Hive高级编程

天照

Agenda

- Hive Components
- MapReduce
- Hive QL
- Hive 优化
- SQL 优化

HIVE: Components

Web UI

Hive CLI

Browsing DDL Queries

Thrift API

MetaStore

DB

Parser

Planner

Optimizer

Execution

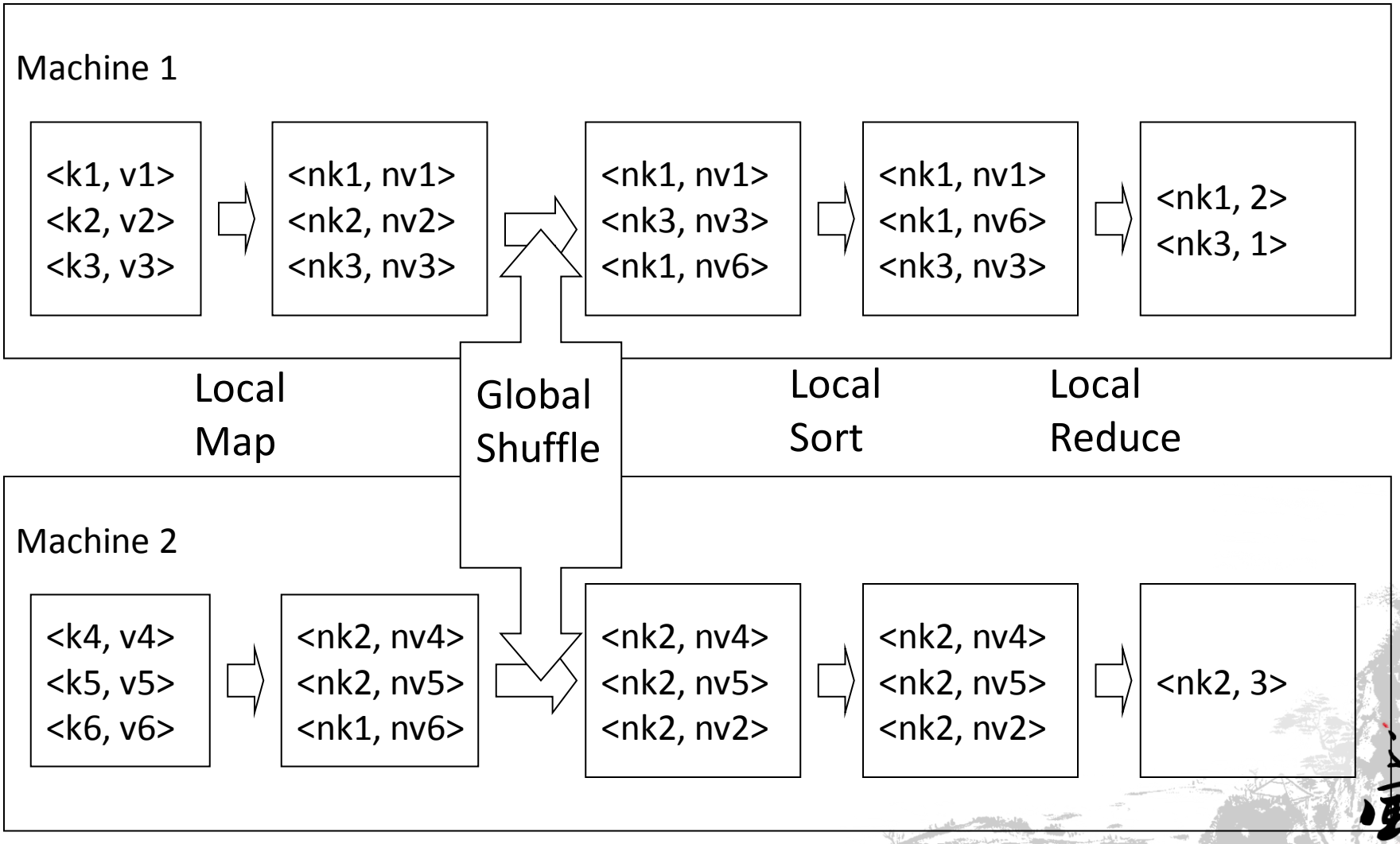
SerDe

Thrift CSV JSON..

Map Reduce

HDFS

(Simplified) Map Reduce Review



Hive QL – Join

page_view

pageid	**userid**	time
1	**111**	9:08:01
2	**111**	9:08:13
1	**222**	9:08:14

X

user

userid	age	gender
111	25	female
222	32	male

=

pv_users

pageid	age
1	25
2	25
1	32

- **SQL:**

```
INSERT INTO TABLE pv_users

SELECT pv.pageid, u.age

FROM page_view pv JOIN user u ON (pv.userid = u.userid);
```

