

# BigData in Real-time

## Impala Introduction

TCloud Computing 天云趋势

孙振南

zhennan\_sun@tcloudcomputing.com

2012/12/13

Beijing

Apache Asia Road Show

## Background (Disclaimer)

- Impala is NOT an Apache Software Foundation project yet
- Impala uses ASLv2
- The speaker (me) is NOT associated with Cloudera

# WHY

# Impala

## The Need For Speed

- **What's wrong with MapReduce?**
  - Batch oriented. Good at complex jobs
  - But slow at startup & shuffle
  - Programmer friendly
- **How about Hive?**
  - SQL friendly. Still slow as ...
- **How about HBase?**
  - Slow data import
  - No SQL

## The leads from the leader

- **Google BigQuery, the service**
  - Based on Google Dremel, the paper
    - SQL-like interface
    - Interactive analysis of PBs data
    - Query Execution Tree
      - Tasks to sub-tasks, instead of identical distributed tasks
    - Columnar storage based on nested ProtoBuffer data
      - Faster traversing
- **Amazon RedShift is another story...**

## Open source alternatives?

- **Apache Drill**
  - No substantial progress
  - Mailing list msg # dropped 80% from Sep to Nov
- **Berkeley Shark/Spark**
  - Shared memory based, good at iteration tasks
  - Different component stack
- **Cloudera Impala**

## Positioning

- **Compared with MR, it's all about trade-off**
  - Complexity or responsiveness
  - General purpose or ad-hoc
- **MPP-RDB paradigms on top of commodity DFS**
  - On par performance in some cases
  - Extremely cheap
  - Linear scalability

# WHAT is Impala



## Features

- **Distributed SQL on raw HDFS files**
  - Select, where, aggregation, join,
  - Insert into/overwrite
  - Text and Sequence files
- **Hive compatible “meta store” and interface**
  - Reuse Hive’s metadata schema, DDL and JDBC/ODBC driver
- **Up to 90x times faster, compared with Hive**
  - Purely I/O bound scenario, 3-4X
  - With joins, 7-45X
  - With memory cached, 20-90X

## Status

- **Announced at Oct/2012**
  - Now 0.3 at Dec/5
  - Has been private beta for half-year
  - Currently in public beta
  - Target GA @ 2013 Q1
- **Entirely developed by Cloudera (by now)**
  - In the past 2 years, 7 full time engineers
- **Completely open source, ASLv2**



## Example

```
$ hive
```

```
hive> CREATE TABLE sales (id STRING, item STRING, price int);
```

```
hive> load data local inpath '/store/sales.txt' into table sales;
```

```
$ impala-shell -impalad=172.16.204.4:21000
```

```
[172.16.204.4:21000]> show tables
```

```
sales
```

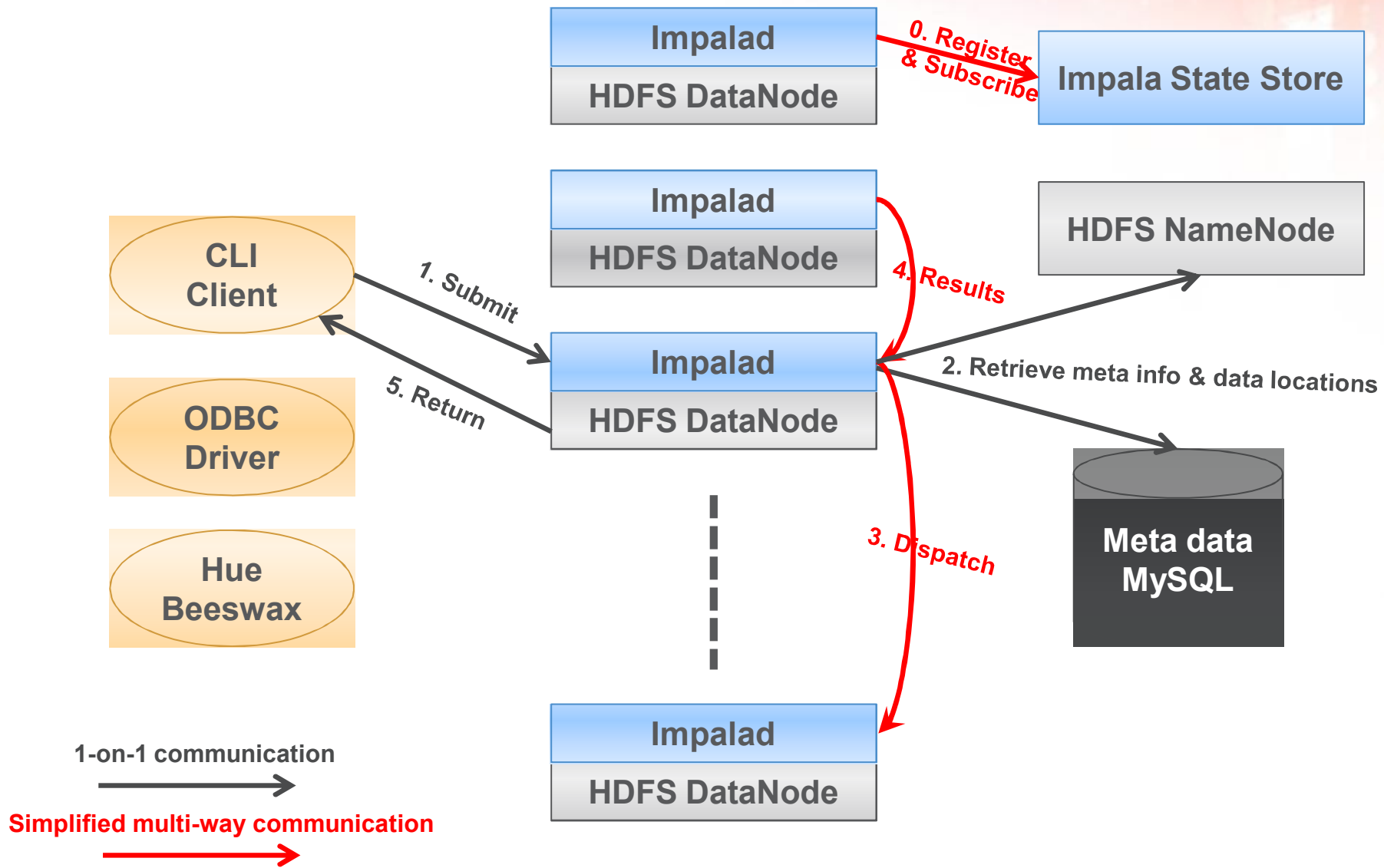
```
[172.16.204.4:21000]> select * from sales where price > 100 limit 3
```

```
13221    COOKIE    138
```

```
38384    DIPER     287
```

```
85845    TV        737
```

