



喜马拉雅数据计算平台XQL

数据组 陈涛
2017.11



Outline

- » XQL总览
- » 系统演进过程
- » 周边产品
- » 经验总结
- » 未来展望



XOL总览

» 研发背景

» 总体架构

» 使用人群与场景

全球互联网技术大会



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE

全球互联网技术大会



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE

全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



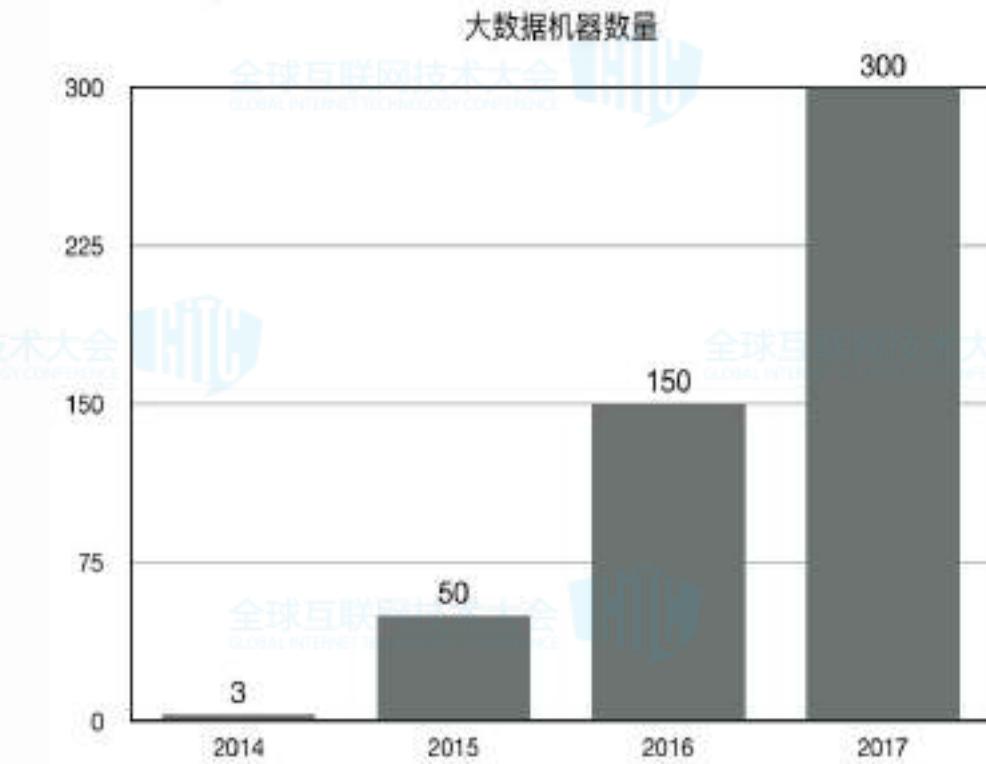
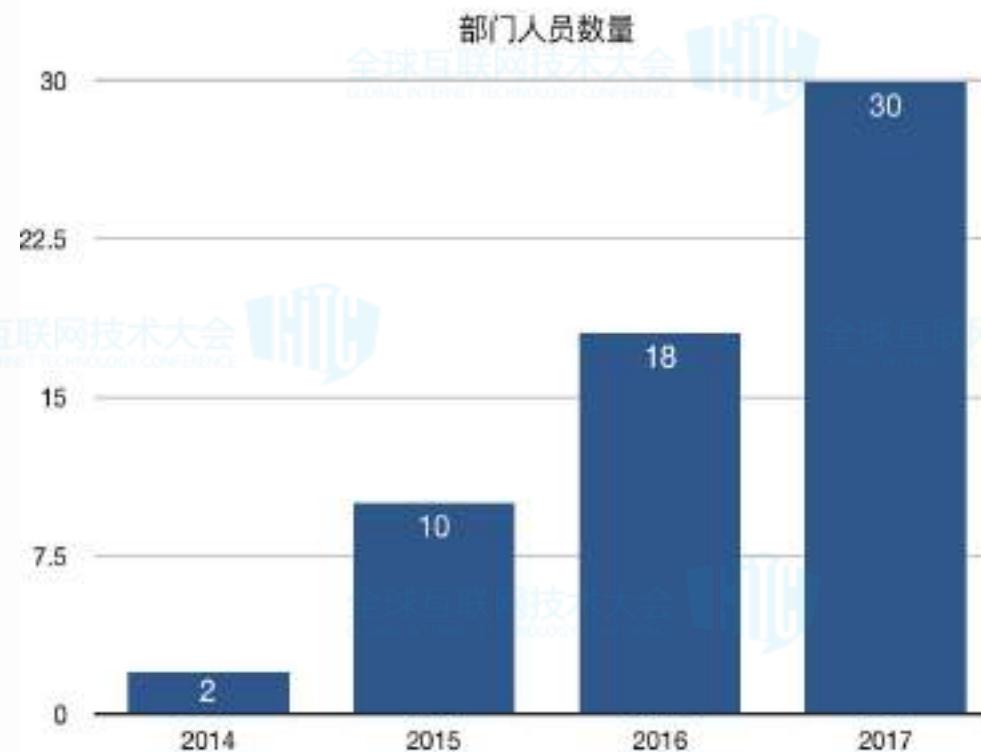
全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



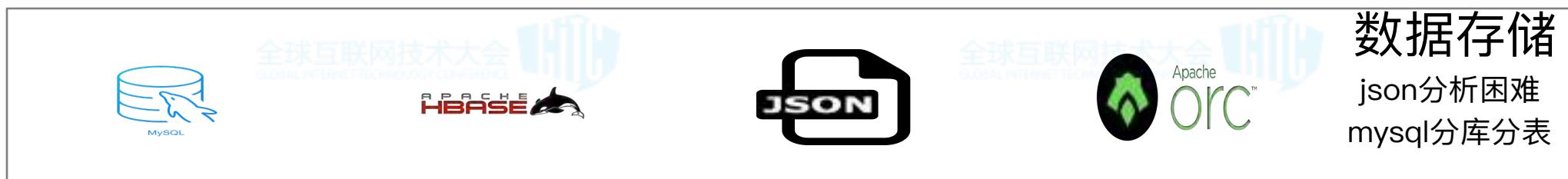
全球互联网技术大会
GLOBAL INTERNET TECHNOLOGY CONFERENCE



