

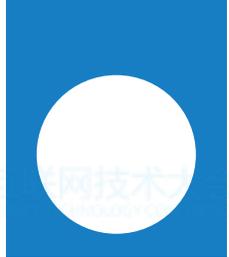


Apache Kylin

加速大数据OLAP

最新进展与典型应用架构分享

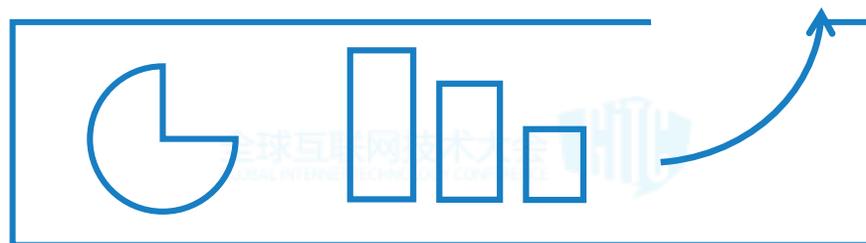
刘一鸣 | Billy Liu
Apache Kylin PMC & Kyligence 产品总监



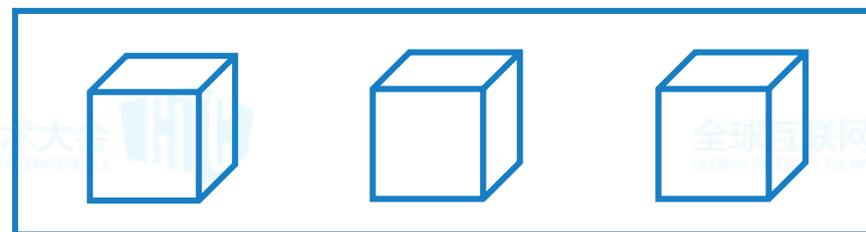
传统企业级数据仓库架构



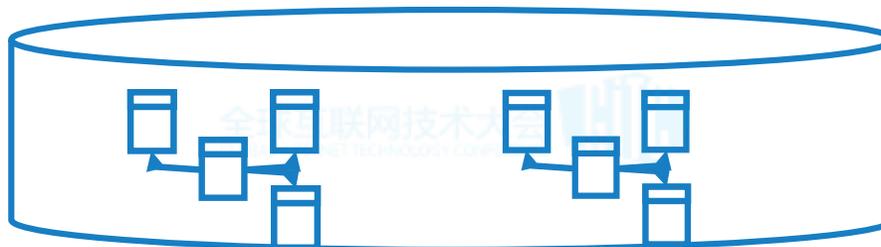
- 针对关键业务分析进行了优化
- 千锤百炼的数据模型
- 行业最佳实践
- 丰富的生态系统
- 大量训练有素的分析师