# Workflow Overview



# HOW the Impala speed up

# 'MPP' SQL

- **PlanNode**

- Node of the Depth-First execution plan tree
- Various types
  - HDFS\_SCAN\_NODE, HBASE\_SCAN\_NODE
  - HASH\_JOIN\_NODE
  - AGGREGATION\_NODE, SORT\_NODE
  - EXCHANGE\_NODE

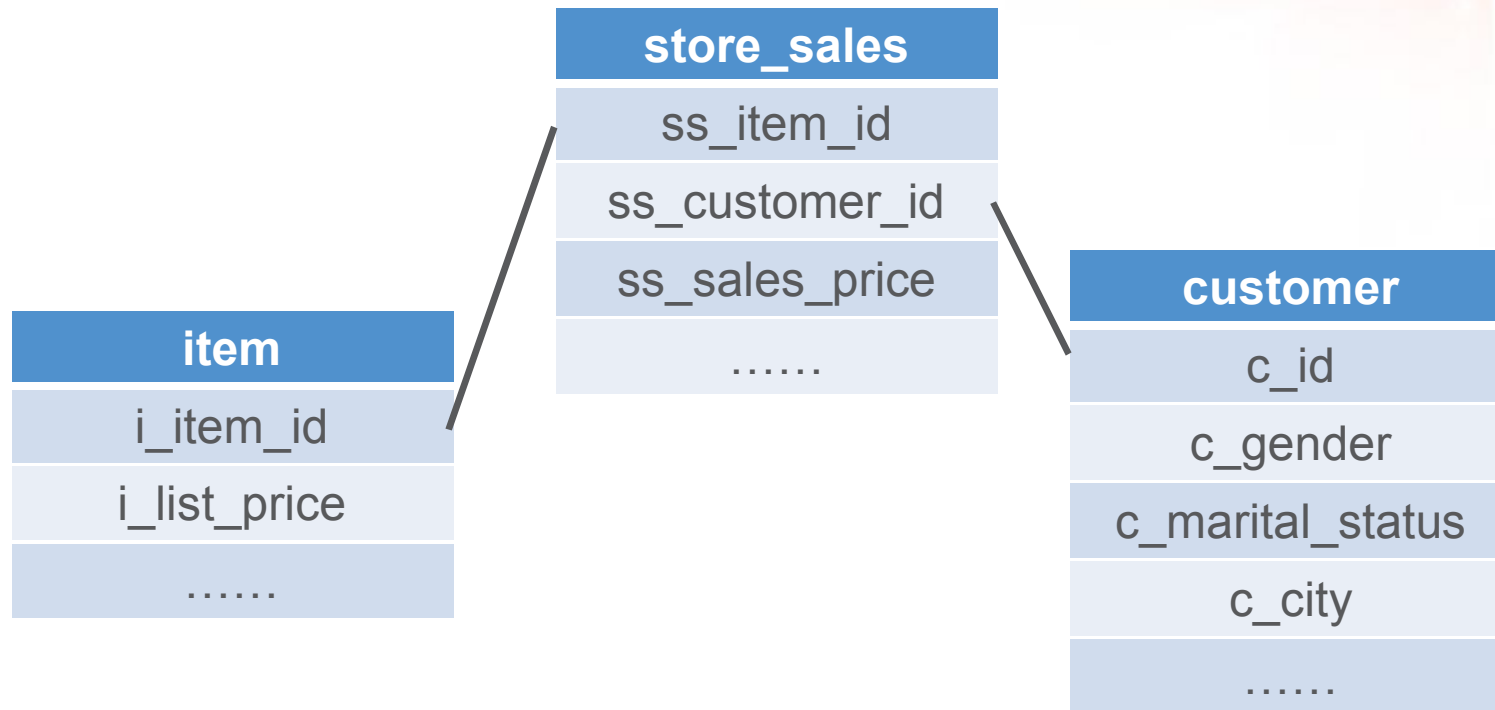
- **Fragment**

- Atomic executable unit, could be distributed
- Contains one or more PlanNodes
  - Depends on the data distribution and the SQL statement