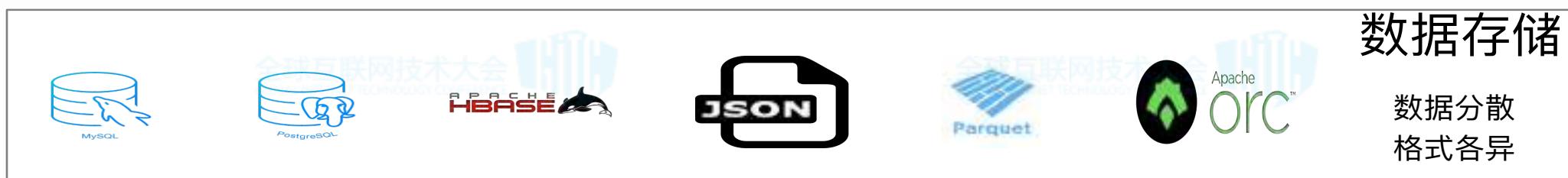
XOL总览 研发背景



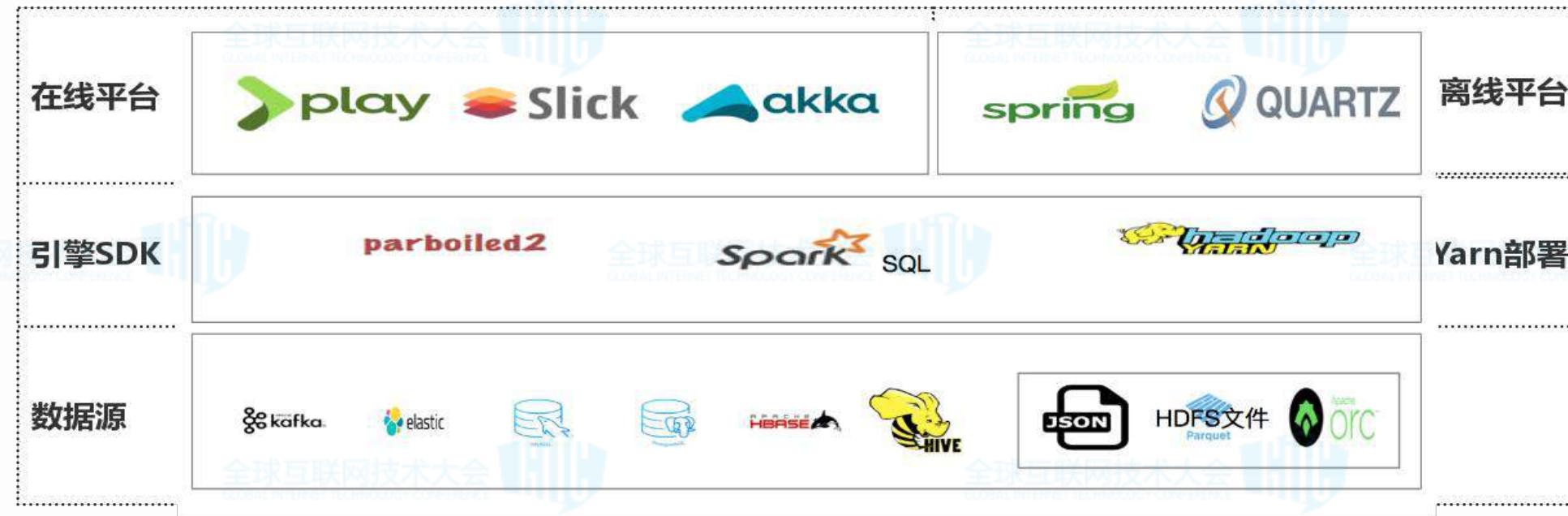
XOL总览 研发背景



XOL总览 研发背景



XQL总览 总体架构

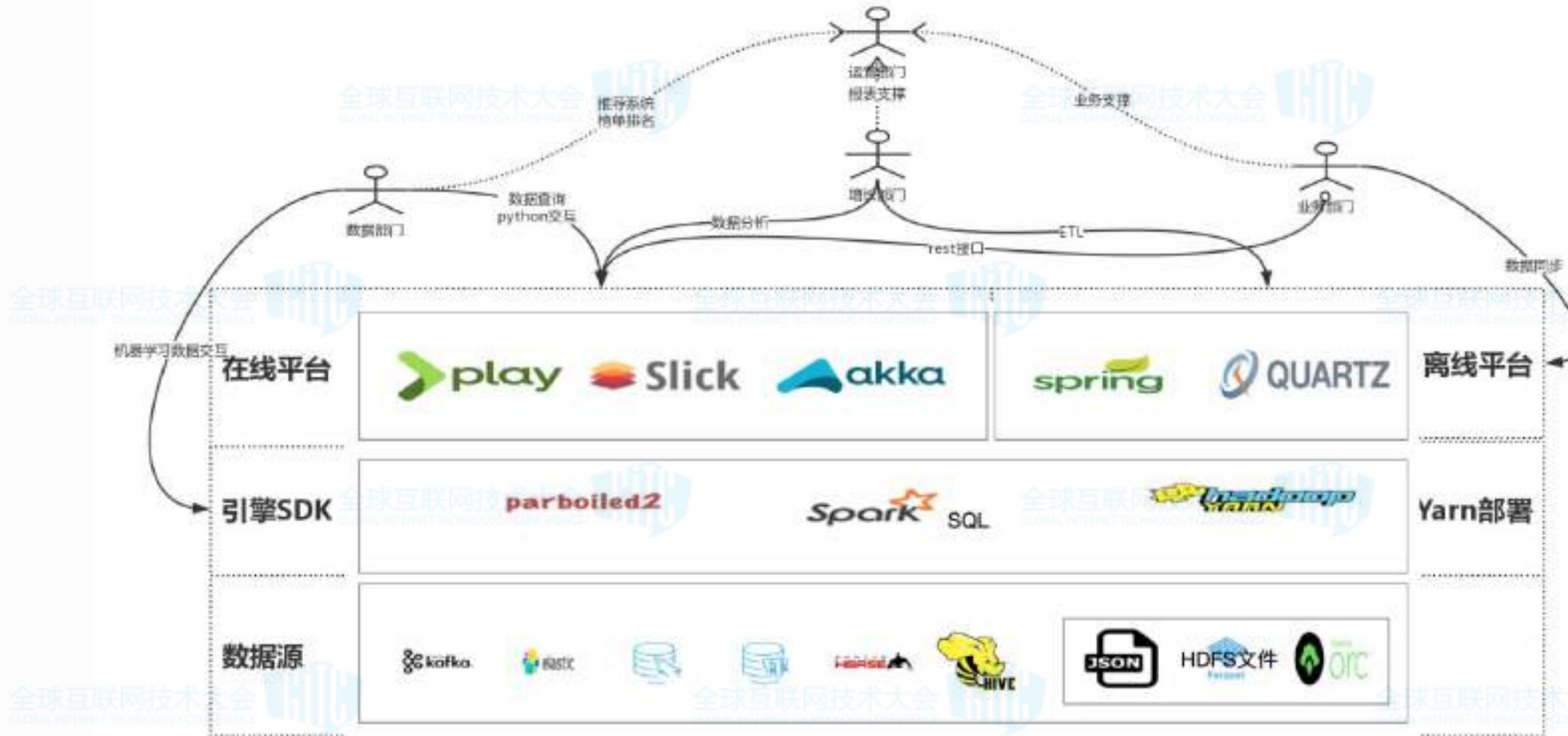


XQL总体架构图

XQL总览 总体架构

- » memory: 4T
- » spark Task: 200w+
- » XQL job: 4000+
- » dataSource: hdfs、hive、hbase、es、kafka、mysql、pg
- » fileFormat: parquet、orc、csv、json、xml