Hive QL – Join in Map Reduce

page_view

pageid	**userid**	time
1	**111**	9:08:01
2	**111**	9:08:13
1	**222**	9:08:14

user

userid	age	gender
111	25	female
222	32	male

Map

key	value
111	<**1**,1>
111	<**1**,2>
222	<**1**,1>

key	value
111	<**2**,25>
222	<**2**,32>

Shuffle
Sort

key	value
111	<**1**,1>
111	<**1**,2>
111	<**2**,25>

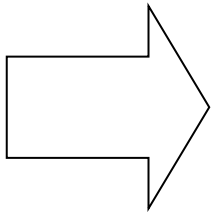
key	value
222	<**1**,1>
222	<**2**,32>

Reduce

Hive QL – Group By

pv_users

pageid	age
1	25
2	25
1	32
2	25

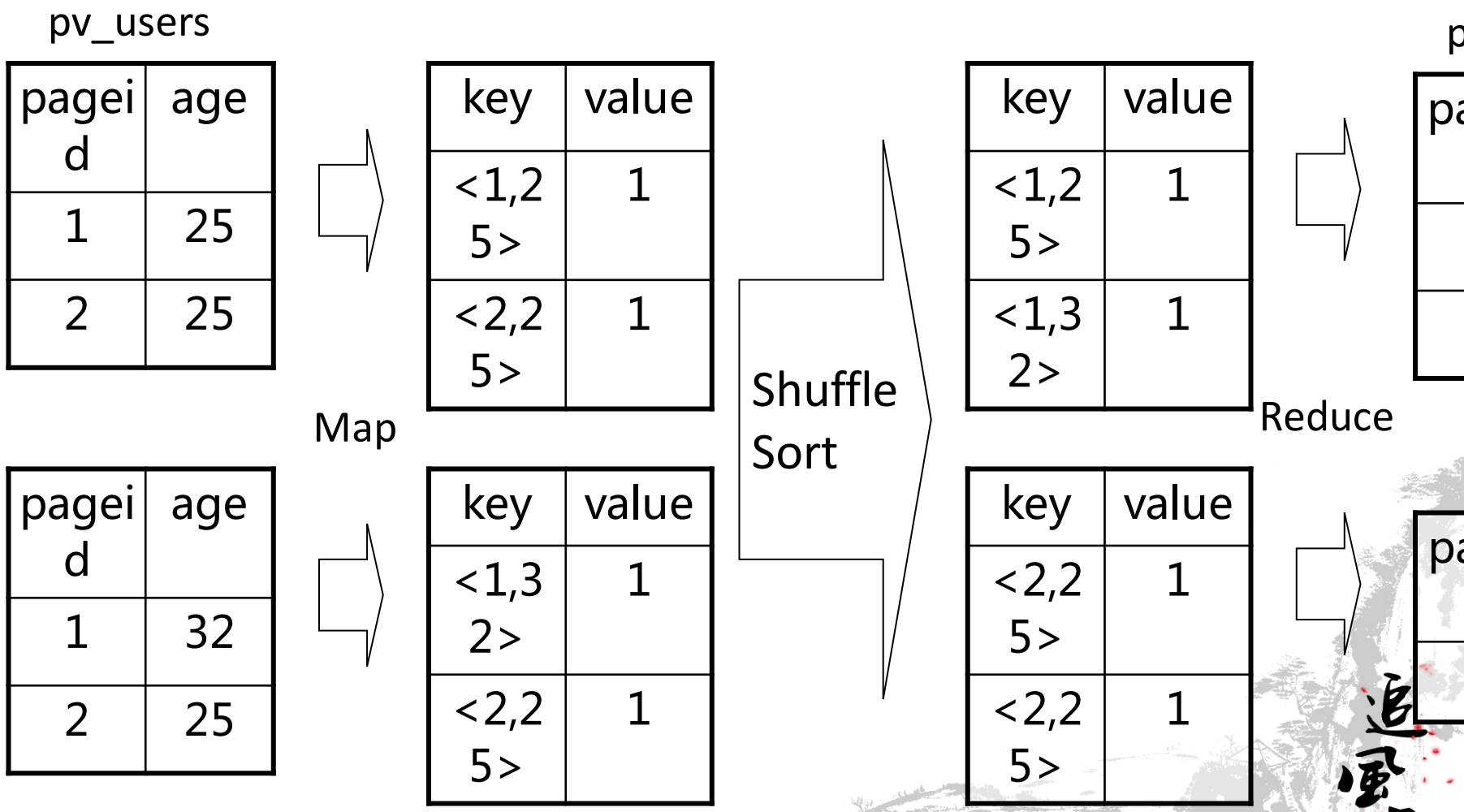


pageid_age_sum

pageid	age	Count
1	25	1
2	25	2
1	32	1

- **SQL:**
  - INSERT INTO TABLE pageid_age_sum
  - SELECT pageid, age, count(1)
  - FROM pv_users
  - GROUP BY pageid, age;