## SQL breakdown sample

- There's a saying that young single males don't use coupon or discount as much as others, is it true?
- We can compare the list price and sales price
  - Items are a little bit expensive
  - Buyers are young, single, male
  - Live in major city

## SQL breakdown sample - tables





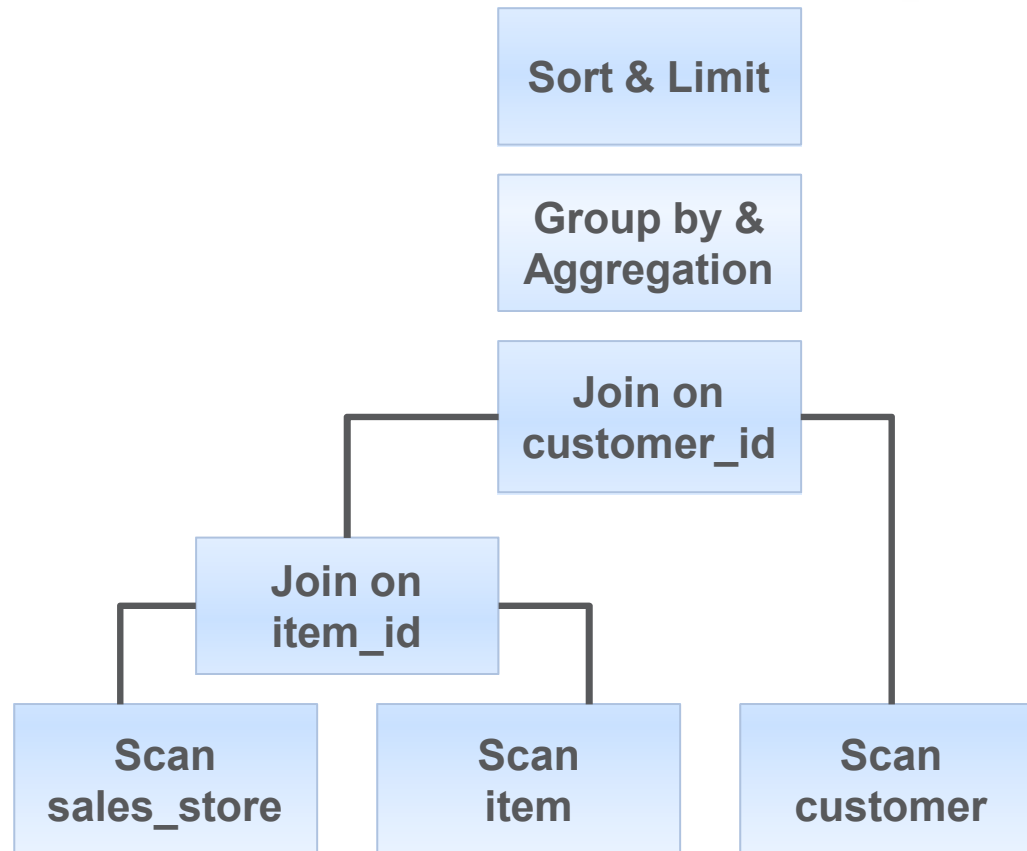
## SQL breakdown sample – SQL statement



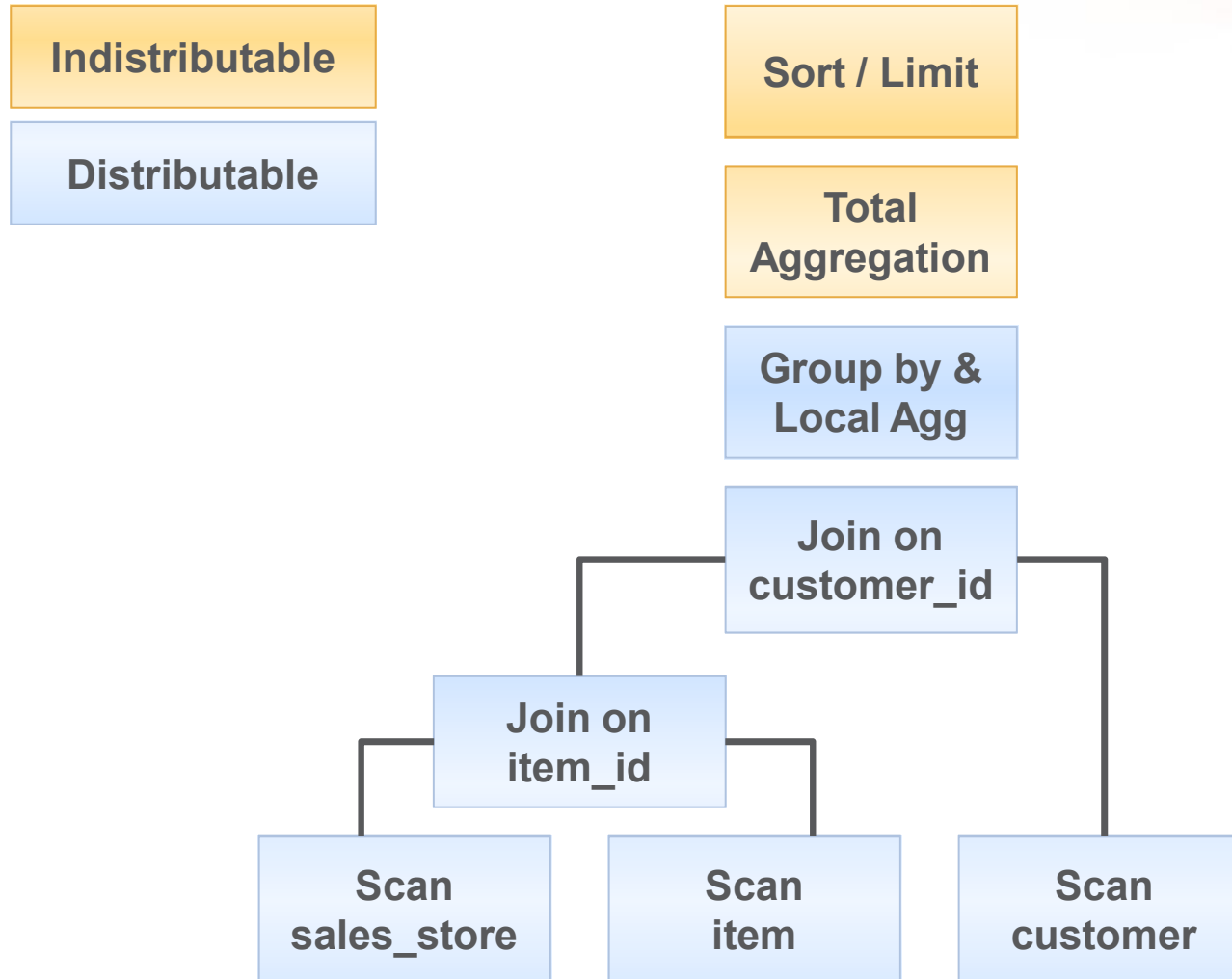
云基地创造  
Created By Cloud Valley

```
select i_item_id, i_list_price, avg(ss_sales_price) agg1  
  
FROM store_sales  
  
JOIN item on (store_sales.ss_item_id = item.i_item_id)  
JOIN customer on (store_sales.ss_customer_id = customer.c_id)  
  
where  
  
    i_list_price > 1000 and  
    c_gender = 'M' and  
    c_marital_status = 'S' and  
    c_city in ('Beijing', 'Shanghai', 'Guangzhou')  
  
group by i_item_id,  
order by i_list_price  
limit 1000
```