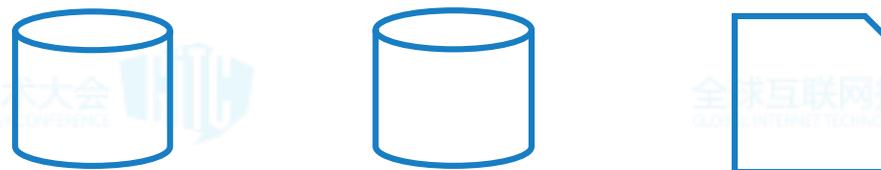
可视化
展现层



OLAP
数据集市



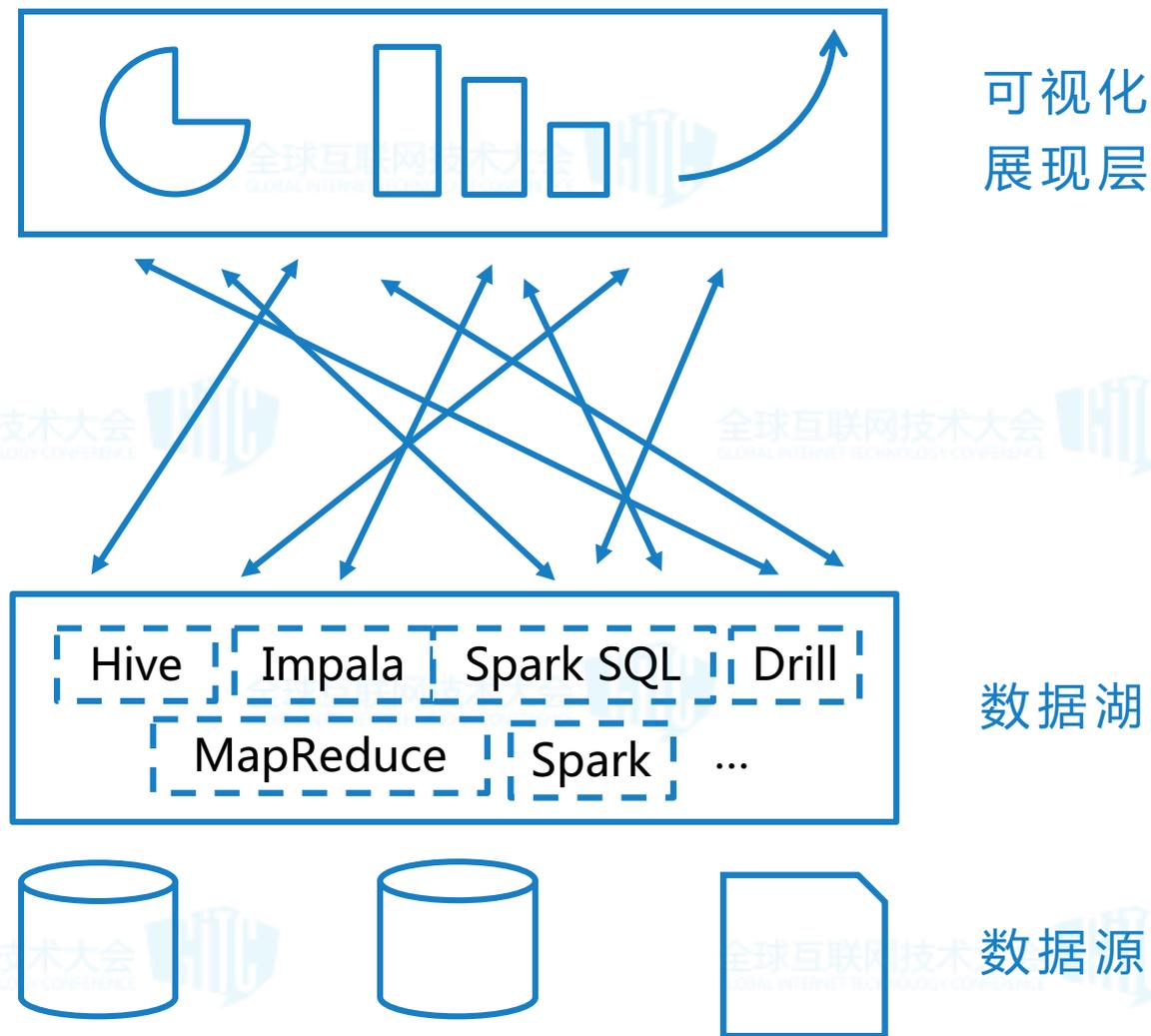
企业级
数据仓库



数据源

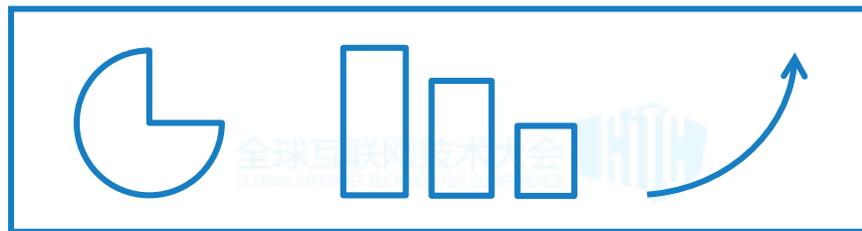
OLAP: 大数据的缺失部分

- 太多的SQL on Hadoop
- 不适合做交互式分析
- 漫长的学习曲线
- 兼容性问题
- 技术 vs 数据

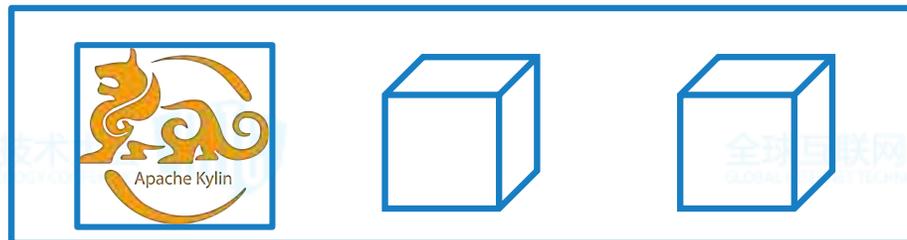


Apache Kylin: 将OLAP/DW带回大数据

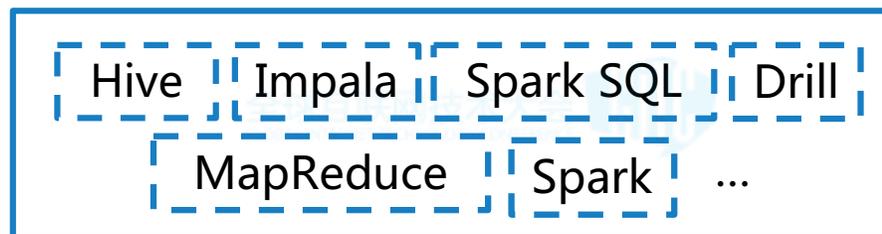
- Hadoop上的OLAP/DW
- 支持传统建模方式 (Kimball)
- 为交互式分析进行优化
- ANSI SQL
- 原生Hadoop应用
- 支持云计算
- 支持高并发、关键型应用