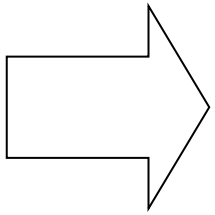
Hive QL – Group By in Map Reduce



Hive QL – Group By with Distinct

page_view

pageid	userid	time
1	111	9:08:01
2	111	9:08:13
1	222	9:08:14
2	111	9:08:20



result

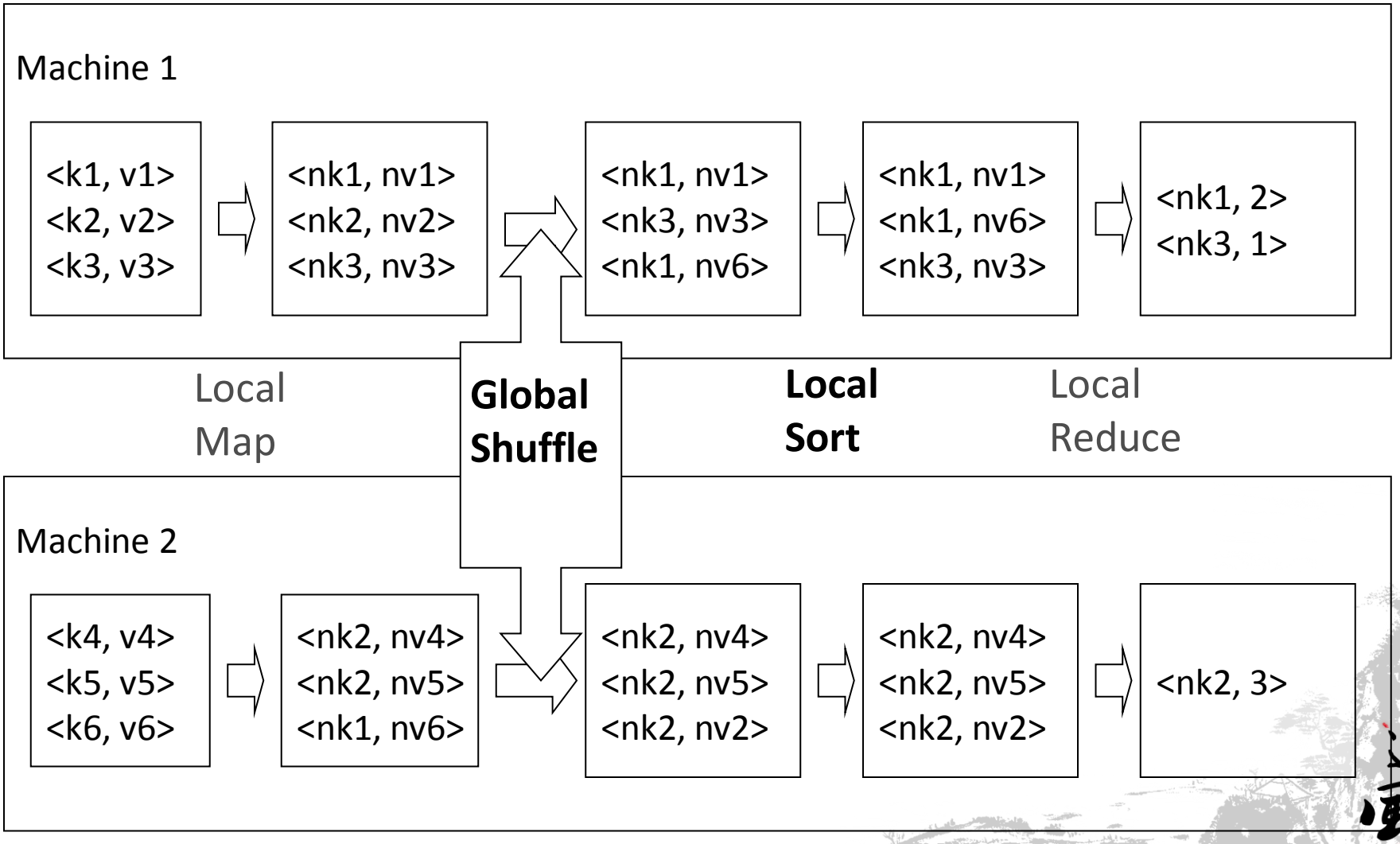
pageid	count_distinct_userid
1	2
2	1

- **SQL**
  - SELECT pageid, COUNT(DISTINCT userid)
  - FROM page_view GROUP BY pageid