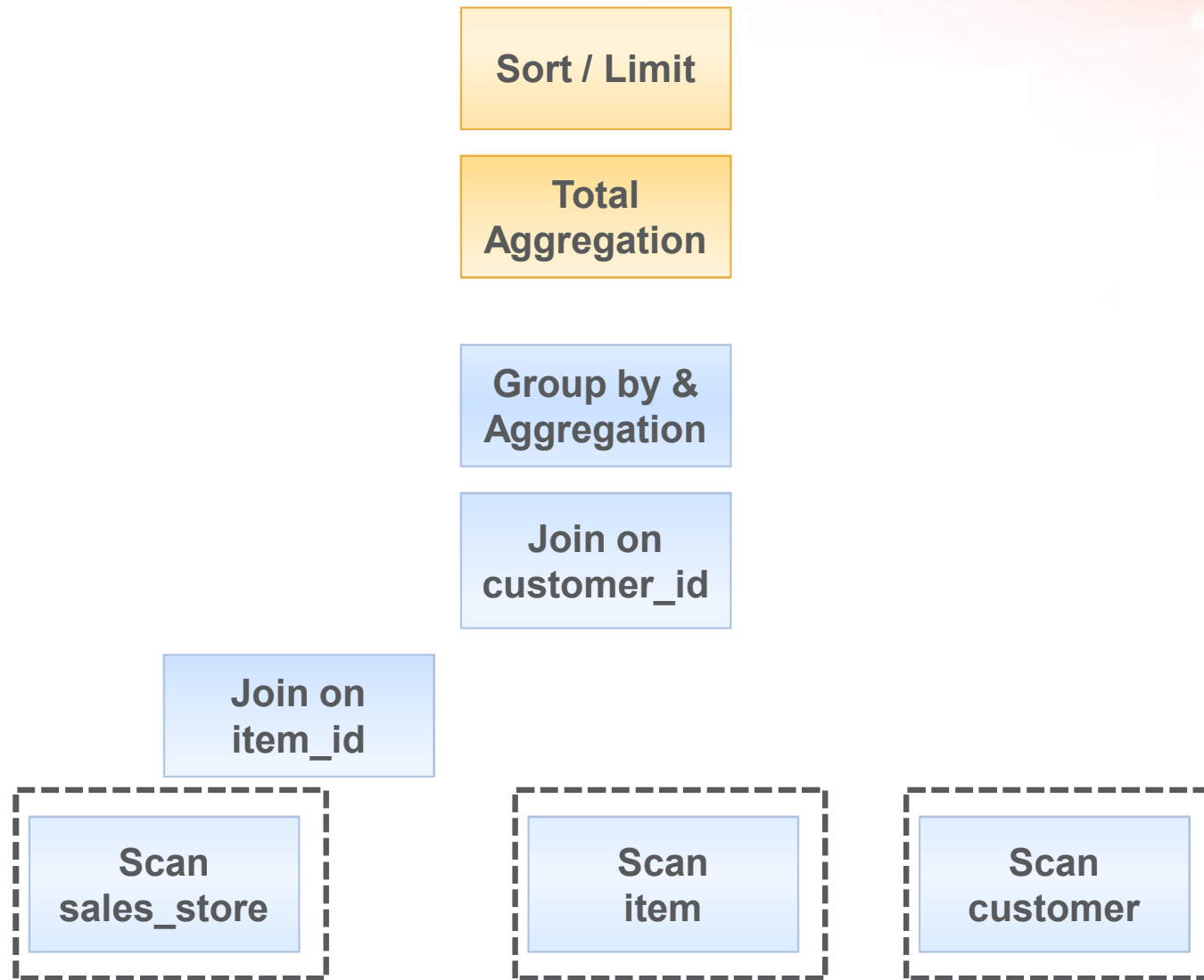
# Execution plan tree



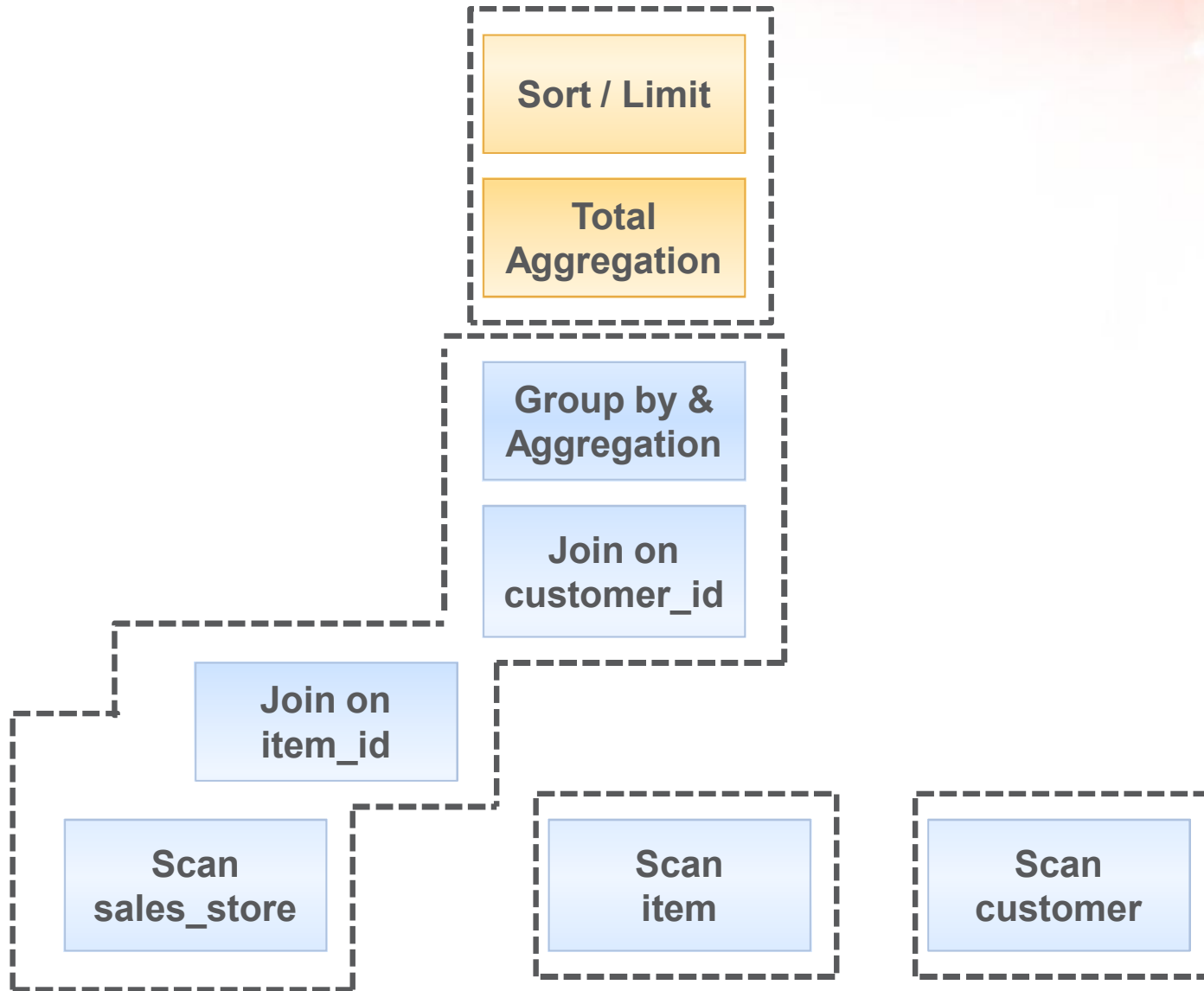
# Execution plan tree – Distributed!



