据仓库目录。
- 另外一种选择是创建外部表，这时Hive会到仓库目录以外的位置访问数据。

如何选择？

- 如果所有处理都由Hive完成，建议使用**托管表**；
- 如果要用Hive和其它工具来处理同一个数据集，应该使用**外部表**。

	托管表	外部表
CREATE/LOAD	把数据移到仓库目录	创建表时指明外部数据的位置
DROP	元数据和数据会被一起删除	只删除元数据

Hive数据存储模型 – 托管表和外部表

- Hive可以创建托管表和外部表：
 - 默认创建托管表，Hive会将数据移动到数据仓库目录。
 - 创建外部表，这时Hive会到仓库目录以外的位置访问数据。
 - 如果所有处理都由Hive完成，建议使用托管表。
 - 如果要用Hive和其它工具来处理同一个数据集，建议使用外部表。

	托管表	外部表
CREATE/LOAD	数据移到仓库目录	数据位置不移动
DROP	元数据和数据会被一起删除	只删除元数据

Hive数据存储模型-托管表和外部表

- 实际数据存放位置

/user/hive/warehouse/文件夹

外部文件夹

- 删除表与删除数据

同时删除元数据和数据

只删除元数据

- Table创建和数据加载

创建表格的同时可以加载数据

创建表格的同时可以指定数据路径

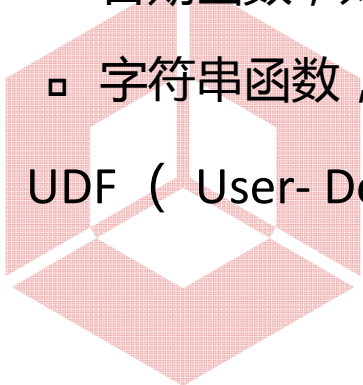
- 关键词

没有关键词

关键词external

Hive支持的函数

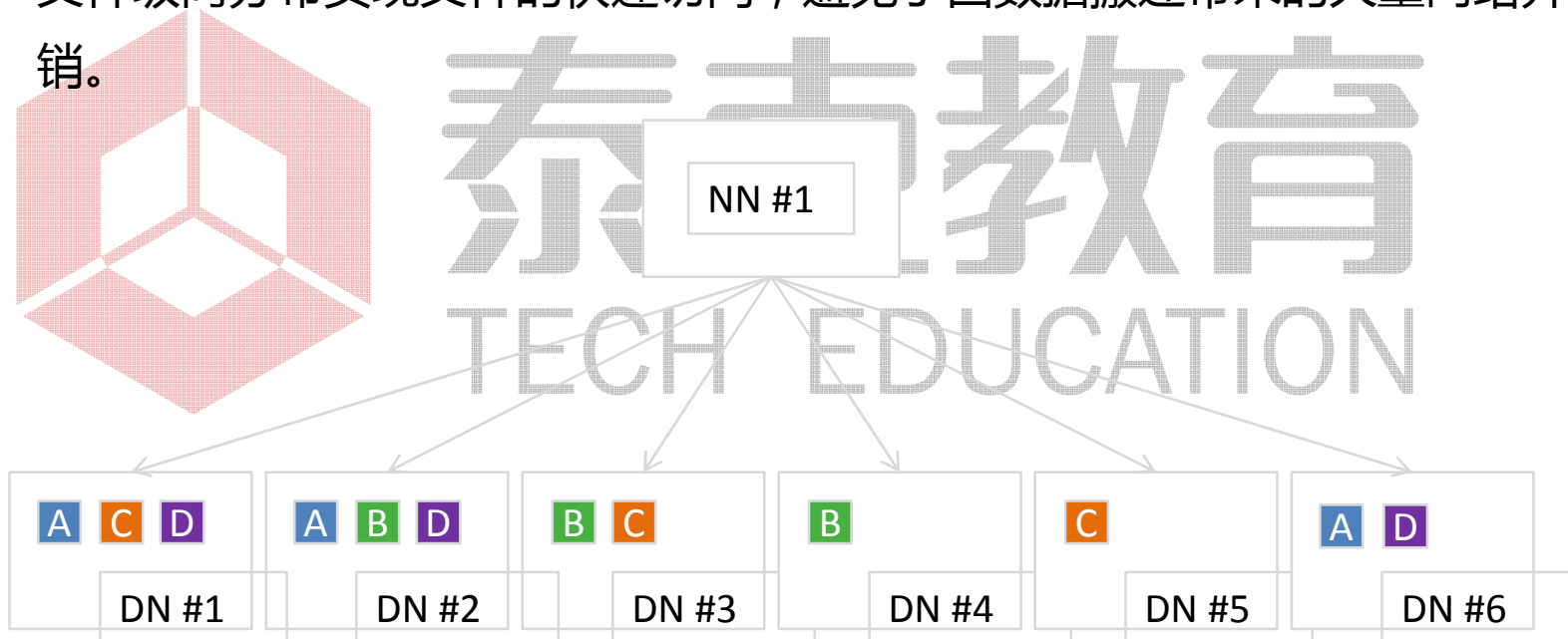
- Hive内置函数:
 - 数学函数, 如round(),floor(),abs(),rand()等。
 - 日期函数, 如to_date(),month(),day()等。
 - 字符串函数, 如trim(),length(),substr()等。
- UDF (User- Defined Funcation)。