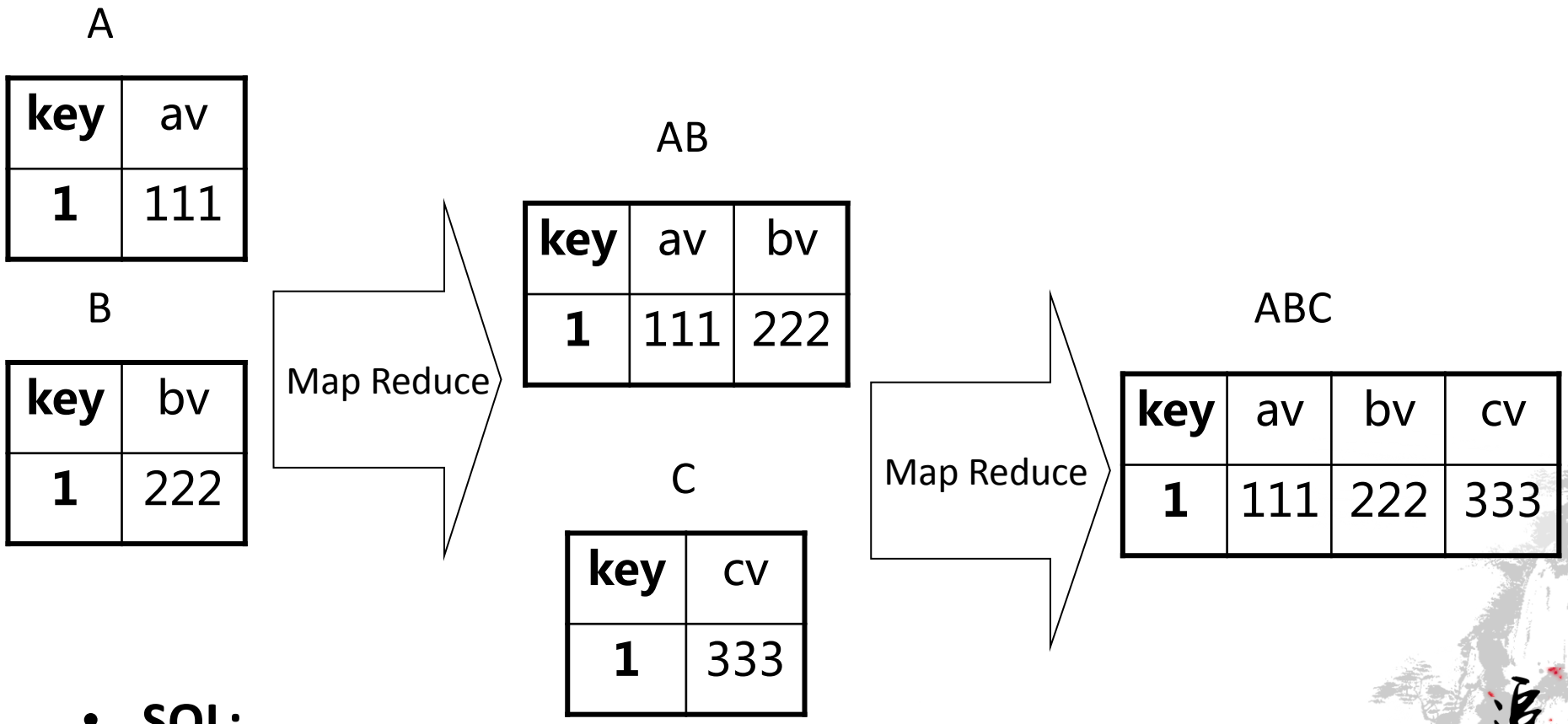
Hive Optimizations

Efficient execution of SQL on Map Reduce

(Simplified) Map Reduce Revisit

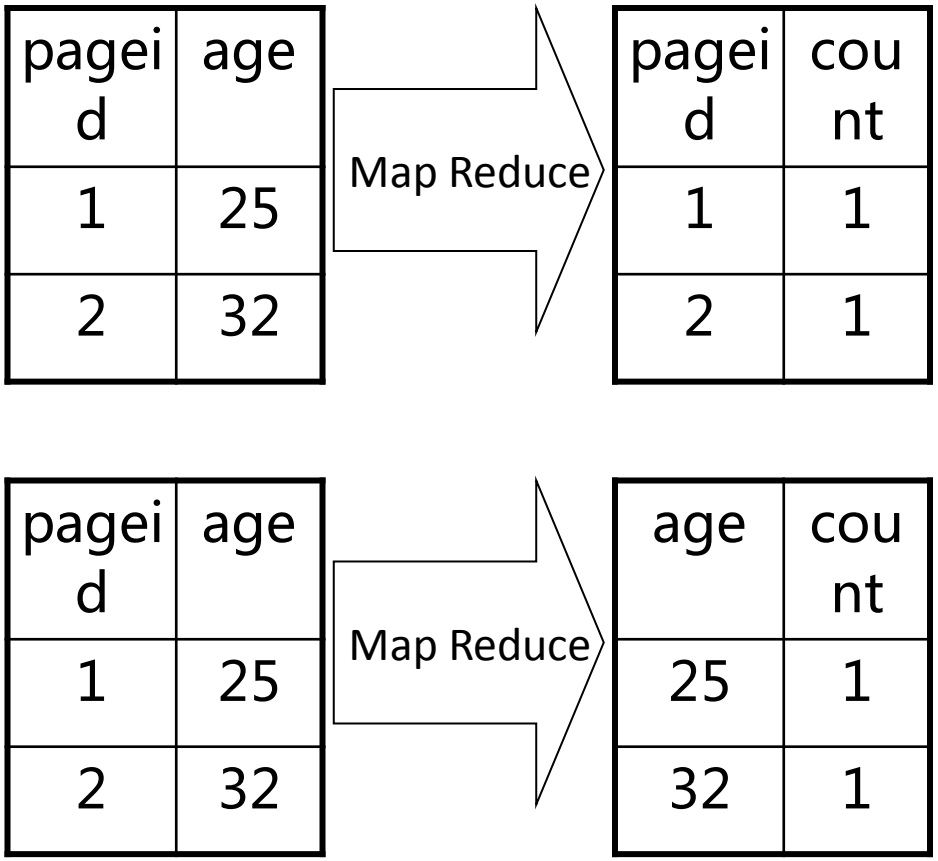


Hive Optimizations – Merge Sequential Map Reduce Jobs



- **SQL:**
  - FROM (a join b on a.key = b.key) join c on a.key = c.key SELECT …

Hive Optimizations – Share Common Read Operations



- **Extended SQL**
  - **FROM pv_users**
  - **INSERT** INTO TABLE pv_pageid_sum
    - SELECT pageid, count(1)
    - GROUP BY pageid
  - **INSERT** INTO TABLE pv_age_sum
    - SELECT age, count(1)
    - GROUP BY age;
- **But，不能太多Multi Insert**

Hive Optimizations –Map Join

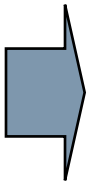
- **Map Joins**
  - User specified small tables stored in hash tables on the mapper
  - No reducer needed

```
INSERT INTO TABLE pv_users
SELECT /*+ MAPJOIN(pv) */ pv.pageid, u.age
FROM page_view pv JOIN user u
ON (pv.userid = u.userid);
```

Hive QL – Map Join

page_view

pageid	**userid**	time
1	**111**	9:08:01
2	**111**	9:08:13
1	**222**	9:08:14



Hash table

key	value
111	<1,2>
222	<2>

user

userid	age	gender
111	25	female
222	32	male

pv_users

Pageid	age
1	25
2	25
1	32

Group by Optimizations

- **Map side partial aggregations**
  - Hash-based aggregates
  - Serialized key/values in hash tables
  - 90% speed improvement on Query
    - `SELECT count(1) FROM t;`

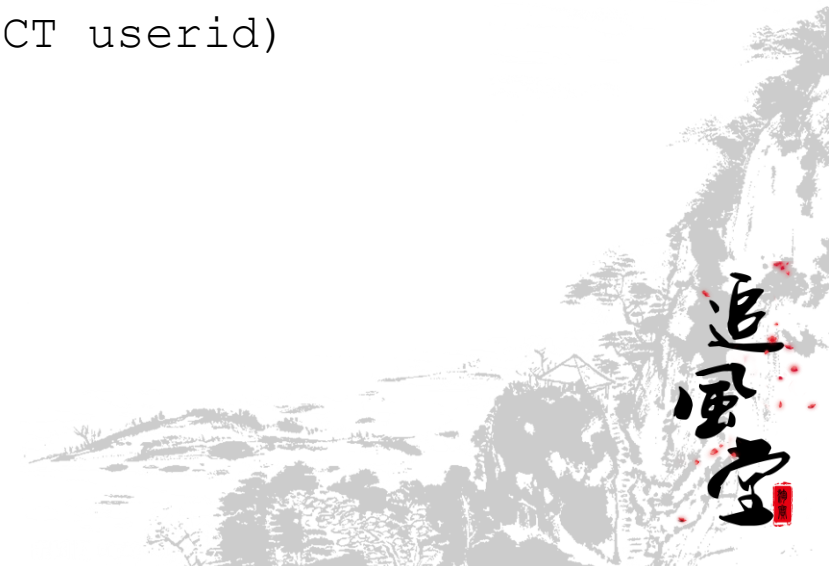
Parameters

- hive.map.aggr = true
- hive.groupby.skewindata = false
- hive.groupby.mapaggr.checkinterval = 100000 （检测频率）
- hive.map.aggr.hash.min.reduction = 0.5（最少的聚合效果）
- hive.map.aggr.hash.percentmemory = 0.5（map端聚合最多能使用的内存）