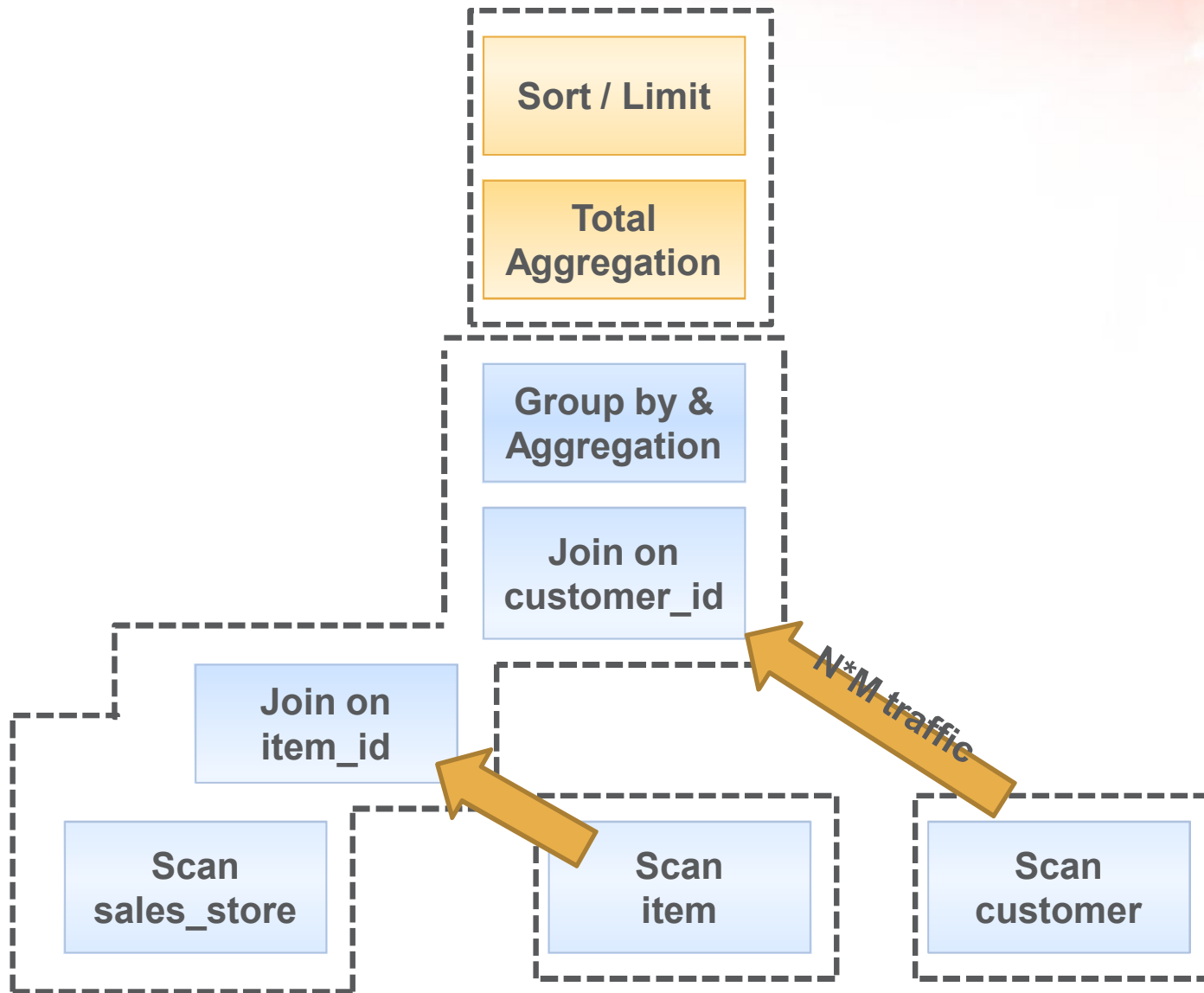
# From execution plan to fragment



# From execution plan to fragment



# From execution plan to fragment



# THAT'S IT?

## There's more...

- **Written in C++**
  - Only used jFlex/CUP to parse the SQL statement
- **Local compilation of fragments**
  - LLVM is used
- **Disk awareness**
  - Not just host awareness
  - `dfs.datanode.hdfs-blocks-metadata.enabled`
  - “40% faster”
- **Direct read**
  - Not via HDFS NameNode then DataNode then ...
  - `dfs.client.read.shortcircuit`, `dfs.client.read.shortcircuit.skip.checksum`