XOL总览 使用人群与场景



系统演进过程

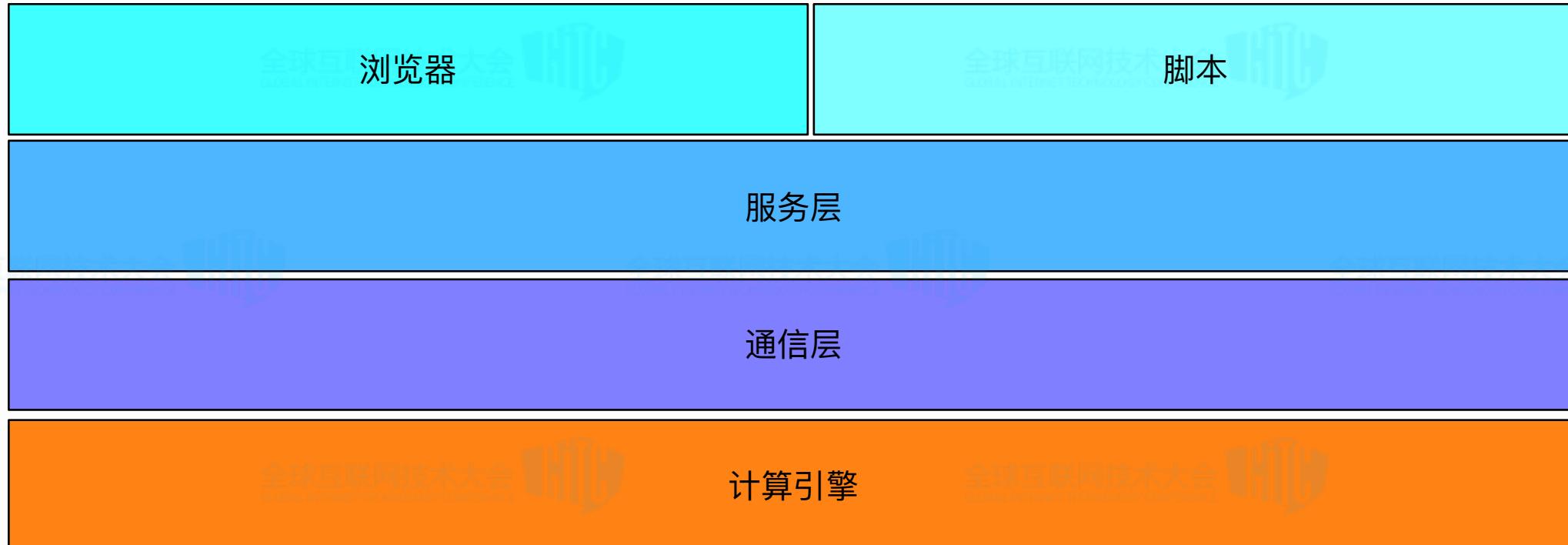
» v1版本

» v2版本

» v3版本

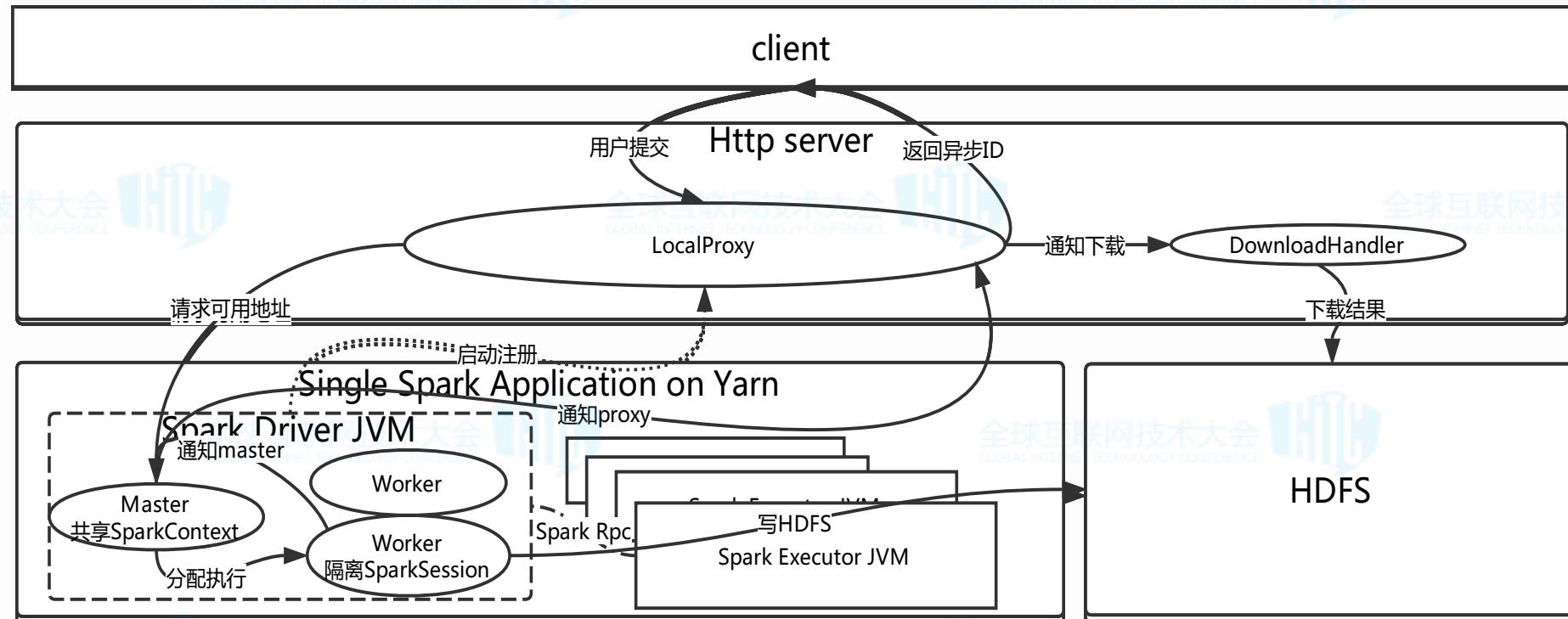


系统演进过程 v1版本架构



v1版本架构图

系统演进过程 v1版本通信逻辑



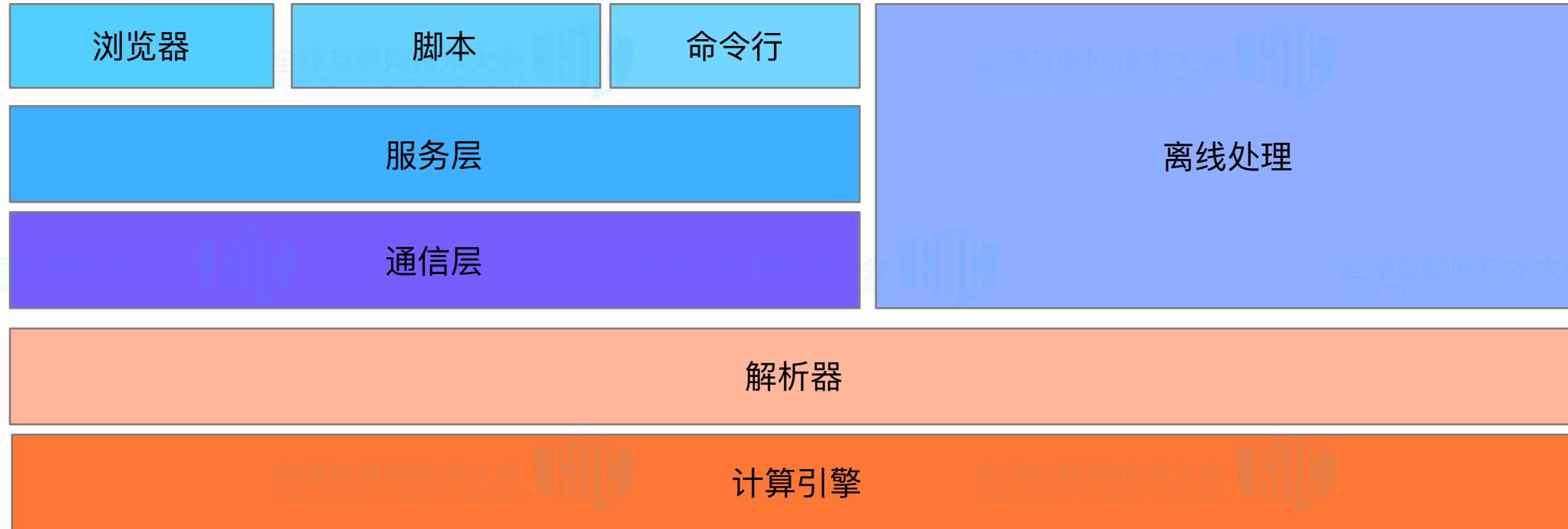
系统演进过程 v1版本其他功能

- » 采用spark sql代替hive
- » spark常驻服务
- » 对hbase和json的hive表做了适配
- » 支持文件的上传下载
- » 提供rest接口，支持shell脚本和程序调用

系统演进过程 v1版本存在的问题

- » 数据源支持比较有限
- » 依赖hive的元数据
- » sql可读性差的问题仍未解决
- » 服务稳定性不高
- » 没有用户权限验证
- » 大任务会耗尽所有资源