泰克教育
TECH EDUCATION

Hive增强特性 - Colocation简介

- Colocation (同分布)：将存在关联关系的数据或可能要进行关联操作的数据存储在相同的存储节点上。
- 文件级同分布实现文件的快速访问，避免了因数据搬迁带来的大量网络开销。



Hive增强特性 - Colocation使用

- 步骤1：通过HDFS接口创建groupid和locatorid

```
hdfs colocationadmin -createGroup -groupId groupId  
-locatorIds locatorid1,locatorid2,locatorid3
```

- 步骤2：在Hive中使用Colocation

```
CREATE TABLE tbl_1 (id INT, name STRING) stored as  
RCFILE  
TBLPROPERTIES ("groupId"="group1", "locatorId"="loca  
tor1");
```

```
CREATE TABLE tbl_2 (id INT, name STRING) row  
format delimited fields terminated by '\t' stored  
as TEXTFILE  
TBLPROPERTIES ("groupId"="group1", "locatorId"="loca  
tor1");
```


Hive增强特性 - 列加密

- 步骤1：在创建表时指定相应的加密列和加密算法

```
create table encode_test (id INT, name STRING, phone  
STRING, address STRING) row format serde  
"org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe"  
WITH SERDEPROPERTIES (  
    "column.encode.columns"="phone,address", "column.encode.  
    classname"="org.apache.hadoop.hive.serde2.AESRewriter"  
);
```

- 步骤2：使用insert语法向设置列加密的表中导入数据

```
insert into table encode_test select id, name,  
phone, address from test;
```

Hive增强特性 - HBase记录批量删除

- 概要说明：

- 在Hive on HBase功能中，FusionInsight HD Hive提供了对HBase表的单条数据的删除功能，通过特定的语法，Hive可以将HBase表中符合条件的一条或者多条数据批量清除。

- 使用：

- 如果要删除某张HBase表中的某些数据，可以执行HQL语句：

```
remove table HBase_table where expression;
```

其中expression规定要删除数据的筛选条件。

Hive增强特性 - 流控特性

- 通过流控特性，可以实现：
 - 当前已经建立的总连接数阈值控制。
 - 每个用户已经建立的连接数阈值控制。
 - 单位时间内所建立的连接数阈值控制。

Parameter	Value
hive.default.fileformat	<input type="radio"/> TextFile <input type="radio"/> SequenceFile <input checked="" type="radio"/> RCfile <input type="radio"/> ORC
hive.exec.compress.output	<input type="radio"/> true <input checked="" type="radio"/> false
hive.exec.reducers.max	999
hive.mapred.mode	<input checked="" type="radio"/> nonstrict <input type="radio"/> strict
hive.server.session.control.maxconnection.peruser	500
hive.server.session.control.maxconnections	500
hive.server.timewindow.delaytime	60
hive.server.timewindow.maxsessions.in.delaytime	500
hive.server2.idle.session.timeout	4320m
hive.server2.session.check.interval	3000ms

Hive增强特性 - 指定行分割符

- 步骤1：创建表时指定InputFormat和OutputFormat

```
CREATE [TEMPORARY] [EXTERNAL] TABLE [IF NOT EXISTS]
[db_name.]table_name
[(col_name data_type [COMMENT col_comment], ...)]
[ROW FORMAT row_format]
STORED AS
inputformat
"org.apache.hadoop.hive.contrib.fileformat.SpecifiedDe
limiterInputFormat"
outputformat
"org.apache.hadoop.hive ql.io.HiveIgnoreKeyTextOutputF
ormat";
```

- 步骤2：查询之前指定配置项

```
set hive.textinput.record.delimiter="!@!";
```



目录

1. Hive概述
2. Hive功能与架构
3. **Hive基本操作**



泰克教育
TECH EDUCATION

Hive SQL介绍

- DDL-数据定义语言
 - 建表，修改表，删表、分区、数据类型。
- DML-数据管理语言
 - 数据导入、数据导出。
- DQL-数据查询语言
 - 简单查询。
 - 复杂查询Group by，Order by，Join等。

泰克教育
TECH EDUCATION

Hive基本操作 (1)

- 数据格式示例：1,huawei,1000.0

--创建托管表

```
CREATE TABLE IF NOT EXISTS example.employee (  
  Id INT COMMENT 'employeeid',  
  Company STRING COMMENT 'your company',  
  Money FLOAT COMMENT 'work money',)  
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','  
STORED AS TEXTFILE;
```

--创建外部表

```
CREATE EXTERNAL TABLE IF NOT EXISTS  
example.employee (  
  Id INT COMMENT 'employeeid',  
  Company STRING COMMENT 'your company',  
  Money FLOAT COMMENT 'work money',) ROW FORMAT  
DELIMITED FIELDS TERMINATED BY ','  
STORED AS TEXTFILE LOCATION '/localtest';
```