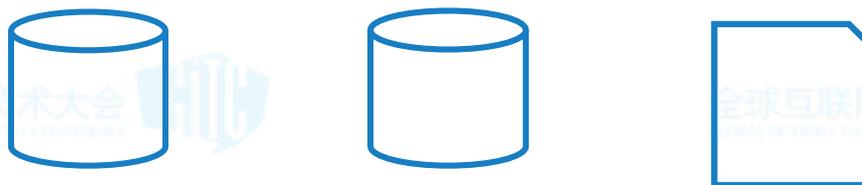
可视化
展现层



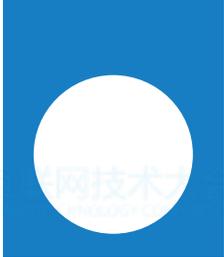
OLAP
数据集市



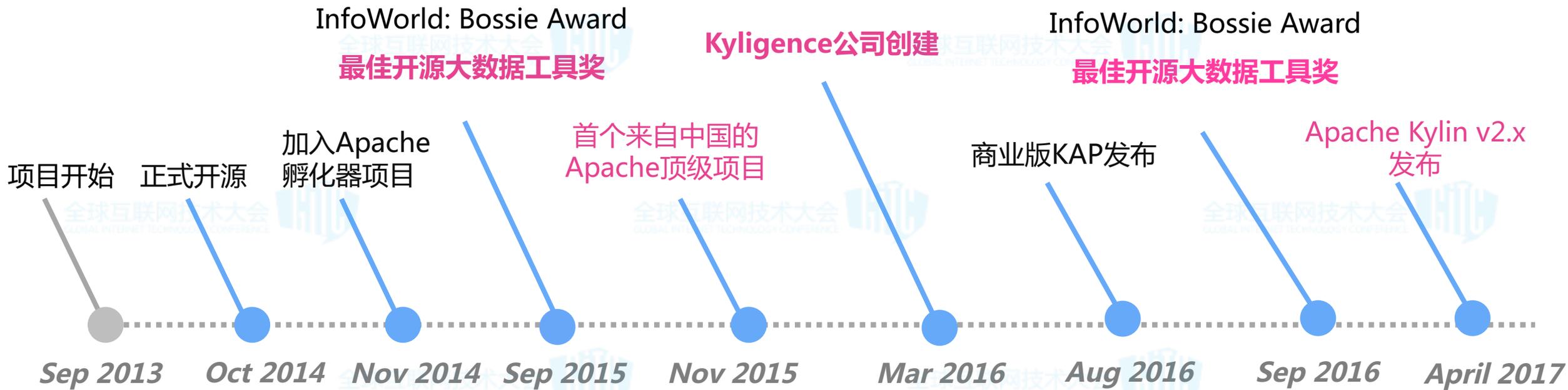
数据湖

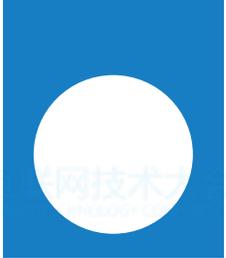


数据源



Apache Kylin 历史

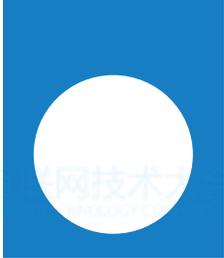




Apache Kylin全球案例

500+ 用户

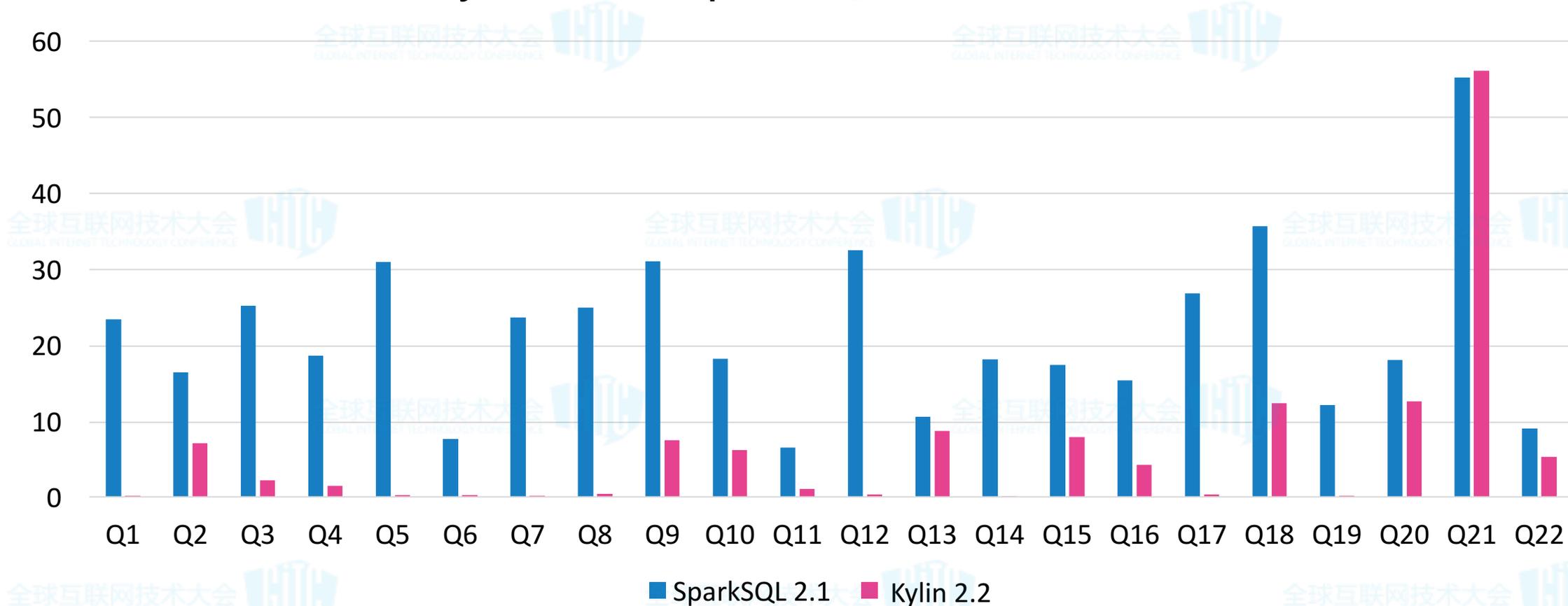




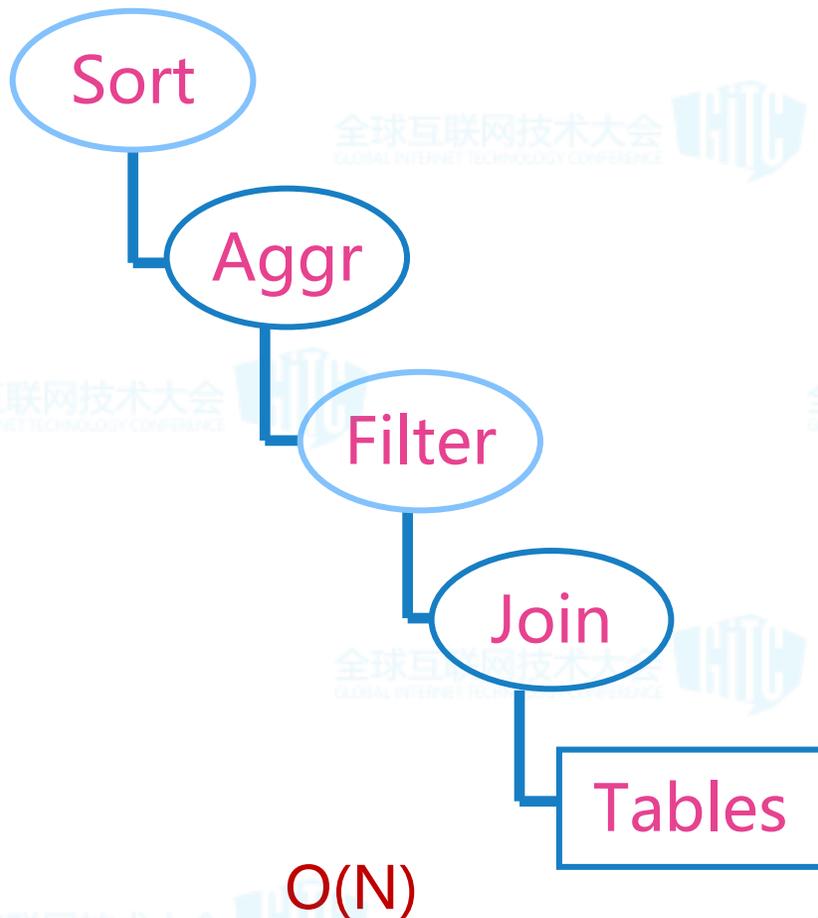
TPC-H Benchmark



Kylin 2.2 vs SparkSQL 2.1 (越低越好)



Apache Kylin 为什么快？



样例：

分析一段时间内，不同“returnflag”和“orderstatus”对应的销售情况