Multi GroupBy

```
FROM pv_users
  INSERT OVERWRITE TABLE pv_gender_sum
    SELECT gender, count(DISTINCT userid),
      count(userid)
      GROUP BY gender
  INSERT OVERWRITE TABLE pv_age_sum
    SELECT age, count(DISTINCT userid)
      GROUP BY age
```

Attention：不能太多



Hive QL – Group By in Map Reduce

pv_users

gender	age	userid
M	25	1
M	25	2
M	25	1
M	24	1
F	24	2
F	24	1

Key: userid
Value: gender, age

gender	dist	count
M	1	3
F	1	1

age	dist
25	1
24	1

gender	dist	count
M	1	1
F	1	1

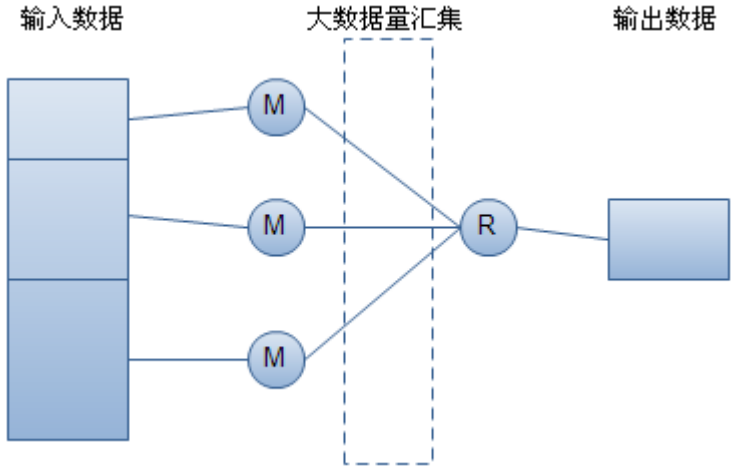
age	dist
25	1
24	1

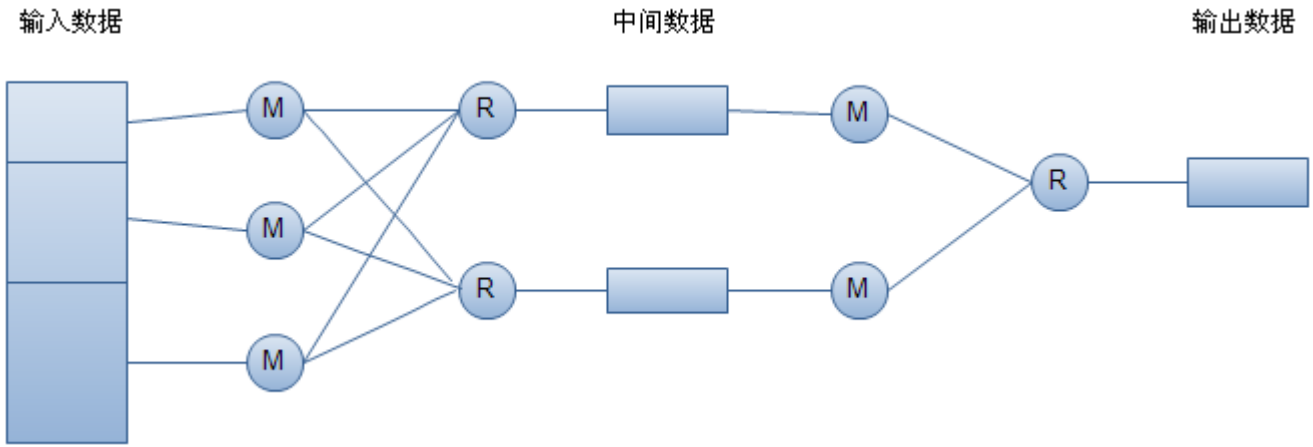
gender	dist	count
M	2	4
F	2	2

age	dist
24	1
25	1

Load balancing for data skew

- **GroupBy数据倾斜**