系统演进过程 v2版本架构



系统演进过程 v2版本综合demo

```
-- CREATE table ...;
INSERT OVERWRITE TABLE hive.old_user_order PARTITION (age)
SELECT uid,
       name,
       age,
       play_avg_duration,
       order_amount_sum FROM(
           (SELECT name,
                  age,
                  avg(play_duration) play_avg_duration ,
                  uid
             FROM user_info
           LEFT JOIN user_event
             ON user_info.uid = user_event.user_id
           WHERE user_info.age > 50) a
           LEFT JOIN
           (SELECT sum(order_amount) order_amount_sum ,
                  order_uid
             FROM user_order
            GROUP BY order_uid) b
           ON a.uid = b.order_uid);

```

Without XQL

```
LOAD hive.`user_info` WHERE age > 50 AS old_user;
LOAD hive.`user_event` AS user_event;
SELECT name,
       age,
       avg(play_duration) play_avg_duration ,
       uid
FROM old_user
LEFT JOIN user_event
ON old_user.uid = user_event.user_id AS old_user_event;
LOAD hive.`user_order` AS user_order;
SELECT uid,
       name,
       age,
       play_avg_duration,
       sum(order_amount) order_amount_sum
FROM old_user_event
LEFT JOIN user_order
ON old_user_event.uid = user_order.order_uid
GROUP BY order_uid AS result;
SAVE overwrite result AS `hive.old_user_order` partitionBy age;
```

XQL

系统演进过程 v2版本load语法

--LOAD DSL, 描述数据输入

```
LOAD format.`param` (schema) condition AS temp_table_name;
```

» Datasource:

- » HDFS (parquet、orc、json、csv)
- » HBase
- » JDBC Sharding
- » Hive

系统演进过程 v2版本load语法

```
--hbase load DSL  
--维护hbase columnFamily:column到spark column的映射  
LOAD hbase.`tableName`  
  (:id,user:name ,user:age int#b as u_age)  
 WHERE row between '0' and '1'  
 AS hb_user;
```

系统演进过程 v2版本save语法

--SAVE DSL, 描述数据输出

```
SAVE saveMode temp_table_name AS format.`param` partitionBy COLUMN;
```

» Datasink:

- » HDFS (parquet、orc、json、csv)
- » HBase
- » JDBC
- » Hive

系统演进过程 v2版本save语法

```
-- 除了 spark 的 saveMode(表级别)
-- (append,overwrite,errorIfExists,ignore)
-- 额外支持行级别的更新和忽略(update,ignoreRecord)
SAVE
    update user
        (id as uid,name ,age)
    AS jdbc.`ds.db.tb_user`
```

系统演进过程 v2版本web演示

The screenshot shows a web-based XQL (eXtensible Query Language) interface. The top navigation bar includes tabs for 'XQL' and 'SQL'. The main workspace has a title '演示' (Demonstration). On the left, there's a sidebar with a tree view containing numerous items labeled 'mess' followed by a number (e.g., mess_1, mess_2, ..., mess_24). The central area contains the following XQL code:

```
1 LOAD json.* AS t;
2 SELECT
3 props.`error_code` response
4 FROM t
5 WHERE props.`error_code` != '200' AS reponse_result;
6 SELECT
7 count(1) errorNums,
8 response
9 FROM reponse_result
10 GROUP BY response
11 ORDER BY errorNums DESC;
```

Below the code, there are two tabs: '查询结果' (Query Results) and '执行计划' (Execution Plan). The '查询结果' tab displays a table with the following data:

#	errorNums	response
0	172174	404
1	5774	400
2	26	501
3	6	500

系统演进过程 v2版本其他功能

- » 大任务报警机制
- » 语法帮助，sql自动生成
- » 上传下载对文件智能转码
- » 采用spark的fair调度
- » 支持邮件订阅结果
- » 统一账户管理

系统演进过程 v2版本监控报警



```

class JobSizeListener(masterRef: ActorRef,
                      numMaxTasks: Int) extends SparkListener {
  private val logger = org.slf4j.LoggerFactory.getLogger(this.getClass)

  override def onJobStart(jobStart: SparkListenerJobStart): Unit =
    try {
      val xqlId = BatchSQLRunnerEngine.getXQLIdByGroupId(jobStart.id)
      val maxTasks = jobStart.stageInfos.map(_.numTasks).max
      if (maxTasks > numMaxTasks) {
        masterRef ! LargeJob(xqlId, jobStart.jobId, maxTasks)
      }
    } catch {
      case e: Exception =>
        logger.error(s"Error occurred while processing job start: ${e.getMessage}")
    }
}

class JobWarningActor(queryResultStorage: QueryResultStorage,
                      authService: AuthService) extends Actor {
  import context.dispatcher
  private final val dingUrl = ConfigFactory.load().as[Option[String]]("dingtalk_url")
  private final val logger = org.slf4j.LoggerFactory.getLogger(this.getClass)
  private val timeInterval = ConfigFactory.load().as[Option[List[String]]]("warning_time_interval").map(_.mkString(" "))

  def receive: Receive = {
    case JobWarning(xqlId, jobId, numTasks) if !(queue.contains(xqlId)) ||
        logger.info("large job ready to warning in job warning actor, job id: " + jobId)
        queue.add(xqlId)
        queryResultStorage.queryByIdAsync(xqlId).foreach { result =>
          authService.getUserEmail(result.userId).foreach(email =>
            DingTalkUtil.sendMessage(dingUrl.get, s"Large job warning: $email, job id: $jobId, num tasks: $numTasks"))
        }
  }
}

```

系统演进过程 v2版本sql补全

The screenshot shows the XSQL IDE interface. On the left, there's a tree view of a database schema under 'recsys'. A node named 'album_info' is selected and highlighted with a red circle. The main panel displays a SQL editor with the following code:

```
load jdbc:mysql://127.0.0.1/recsys.album_info` (album_id,correlation) limit 10  
as t;  
select album_id,correlation from t
```