select

```
l_returnflag,  
o_orderstatus,  
sum(l_quantity) as sum_qty,  
sum(l_extendedprice) as sum_base_price
```

...

from

```
v_lineitem  
inner join v_orders on l_orderkey = o_orderkey
```

where

```
l_shipdate <= '1998-09-16'
```

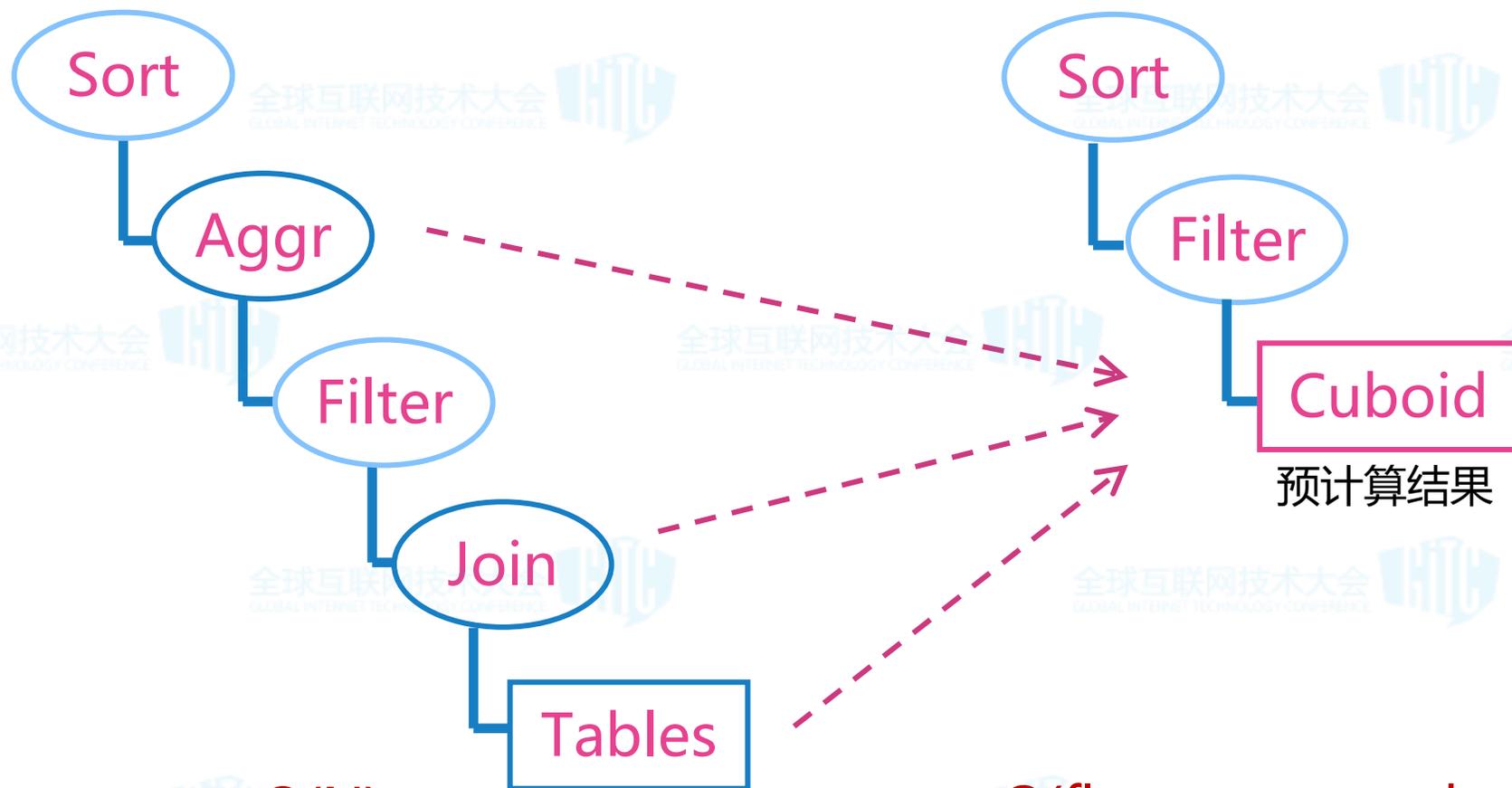
group by

```
l_returnflag,  
o_orderstatus
```

order by

```
l_returnflag,  
o_orderstatus;
```

Apache Kylin 为什么快？

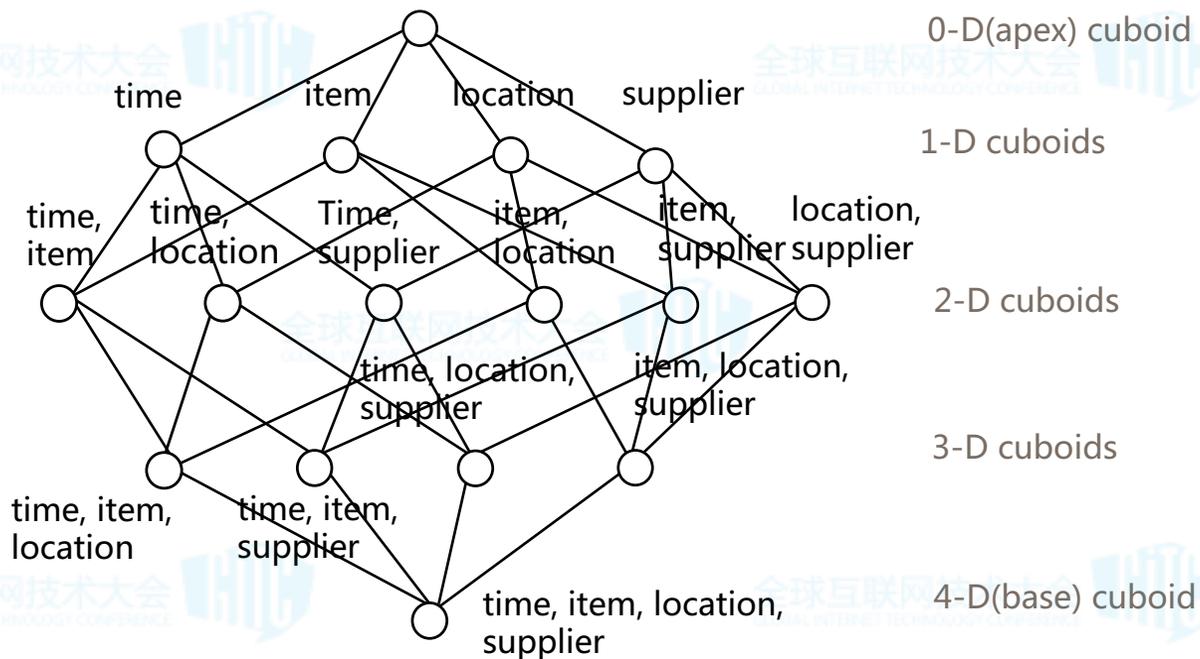
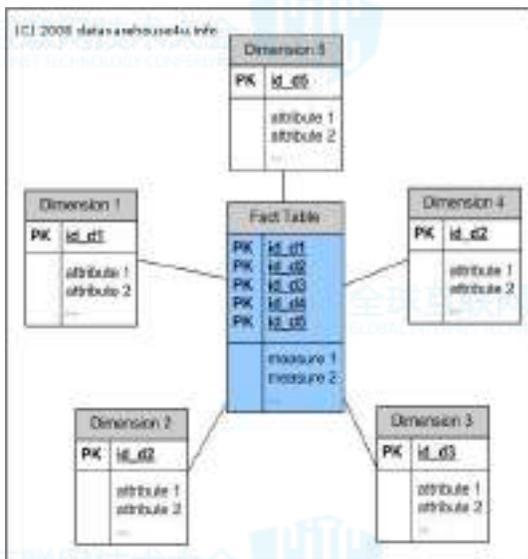