  - skewindata优化
  - 用法
    - set hive.groupby.skewindata=true





Hive Optimizations – Load Balance Problem

pv_users

pageid	age
1	25
1	25
1	25
2	32
1	25

Map-Reduce

pageid	age	count
1	25	2
2	32	1
1	25	2

SQL优化-find and control

- 数据倾斜
- Join顺序
- Map only
- UDF
- Others

- **数据倾斜**
  - 倾斜的原因？
    - group by/distinct

    1. set hive.groupby.skewindata=true

    - Join

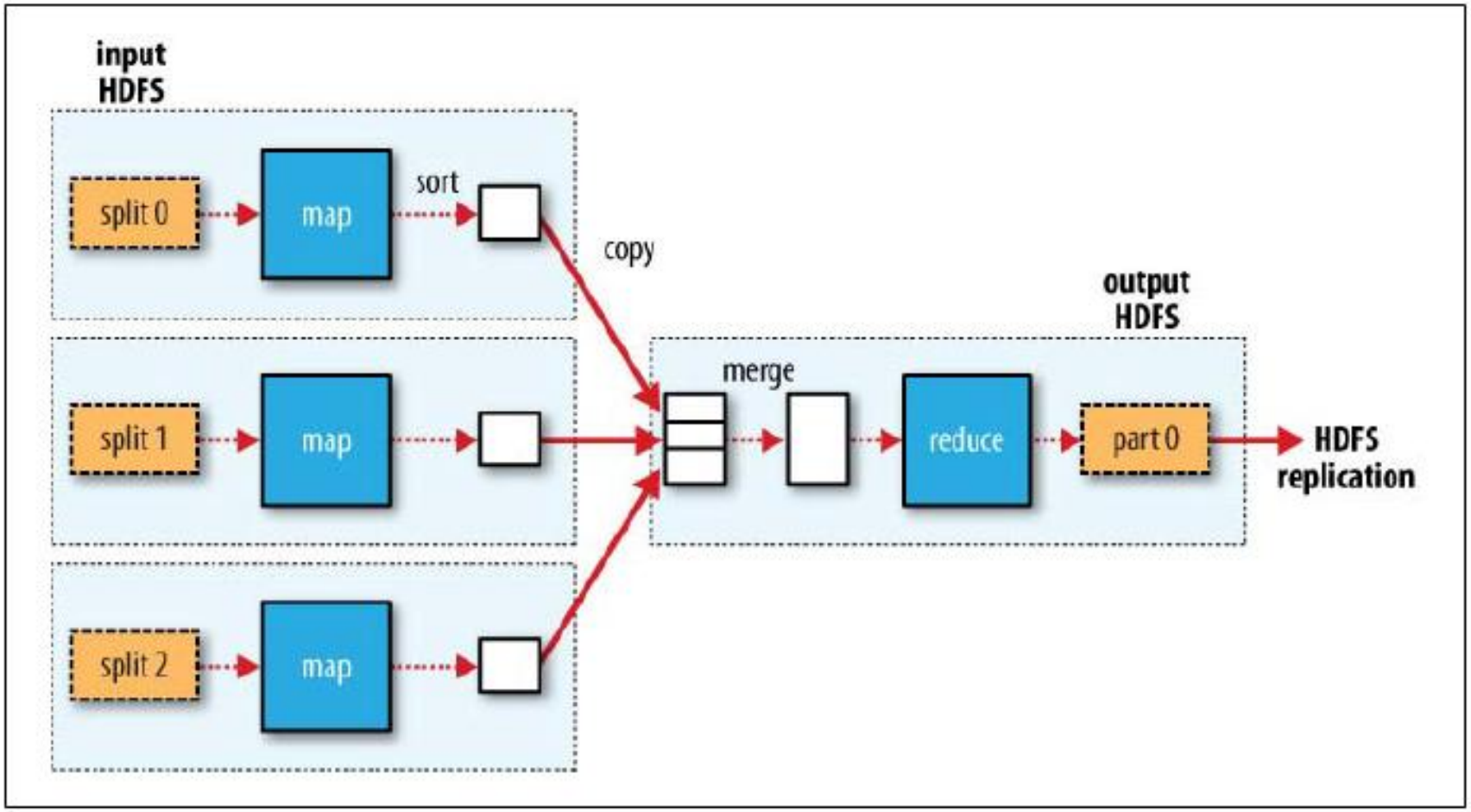
    1. mapjoin
    2. 业务层面解决

    http://baike.corp.taobao.com/index.php/Hive%E4%BC%98%E5%8C%96%E6%A1%88%E4%BE%8B

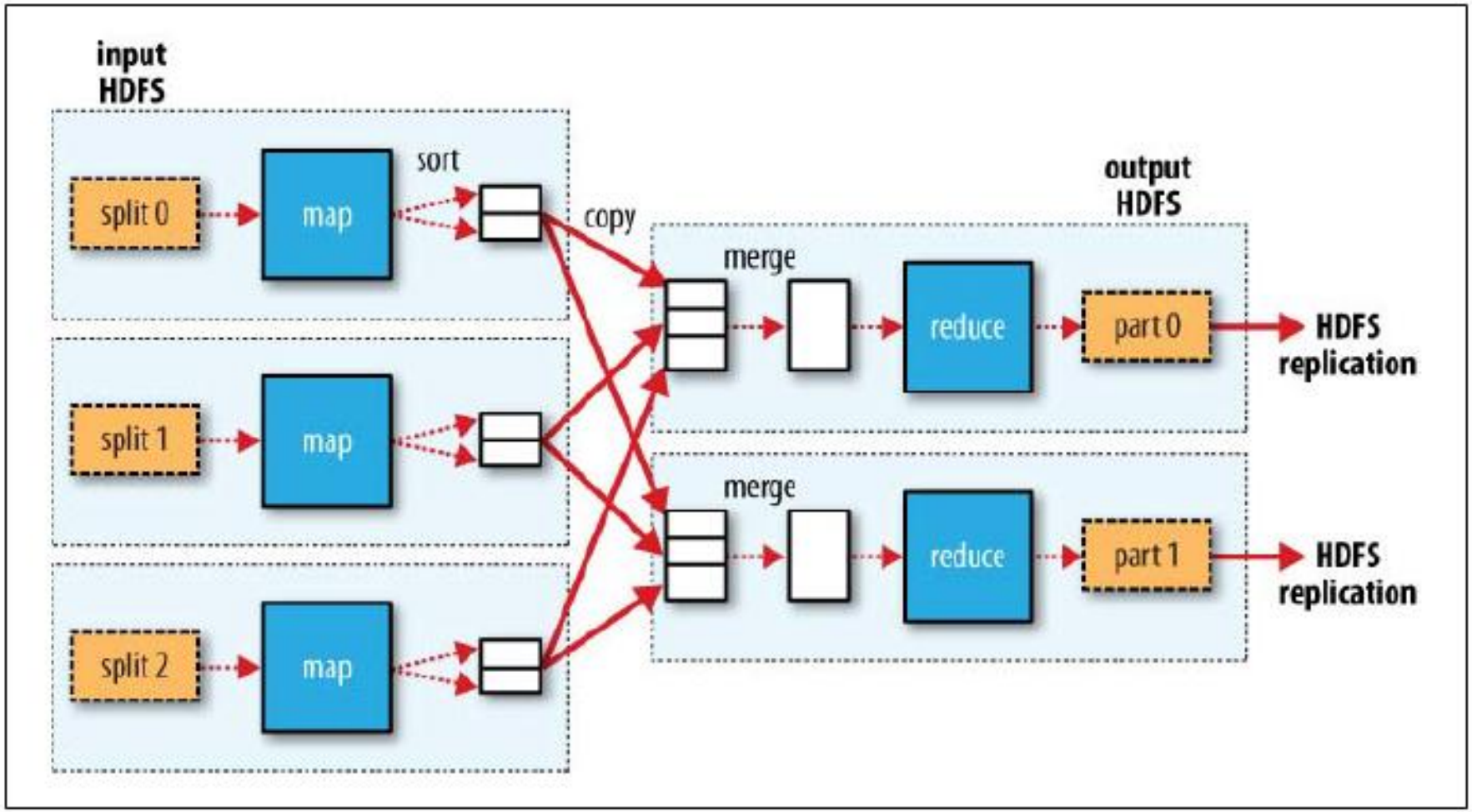
Join 顺序

- **内存优化**
  - 驱动表
    - 使用大表做驱动表，避免内存溢出
    - Join中最右边的表是驱动表
    - MapJoin无视Join顺序，使用大表做驱动表
    - STREAMTABLE