系统演进过程 v2版本存在的问题

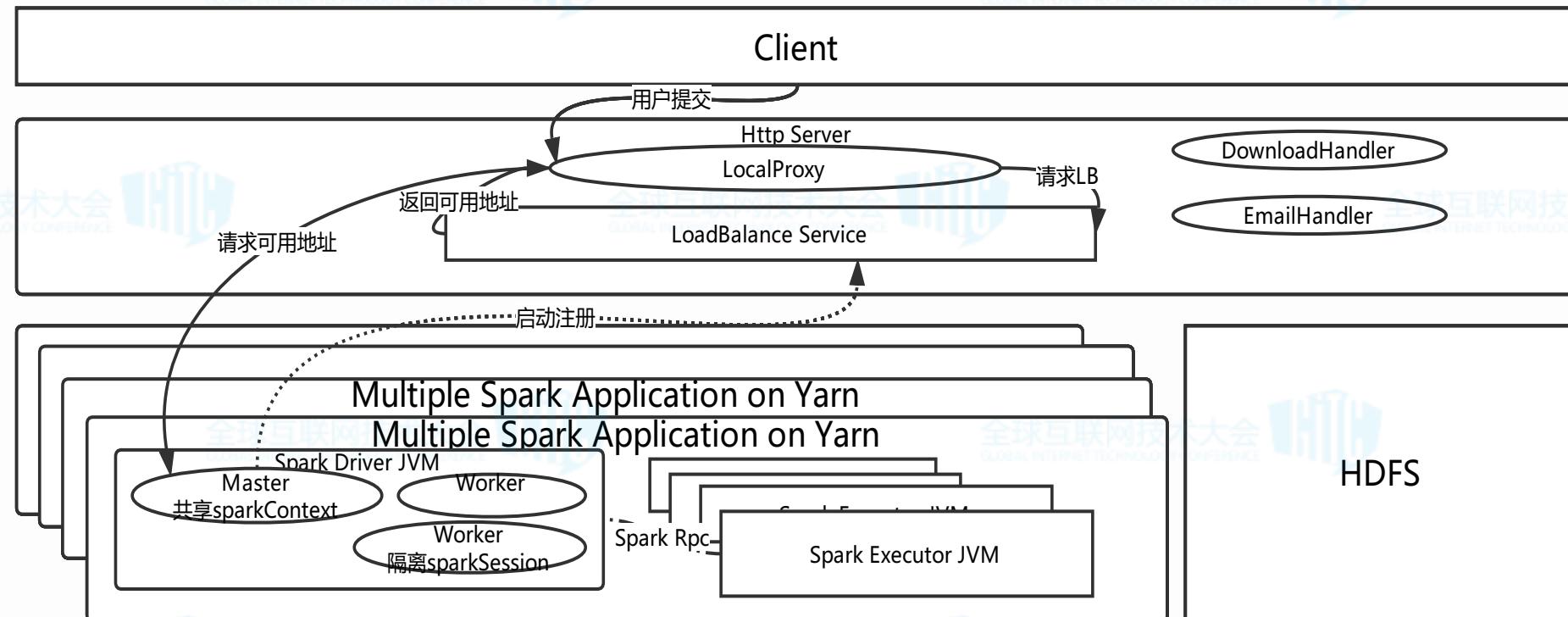
- » 没有负载均衡
- » 缺少权限和审计
- » tableau(商业分析软件)的最后一公里
- » 开发人员的拓展需求

系统演进过程 v3版本架构图



v3版本私有部署架构图

系统演进过程 v3版本通信逻辑



系统演进过程 v3版本权限模块

- » id加密混淆
- » 数据源读写权限验证
- » 用户操作日志实时落盘



系统演进过程 v3版本权限模块

ID加密



On xql架构 - ProcessOn

tail?_queryid=4+H4NOIGMiY=

自助查询XQL工具

下载文件

执行 7 s, 共影响 242 行

#	节点名	父节点名	类型	值	操作	状态	修改时间	创建时间
0	1		4.0	0		400	36	

@111 举报 [http://www.ximalaya.com/web/detail?_queryid=4+H4NOIGMiY=](#)

全球互联网技术大会

这个是今天刚在做的新功能

XQL已经成为我司开发沟通的一种新方式

07-11 16:08

系统演进过程 v3版本权限模块

```
{  
    "_source": {  
        "conditions": [  
            "limit 10"  
        ],  
        "singleXql": "select * from _zanzhu limit 10",  
        "tableToRead": [  
            "or_zanzhu"  
        ],  
        "tableToWrite": [],  
        "userEmail": "ximalaya.com",  
        "userMobile": ""  
    }  
}
```

审计日志实时落盘ES

系统演进过程 v3版本权限模块

XQL 全球互联网技术大会

写入检测 x

19s前 1 LOAD json.`/root` AS t; 耗 0 s
31s前 2 SAVE overwrite t as parquet.`/root` 耗 1 s
50s前 3 SELECT * FROM t 耗 10 s
55s前 4 SELECT * FROM t 耗 2 s
1min前 5 SELECT * FROM t 耗 5 s
2day前 6 SELECT * FROM t 耗 8 s
2day前 7 SELECT * FROM t 耗 0 s