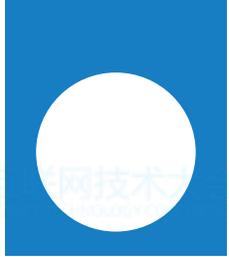
$O(N)$

$O(\text{flag} \times \text{status} \times \text{days}) = O(1)$

OLAP原理：空间换时间

- OLAP Cube 理论基础
- Model 和 Cube 定义预计算范围
- Build Engine 执行预计算任务
- Query Engine 在预计算结果上完成查询



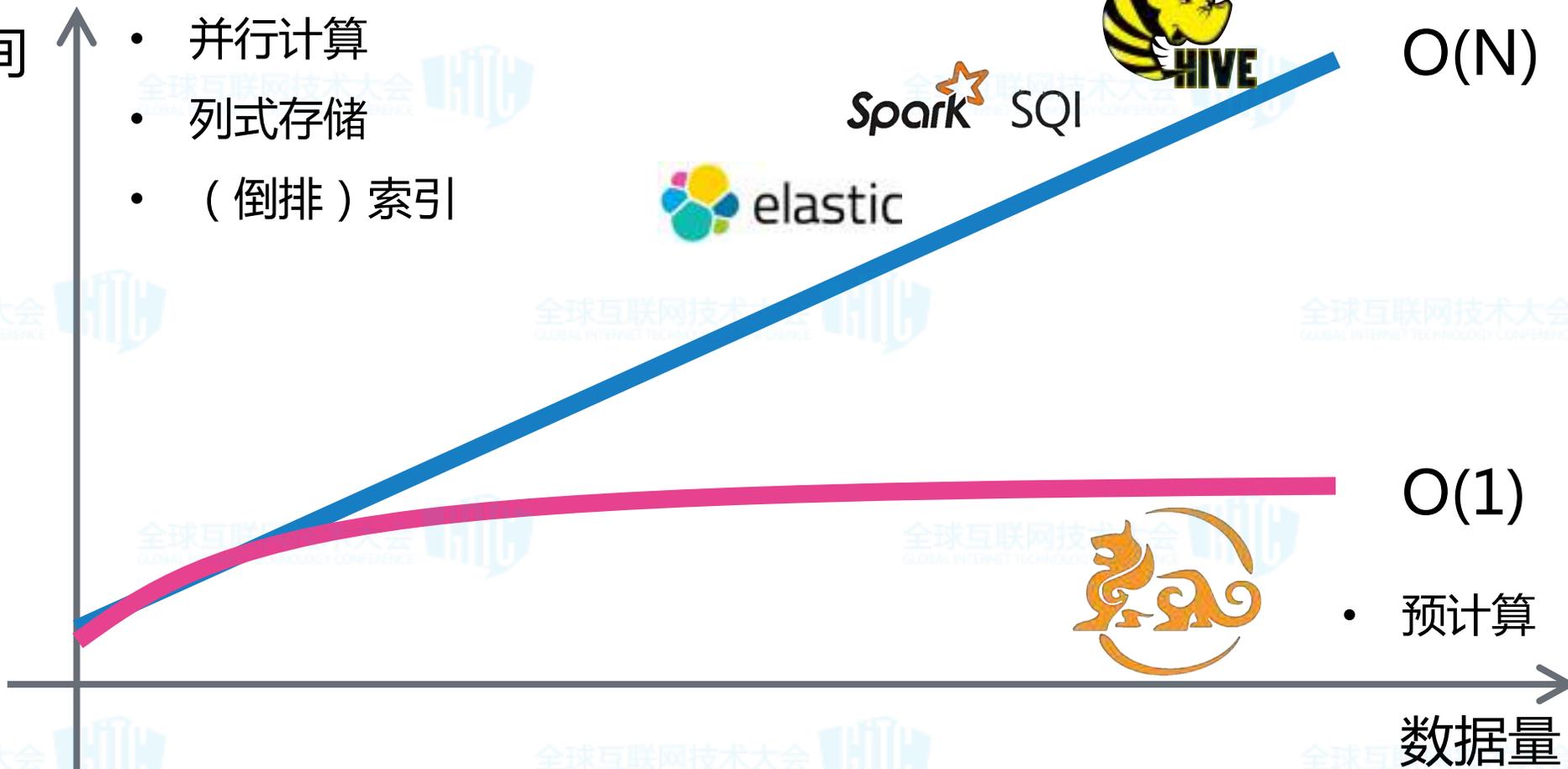


理论基础：O(1) Vs O(N)



响应时间

- 并行计算
- 列式存储
- (倒排)索引



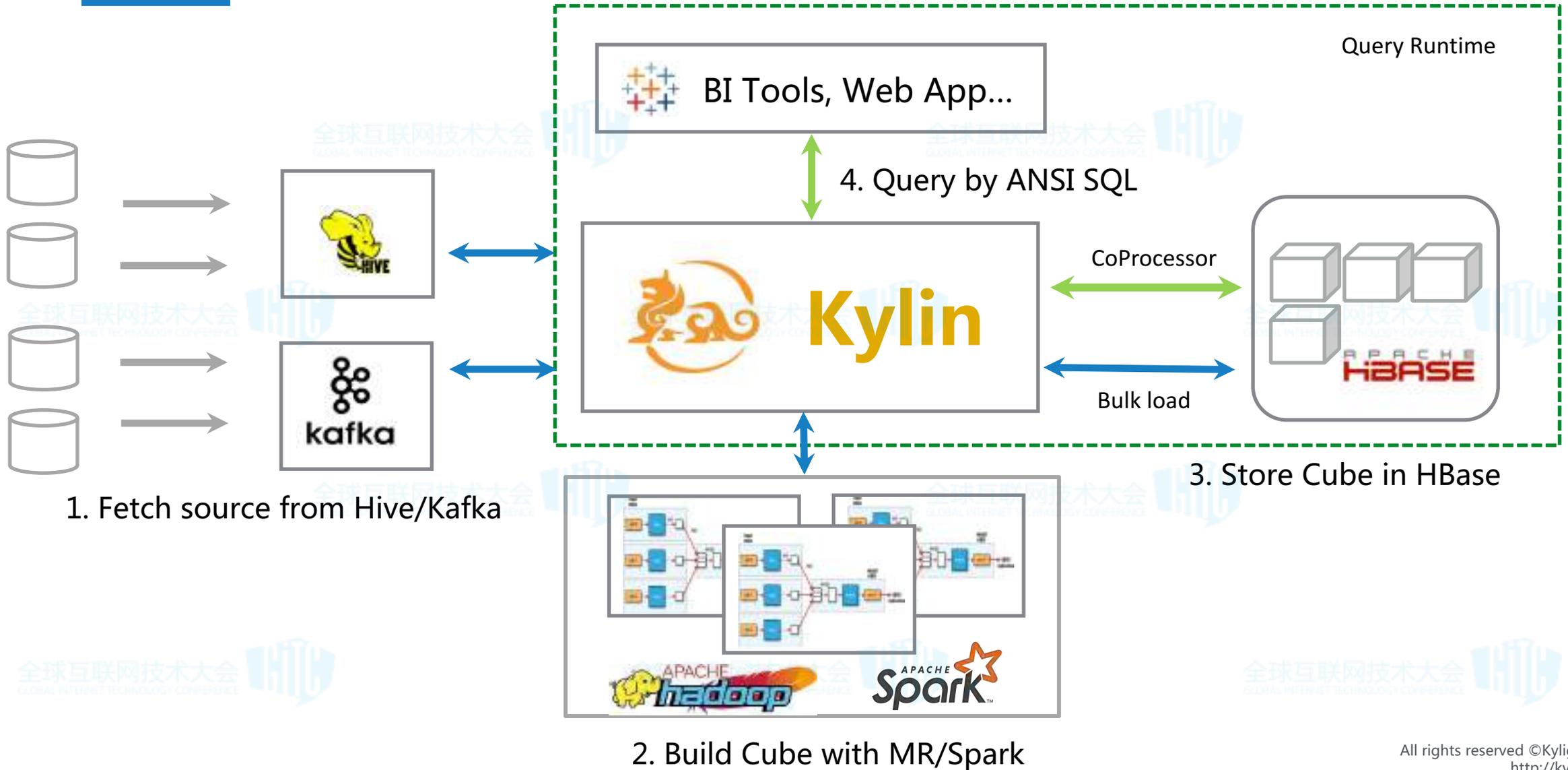
O(N)