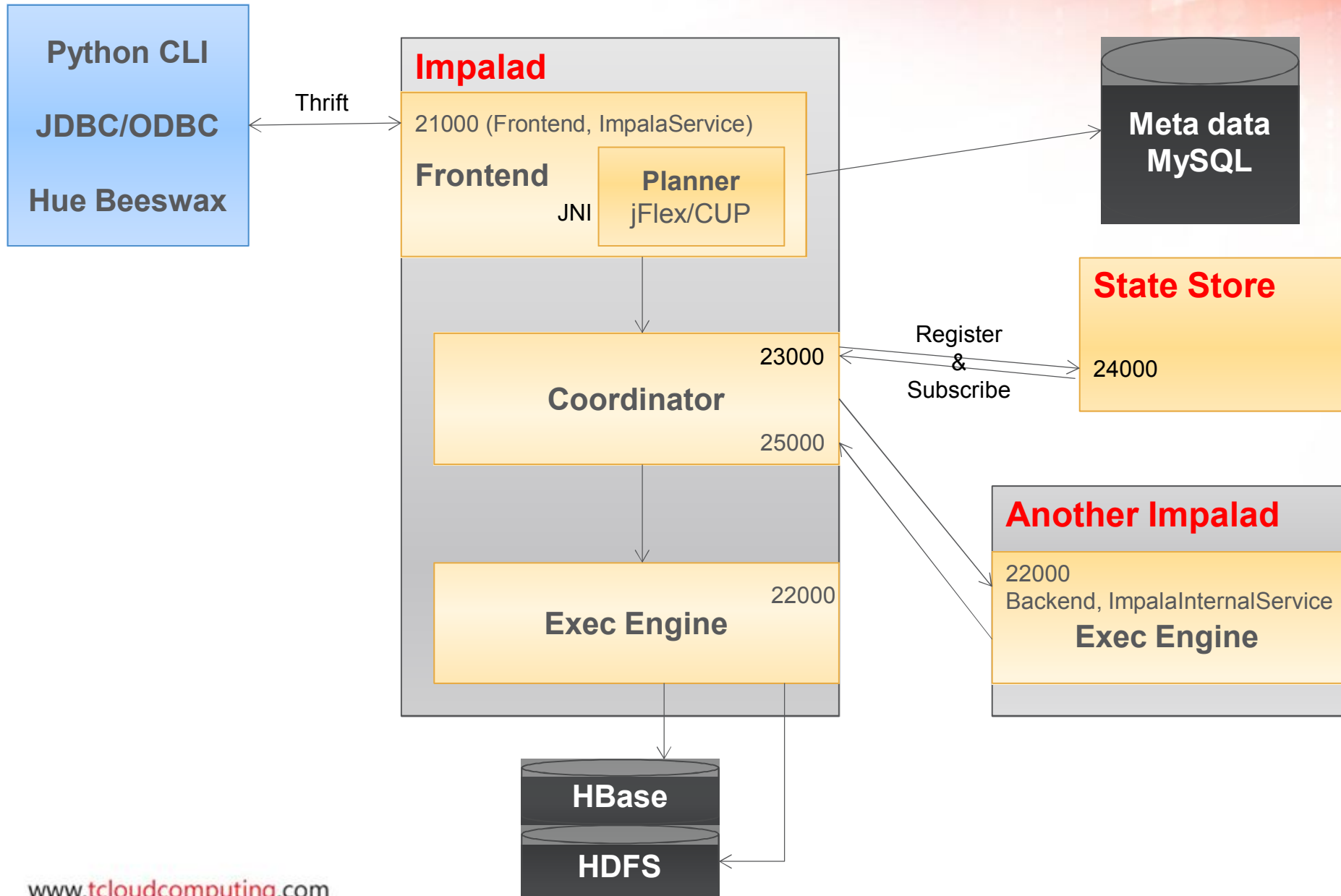