读写权限验证

查询结果

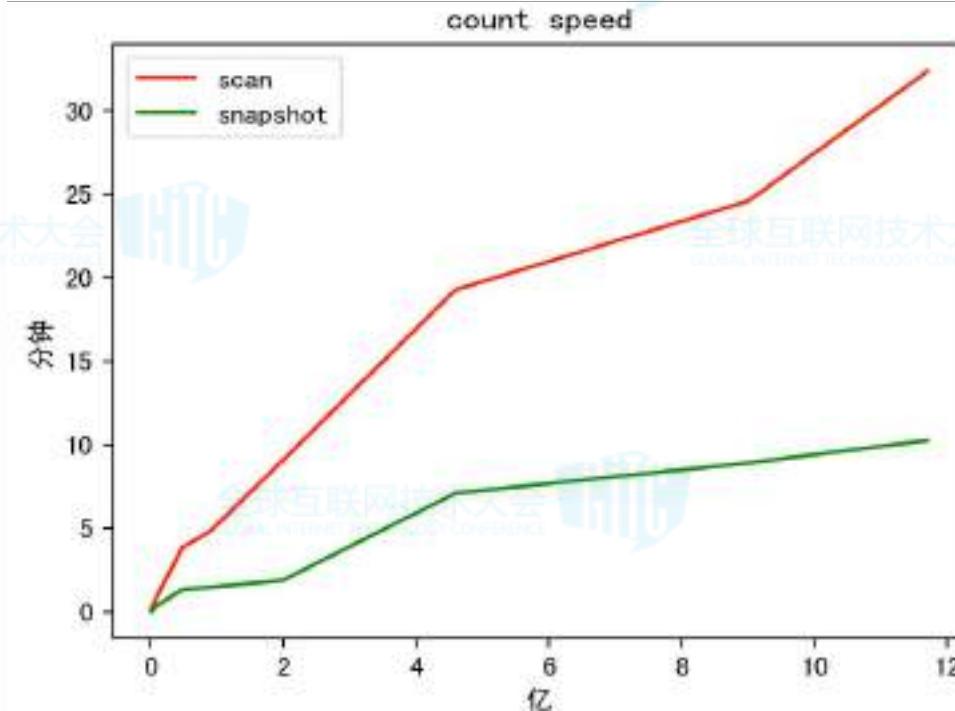
/root is illegal path,can't save to this path use Some(overwrite) mode



系统演进过程 v3版本性能优化

- » hbase增加snapshot的读取方式
 - » 测试环境xql，内存400G
 - » spark 2.2
 - » hbase 0.98.18

```
1 --hbase snapshot test
2 load hbase.`snapshot_test` (:key,cf1:column1) as hb_test;
3 select count(*) from hb_test;
```



系统演进过程 v3版本性能优化

- » 小文件写入优化
- » 支持按字段merge
- » 支持数据重排

```

1 load flatfile as result;
2 save overwrite result as hive,'test_small_file';
3
    
```

查询结果

1	9/15/2017, 9:56:39 AM	3	128 MB
2	9/15/2017, 9:56:39 AM	3	128 MB
3	9/15/2017, 9:56:39 AM	3	128 MB
4	9/15/2017, 9:56:39 AM	3	128 MB

几十个小文件

part-00057-866cd5b5-4b4f-4a11-939f-17b8e9d1257a-0000.snappy.parquet
part-00068-866cd5b5-4b4f-4a11-939f-17b8e9d1257a-0000.snappy.parquet
part-00070-866cd5b5-4b4f-4a11-939f-17b8e9d1257a-0000.snappy.parquet

```

1 load flatfile as result;
2 save overwrite result as hive,'test_small_file' coalesce 5;
3
    
```

查询结果

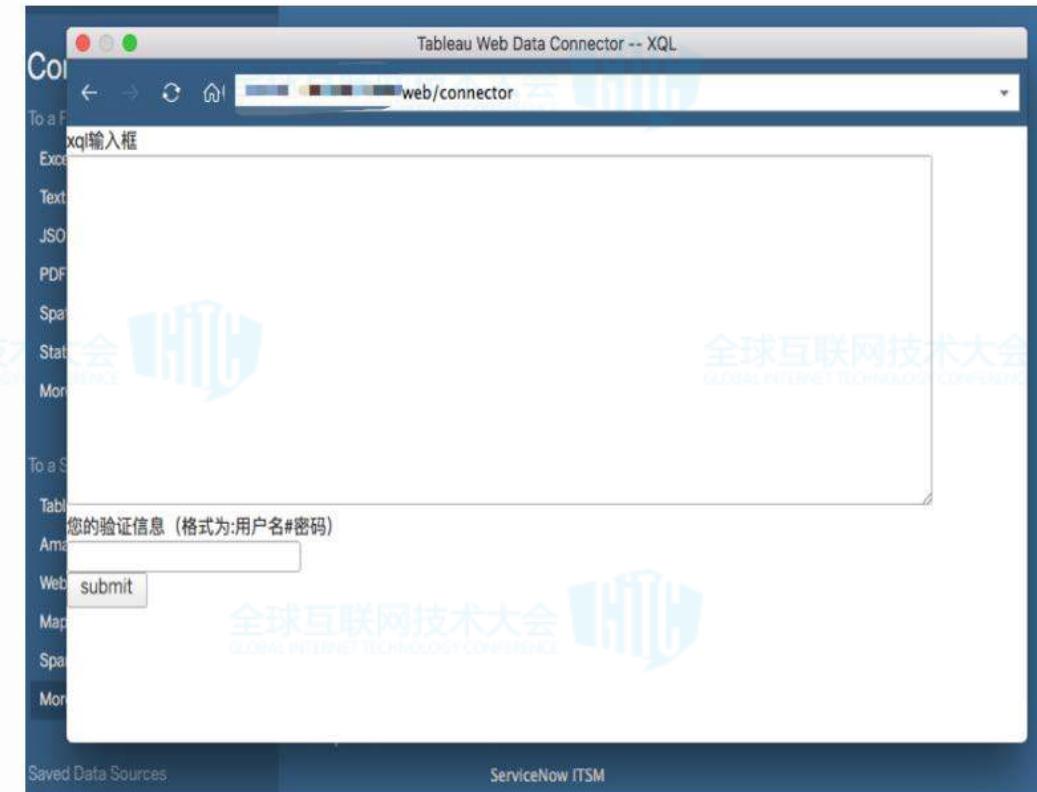
1	9/15/2017, 10:02:21 AM	3	128 MB
2	9/15/2017, 10:02:21 AM	3	128 MB
3	9/15/2017, 10:02:21 AM	3	128 MB
4	9/15/2017, 10:02:21 AM	3	128 MB
5	9/15/2017, 10:02:21 AM	3	128 MB