O(1)

- 预计算

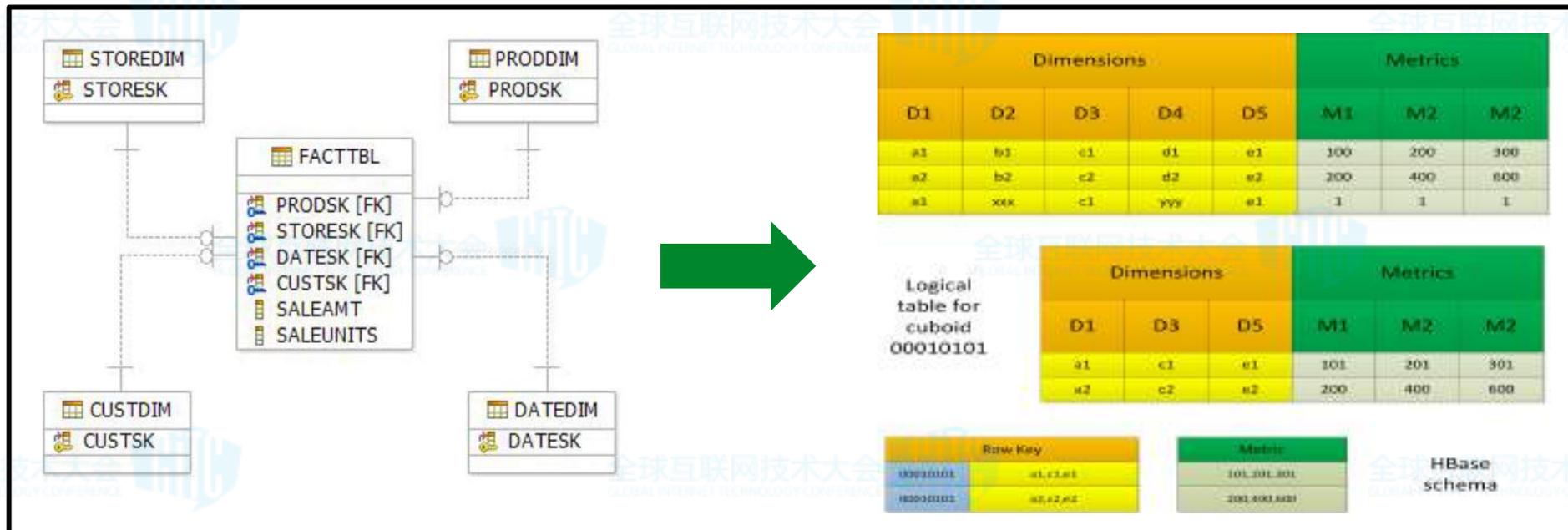
数据量

Apache Kylin 系统架构

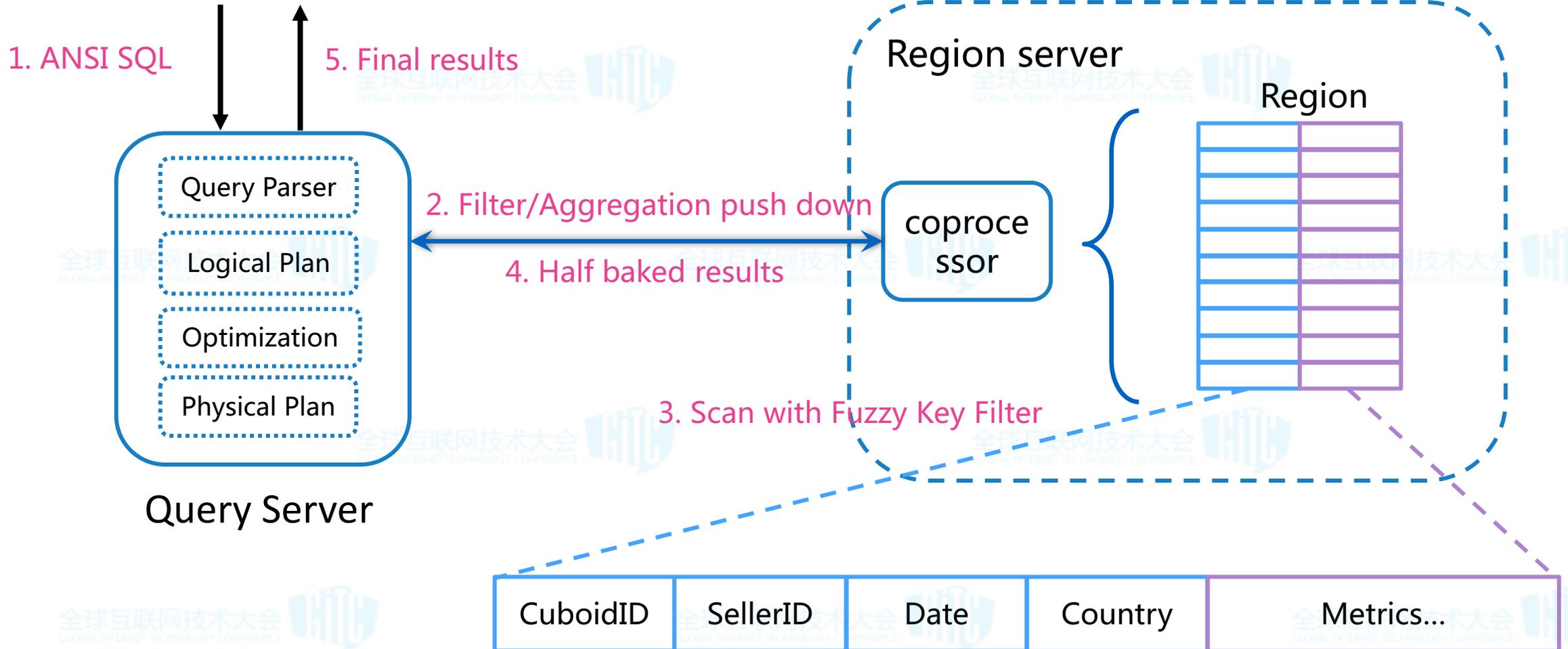


基于HBase的列式存储

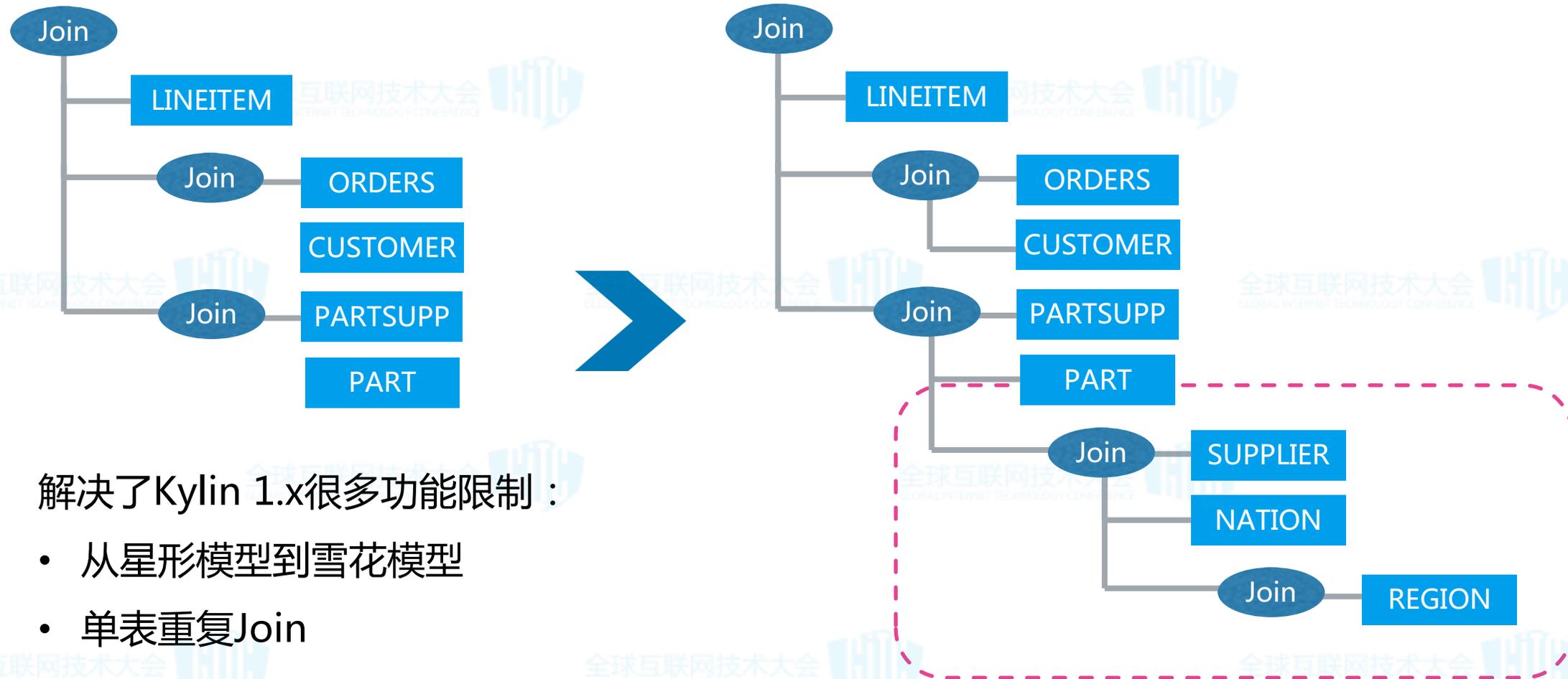
- 计算结果集保存在HBase中，原有的基于行的关系模型被转换成基于键值对的列式存储
- 维度组合作为Rowkey，查询访问不再需要昂贵的表扫描，提供高速的扫描性能
- 维度值通过编码算法（字典、定长、时间戳等）高度压缩
- 指标通过Column存储，可以灵活、无限制的增加指标数量
- 预先计算的结果为高速高并发分析带来了可能