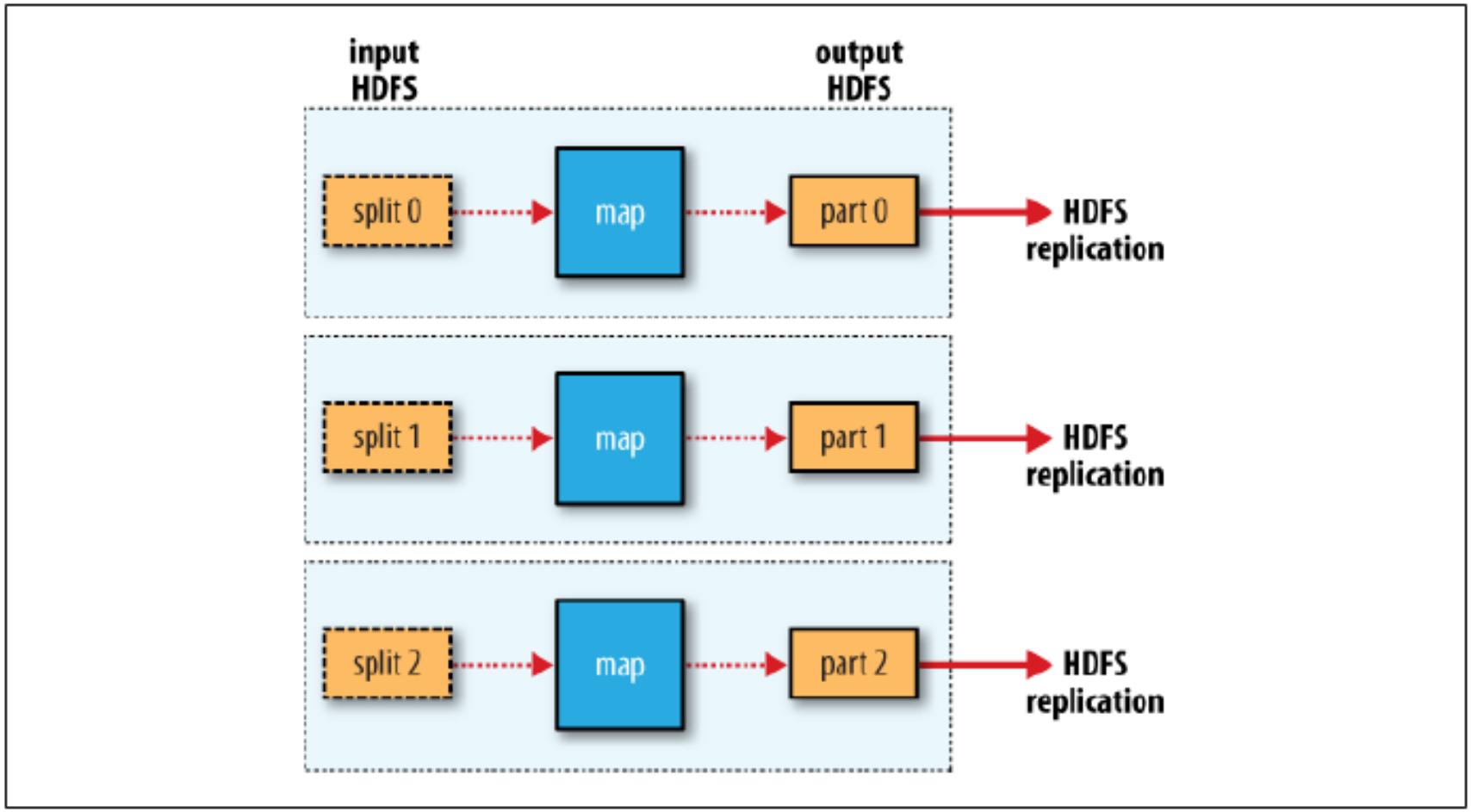
Map only


input
HDFS
split 0
map
sort
split 1
map
split 2
map
copy
output
HDFS
merge
reduce
part 0
HDFS
replication

Map only


input
HDFS
split 0
map
sort
split 1
map
split 2
map
copy
merge
reduce
part 0
output
HDFS
HDFS
replication
merge
reduce
part 1
HDFS
replication

Map only


input
HDFS
output
HDFS
split 0
map
part 0
HDFS
replication
split 1
map
part 1
HDFS
replication
split 2
map
part 2
HDFS
replication

Map Only

- **Map only**
  - 特征
    - 没有Join、 GroupBy、 Order by、 Sortby等，导致无Reduce
    - 每个Map有一个输出文件，输入数据量大，Map数很多导致输出文件很多
  - 缺点
    - 依赖此job输出的下一个job，map数很大
    - Fetch 结果很慢

Map Only

- **Map only**
  - 改进前
    - select return_fee from r_crm_refund_trade

      where pt='20111123000000' AND dateCompare(gmt_created,'2011-11-23',0)=0 ;
  - 改进后
    - select return_fee from r_crm_refund_trade

      where pt='20111123000000' AND dateCompare(gmt_created,'2011-11-23',0)=0

      distribute by rand(12345) ;

    http://baike.corp.taobao.com/index.php/Hive%E4%BC%98%E5%8C%96%E6%A1%88%E4%BE%8B#map_only.E7.9A.84.E4.BD.9C.E4.B8.9A

– 优先官方UDF

- https://cwiki.apache.org/confluence/display/Hive/LanguageManual+UDF

- 耗时的操作，耗cpu
  - 正则，优先like，后rlike
  - 编解码，url 的*encode/decode*
  - Text复用，new Text(String s) *encode String*->bytes(UTF-8)，toString *decode* bytes(UTF-8) -> *String*
  - for 循环中String+String（gc）-> StringBuilder
  - SimpleDateFormat 复用，构造函数中的操作耗时
  - ……