五个文件

part-00000-1a0dec3b3-df51-4c18-bc9b-37dc295015b7-c000.snappy.parquet
part-00001-1a0dec3b3-df51-4c18-bc9b-37dc295015b7-c000.snappy.parquet
part-00002-1a0dec3b3-df51-4c18-bc9b-37dc295015b7-c000.snappy.parquet
part-00003-1a0dec3b3-df51-4c18-bc9b-37dc295015b7-c000.snappy.parquet
part-00004-1a0dec3b3-df51-4c18-bc9b-37dc295015b7-c000.snappy.parquet

系统演进过程 v3版本tableau支持

- » before: pg、hive
- » now: web connect



系统演进过程 v3版本其他功能

- » 离线版本支持用户udf反射注册
- » 提供公共classpath和私有classpath并存的部署模式，减少冲突
- » 支持kafka、es作为数据源
- » 支持xml格式的hdfs文件
- » 支持灰度发布

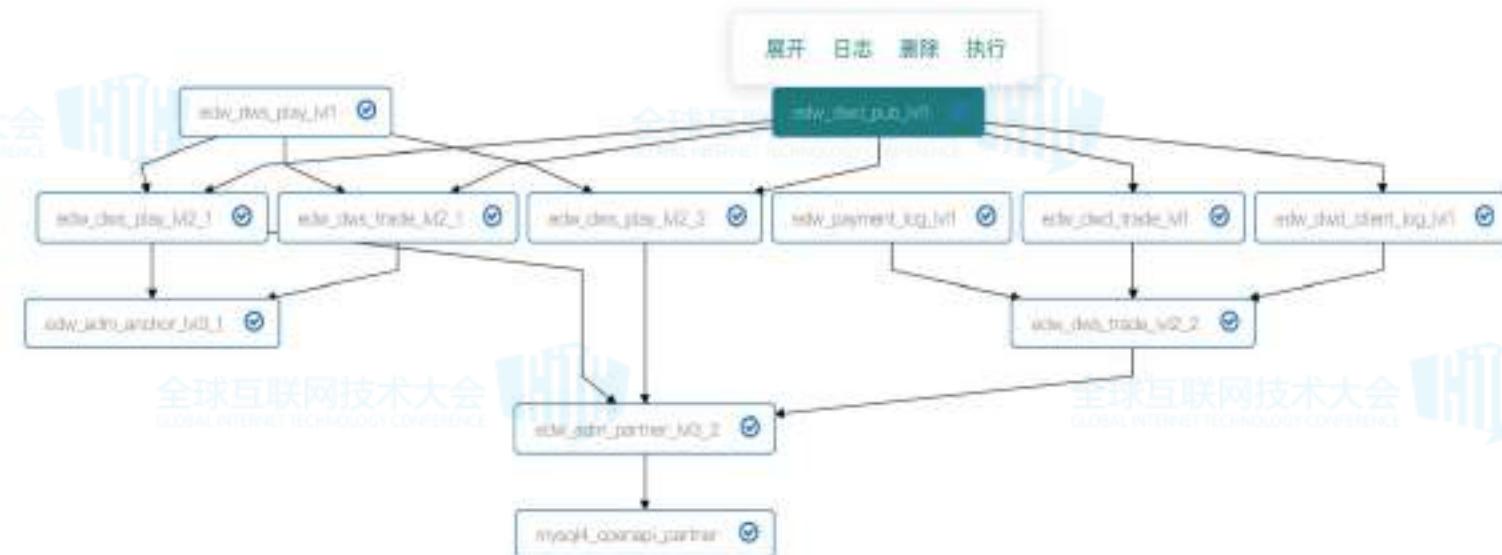
周边产品 调度系统



edw_daily 1

近期执行 0 0 0 0

全球互联网技术大会



返回首页

job属性

job名
edw_dwd_pub_lv1

运行状态
成功

操作人
qianjiali

任务数
3

数据输出
0

最后执行时间
2017/11/13 01:00:01

最后执行时长
15分43秒

创建时间
2017/08/31 19:48:38

描述
数仓第一层公共维度调度

Job DAG



周边产品 调度系统

任务名: edw_dwd_pul

任务类型: 默认任务

spark参数: 自定义

依赖: 480

设置task依赖

节点数: 4

节点核数: 3

节点内存: 格式:8g, 默认:6g

driver内存: 格式:2g, 默认:2g

XSQL *
点击预览
验证sql

Task编辑

task属性	
task名	dwd_anchor_verification_df
运行状态	成功
操作人	qianjial
数据输出	false
数据时间	2017/11/13 00:00:00 ~
最后执行时间	2017/11/13 01:00:02
最后执行时长	5分44秒
描述	加V主播维度表

周边产品 调度系统



数据输出

全球互联网技术大会

展示SQL

请输入内容

全球互联网技术大会

全球互联网技术大会

最大输出条数 ① 20

全球互联网技术大会

表名 ①

默认: dwd_anchor_verification

格式 ①

json

全球互联网技术大会

地址 ①

redis

权限等级 ①

仅自己

邮箱

example@ximalaya.com

抄送人

example@ximalaya.com



example@ximalaya.com



全球互联网技术大会

全球互联网技术大会

取消

确定

数据输出

周边产品 数据看板



经验总结

- » 单元测试的覆盖率
- » 升级前预发布环境流量重放
- » 先小而美，再大而全，快速上线，持续迭代
- » 注重用户体验
- » 积极参与社区

未来展望

- » SQL的优化建议
- » 数据源支持与强化：redis、carbondata、es与kafka
- » 支持类似awk和grep的文本格式的分析
- » 多实例的 AB test



THANKS

----- Q&A Section -----



喜马拉雅FM招大量Java开发
base 上海、成都

todd.chen@ximalaya.com