预计算 + 后计算



Kylin 2.0 支持雪花模型

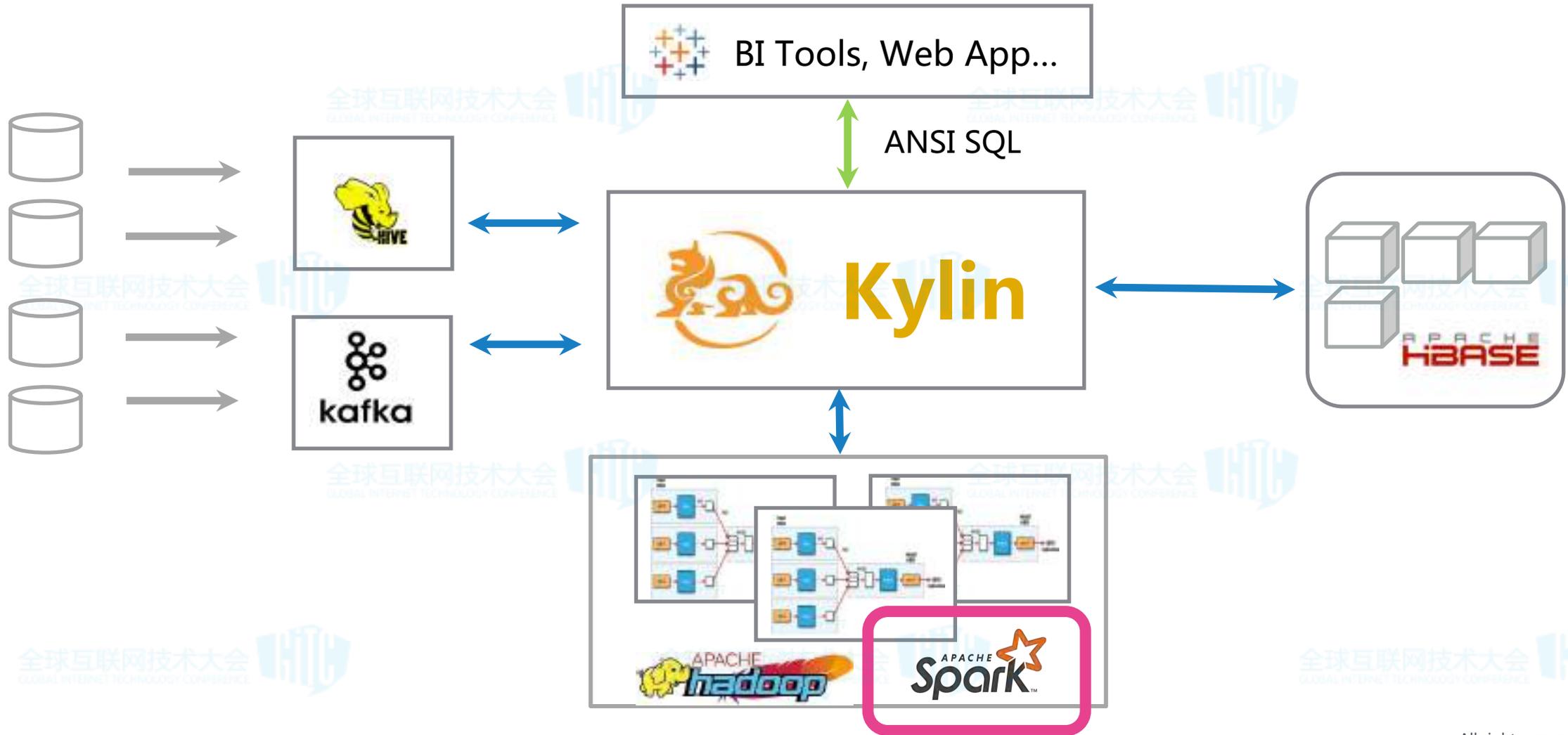


解决了Kylin 1.x很多功能限制：

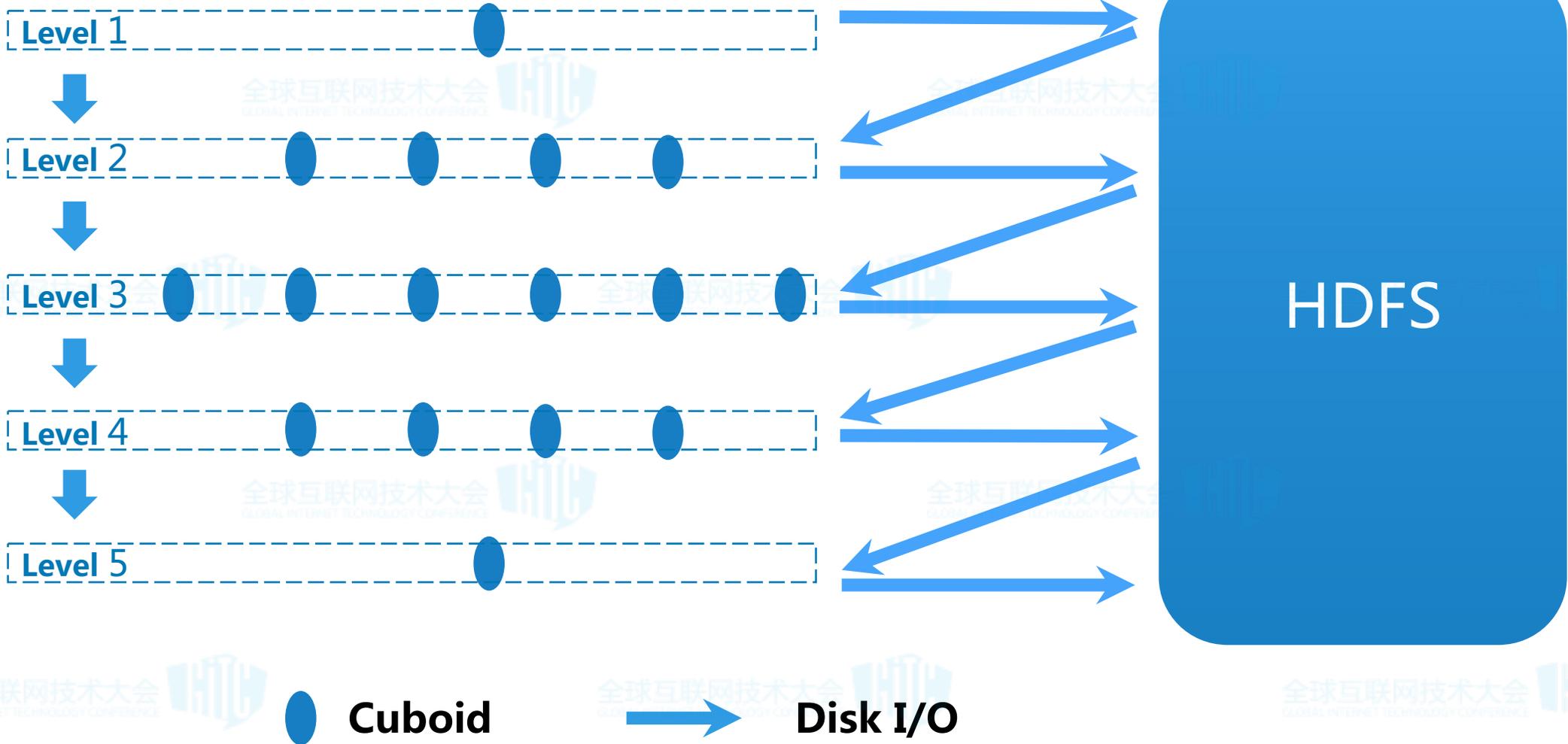
- 从星形模型到雪花模型
- 单表重复Join
-



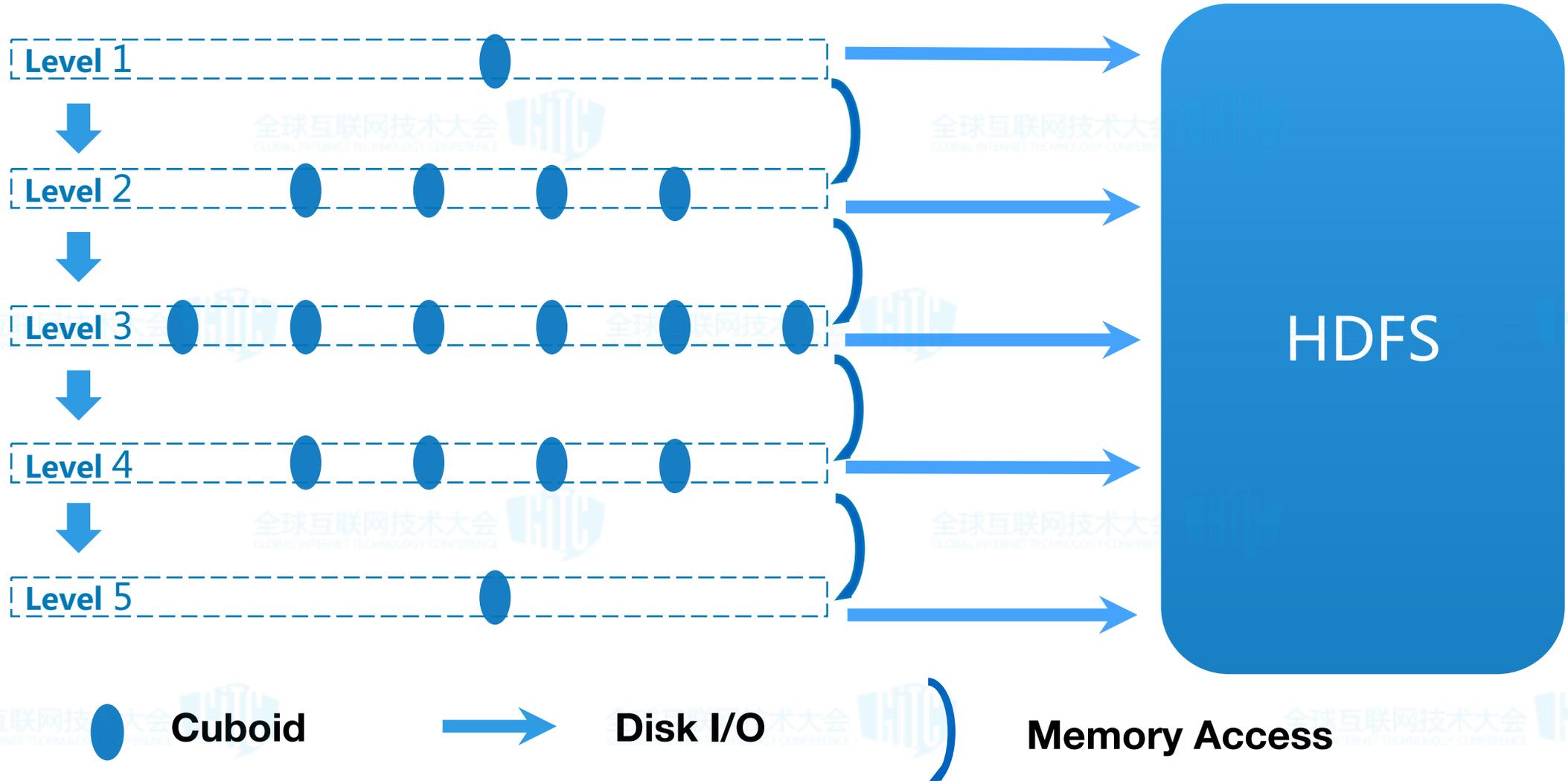
Kylin 2.1 Cubing with Spark



MR-based Layered Cubing



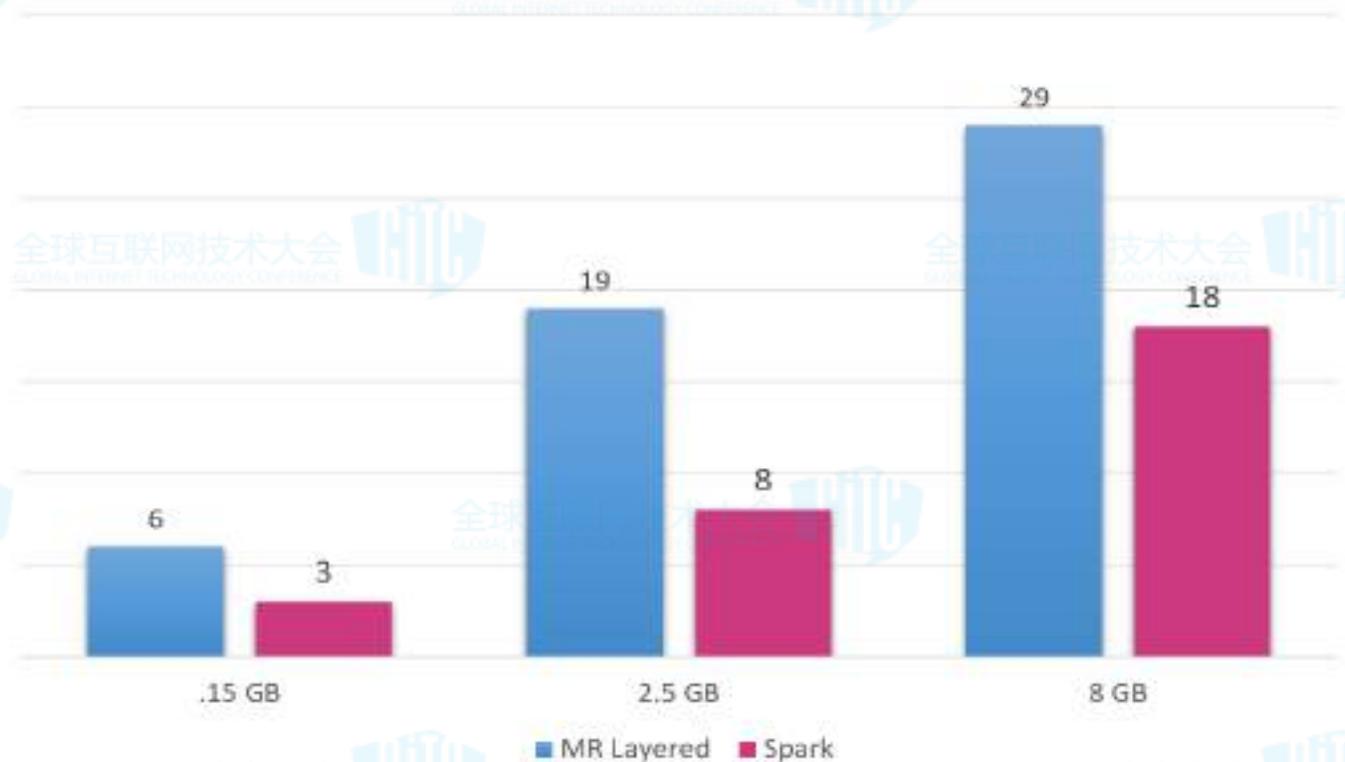
Spark-based Layered Cubing



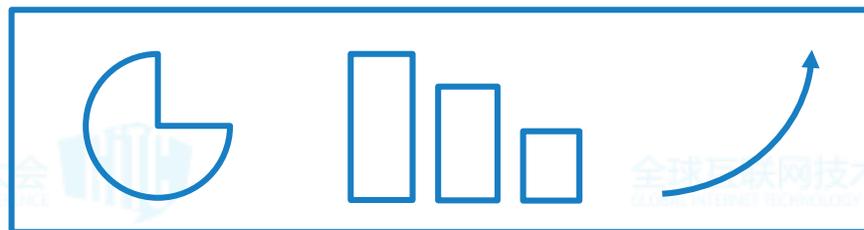
Spark Cubing vs. MR Layered Cubing

构建时间减半，但是可以观察到优势随着数据量的增加而减少

- 4 节点的集群
- Spark 1.6.3 on YARN
- 24 vcores, 30 GB memory
- 3 data sets of increasing size:
.15 GB / 2.5 GB / 8 GB