UDF

- date_format yyyy-MM-dd HH:mm:ss -> yyyy-MM-dd 慢

```
public class UDFDate extends UDF {
 private Text t = new Text();
  public UDFDate() {
  }
  public Text evaluate(Text dateString) {
    if (dateString == null) {  return null; }
    try {
      SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
      Date date = formatter.parse(dateString.toString());
      t.set(formatter.format(date));
      return t;
    } catch (ParseException e) {
      return null;
    }
  }
}
```

UDF

- date_format　yyyy-MM-dd HH:mm:ss -> yyyy-MM-dd 快

```
public class UDFDate extends UDF {
 private final SimpleDateFormat formatter = new SimpleDateFormat("yyyy-MM-dd");
 private Text t = new Text();
 public UDFDate() {
 }
 public Text evaluate(Text dateString) {
  if (dateString == null) {  return null; }
  try {
   Date date = formatter.parse(dateString.toString());
   t.set(formatter.format(date));
   return t;
  } catch (ParseException e) {
   return null;
  }
 }
}
```

Others - dynamic.partition

- **改进前**

```
insert OVERWRITE TABLE r_mid_alipay_play partition (pt)
select  pt as day_id, buyer_nick, buyer_id, cat1, name1,
        concat(pt,'000000') as pt
from   r_gmv_alipay a
where pt>='20110101' and alipay = 'alipay'
group by  pt, buyer_nick, buyer_id, cat1, name1,
         concat(pt,'000000')
;
```

Others - dynamic.partition

- **改进后**

```
insert OVERWRITE TABLE r_mid_alipay_play partition (pt)
select day_id, buyer_nick, buyer_id, cat1, name1,
    cat11, cat22, pt
from (
   select pt as day_id, buyer_nick , buyer_id , cat1 , name1 ,
       concat(pt,'000000') as pt
   from r_gmv_alipay a
   where pt>='20110101' and alipay = 'alipay'
   group by pt, buyer_nick , buyer_id , cat1 , name1 ,
      concat(pt,'000000')
)
distribute by pt
sort by pt
;
```

http://baike.corp.taobao.com/index.php/Hive_sql_%E7%9B%B8%E5%85%B3%E7%94%A8%E6%B3%95#.E5.8A.A8.E6.80.81.E5.88.86.E5.8C.BA.E4.BA.A7.E7.94.9F.E5.A4.A7.E9.87.8F.E6.96.87.E4.BB.B6.EF.BC.8C.E5.AF.BC.E8.87.B4namenode_load.E5.BE.88.E9.AB.98

Others - 如何快速取出现次数多的前几个id

- 慢
  - select auction_id,count(1) as num
    from r_auction_auctions_20110107_p
    where auction_id <> 0 and auction_id is not null
    group by auction_id
    order by num desc
    limit 100
- 快：
  - select auction_id, num
    from (select auction_id,count(1) as num
    from r_auction_auctions_20110107_p
    where auction_id <> 0 and auction_id is not null
    group by auction_id
)subqq
    where num > 100
    order by num desc
    limit 100;

http://baike.corp.taobao.com/index.php/Hive_sql_%E7%9B%B8%E5%85%B3%E7%94%A8%E6%B3%95#.E5.A6.82.E4.BD.95.E5.BF.AB.E9.80.9F.E5.8F.96.E5.87.BA.E7.8E.B0.E6.AC.A1.E6.95.B0.E5.A4.9A.E7.9A.84.E5.89.8D.E5.87.A0.E4.B8.AAid

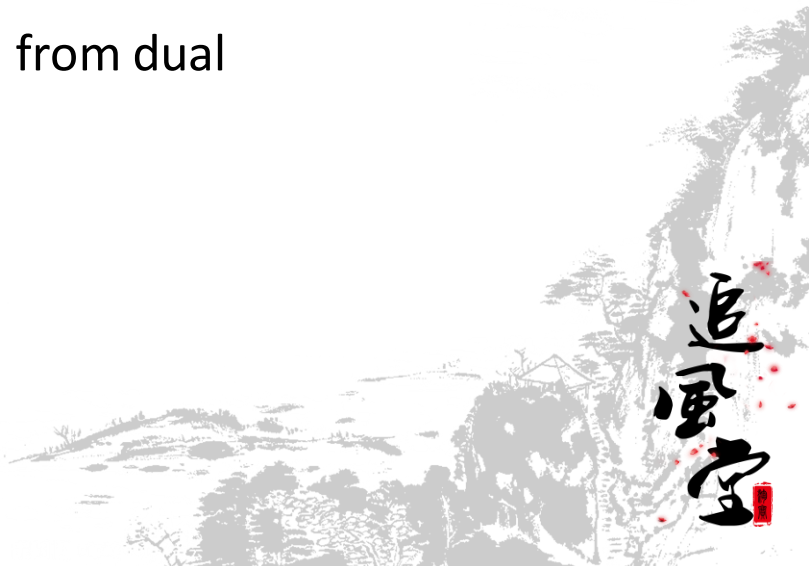
Others -通过复合结构来优化udf的调用

- 慢
  - select split("accba", "b")[0],split("accba", "b")[1] from dual;
- 快：
  - select a[0],a[1]

    from (

    select split("accba", "b") as a from dual

)subqq;



作业运行长的原因

- （1）数据量，文件数
- （2）SQL长，处理逻辑多
- （3）计算处理耗时
- （4）倾斜
- （5）RPC调用
- （6）参数设置不合理
- （7）调度
- ……

参考资料

- http://baike.corp.taobao.com/index.php/Hive
- https://cwiki.apache.org/confluence/display/Hive/Home
- http://www.tbdata.org/archives/category/cloud-computing/hive
- **Hadoop.The.Definitive.Guide.2nd.Edition 第12章**

课程回顾、总结页

- Hive Components
- MapReduce
- Hive QL
- Hive 优化
- SQL 优化

谢 谢 ！