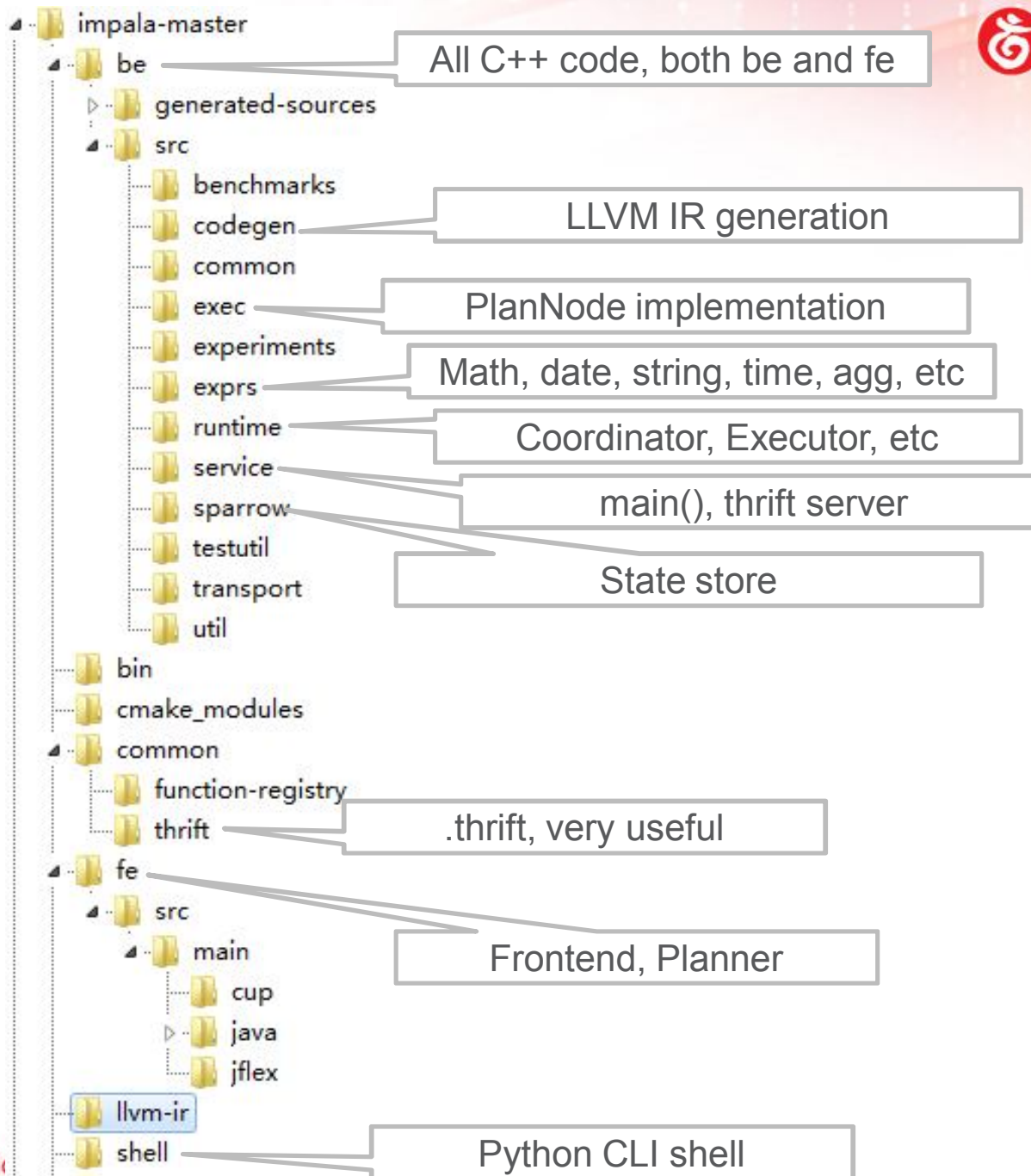
# GOOD ENOUGH?