Hive基本操作 (2)

--修改列

```
ALTER TABLE employee1 CHANGE money string COMMENT  
'changed by alter' AFTER dateincompany;
```

--添加列

```
ALTER TABLE employee1 ADD columns(column1 string);
```

--修改文件格式

```
ALTER TABLE employee3 SET fileformat TEXTFILE;
```

--删除表数据/表

```
DELETE column_1 from table_1 WHERE column_1=??;  
DROP table_a;
```

--描述表

```
DESC table_a;
```

--显示表创建语句

```
SHOW CREATE table_a;
```

Hive基本操作 (3)

--从本地文件中数据到Hive表

```
LOAD DATA LOCAL INPATH 'employee.txt' OVERWRITE INTO TABLE  
example.employee;
```

--从另一个表加载数据到Hive表

```
INSERT INTO TABLE company.person PARTITION(century= '21',year='2010')  
SELECT id, name, age, birthday FROM company.person_tmp WHERE  
century= '23' AND year='2010';
```

--导出数据到HDFS

```
EXPORT TABLE company.person TO '/department';
```

--从HDFS导入数据

```
IMPORT TABLE company.person FROM '/department';
```

--插入数据

```
INSERT INTO TABLE company.person  
SELECT id, name, age, birthday FROM company.person_tmp  
WHERE century= '23' AND year='2010';
```

Hive基本操作 (4)

--WHERE 查询

```
SELECT id, name FROM employee WHERE salary >= 10000;
```

--GROUP BY查询

```
SELECT department, avg(salary) FROM employee GROUP BY department;
```

--UNION ALL

```
SELECT id, salary, date FROM employee_a UNION ALL  
SELECT id, salary, date FROM employee_b;
```

--JOIN查询

```
SELECT a.salary, b.address FROM employee a JOIN employee_info  
b ON a.name=b.name;
```

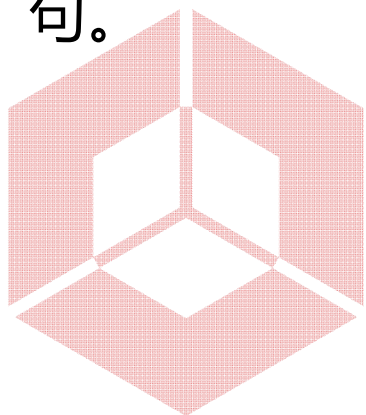
--子查询

```
SELECT a.salary, b.address FROM employee a JOIN (SELECT  
address FROM employee_info where province='zhejiang') b ON  
a.name=b.name;
```



本章总结

- 通过本章的学习，介绍了Hive应用场景与基本原理，然后介绍了Hive在FusionInsight 中的增强特性以及常用的Hive SQL语句。



泰克教育
TECH EDUCATION

思考题

1. 以下哪些是Hive适用的场景？（ ）
 - A. 实时的在线数据分析
 - B. 数据挖掘（用户行为分析，兴趣分区，区域展示）
 - C. 数据汇总（每天/每周用户点击数，点击排行）
 - D. 非实时分析（日志分析，统计分析）

泰克教育
TECH EDUCATION

思考题

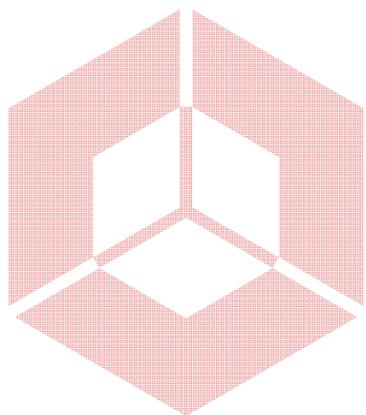
1. 以下关于Hive SQL基本操作描述正确的是？（ ）
 - A. 创建外部表使用external关键字，创建普通表需要指定internal关键字
 - B. 创建外部表必须要指定location信息
 - C. 加载数据到Hive时源数据必须是HDFS的一个路径
 - D. 创建表时可以指定列分隔符

泰克教育
TECH EDUCATION



更多信息

- 下载培训资料：
 - <http://support.huawei.com/learning/trainFaceDetailAction?lang=zh&pbiPath=term1000025185&courseId=Node1000009072>
- eLearning课程：
 - <http://support.huawei.com/learning/nodeQueryAction!loadTrainProjectInfo?lang=zh&pbiPath=term1000025185&courseId=Node1000009421&navId=MW000001>
- 考试大纲：
 - <http://support.huawei.com/learning/Certificate!toExamOutlineDetail?lang=zh&nodeId=Node1000003516>
- 模拟考试：
 - <http://support.huawei.com/learning/Certificate!toSimExamDetail?lang=zh&nodeId=Node1000004285>
- 认证流程：
 - [http://support.huawei.com/learning/NavigationAction!createNavi#navi\[id\]=_40](http://support.huawei.com/learning/NavigationAction!createNavi#navi[id]=_40)



谢谢

www.huawei.com

泰克教育
TECH EDUCATION