## TODOs

- No Data Definition Language yet
- No User Defined Function yet
- No fault tolerance yet
- Avro, RCFile, LZO, Trevni support is on the way
  - Impala + Trevni will introduce another performance boost
    - With more SQL functions compared with BigQuery
- In memory Join only
  - Will be fixed in GA
- Partition before join, reduce traffic
- Support Hive partitions, but not buckets yet

# INSIDE





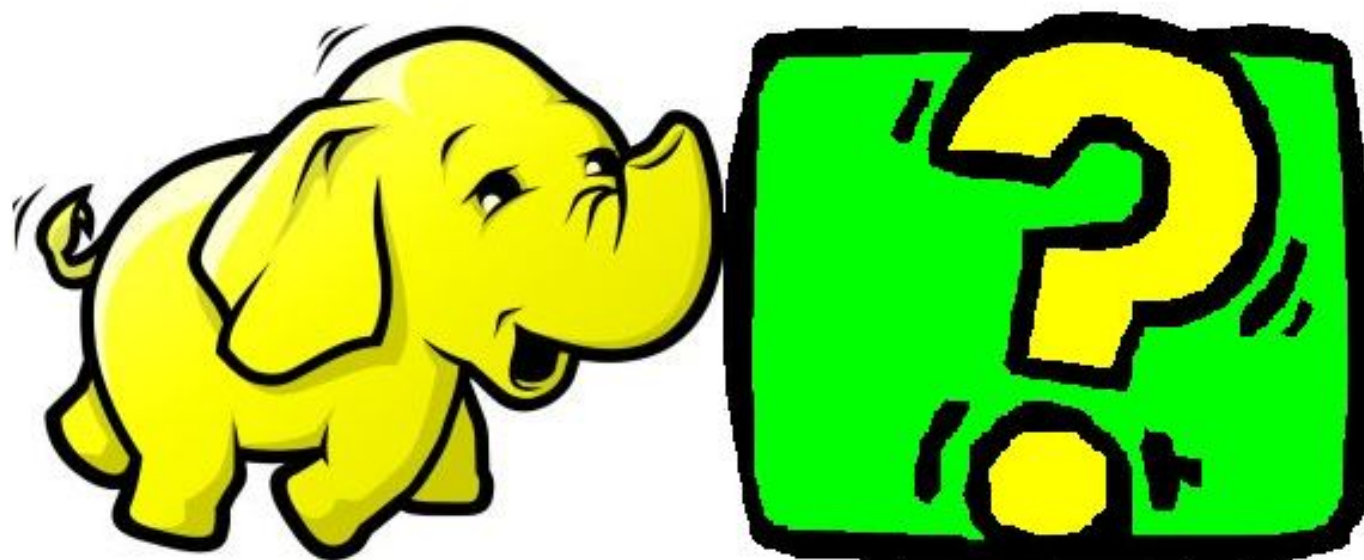
## 趋势科技如何使用Hadoop

- 天云趋势由天云科技和趋势科技共同投资成立于**2010年3月**
- 趋势科技是**Hadoop**的重度使用者:
  - 2006年开始使用, 用于处理Web Reputation和Email Reputation等
  - 五个数据中心, 近1000个节点
  - 日均处理**3.6T**日志数据
    - 含HoneyPot收到的垃圾邮件但不含爬取的网页等
  - 亚洲最早, 也是最大的代码贡献者
  - HBase 0.92新功能的主要开发者 (coprocessors, security)

## 天云趋势的BigData业务

- TCloud BigData Platform, 集部署、监控、配置优化、数据处理以及扩展SDK库于一体的集群管理和开发平台
- 丰富的垂直行业开发经验, 包括电信、电力、安全等
- 为多家大型传统软件开发厂商提供咨询服务
- Hadoop管理员、开发者培训, 也可为企业内部进行定制化培训

<http://www.tcloudcomputing.com.cn>



 新浪微博 @少年振南