Kylin 2.2 查询下压: 支持数据探索 and 发现



Query Router

- 统一的查询入口
- 统一的ACL
- 自适应不同SQL方言
- 可扩展架构, 支持更多下压引擎
- 为模型优化收集数据

查询下压: 分钟级延迟

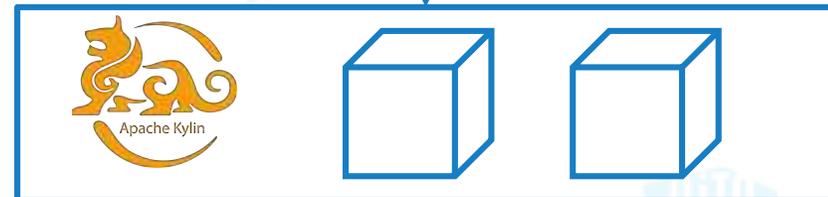
多维数据集访问: 亚秒级延迟



数据探索/发现



模型演进



关键业务分析

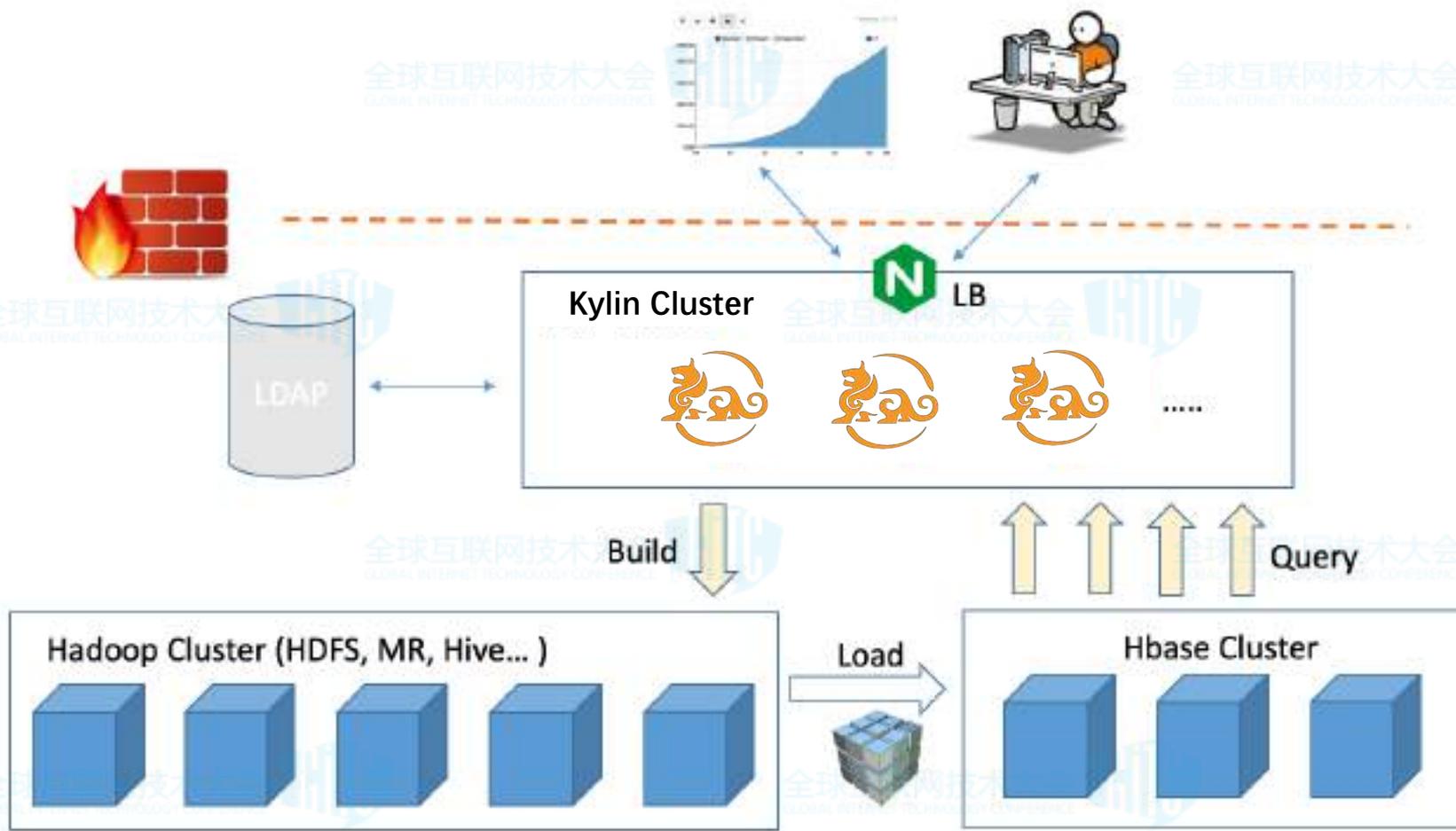


更多来自2.X的更新

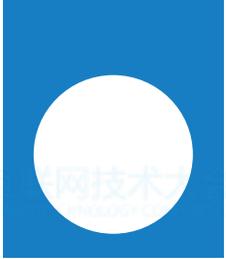


- Robust Global Dictionary
- Job Engine HA
- RDBMS as Data Source
- Project/Table ACL
- Apache Ranger Integration
- Percentile measure

可扩展的部署架构



- ◆ Job Server基于ZK高可靠部署
- ◆ Query Server无状态，基于LB进行水平扩展
- ◆ 计算集群与查询集群相分离，隔离不同 workload
- ◆ Hadoop原生应用，非侵入式部署



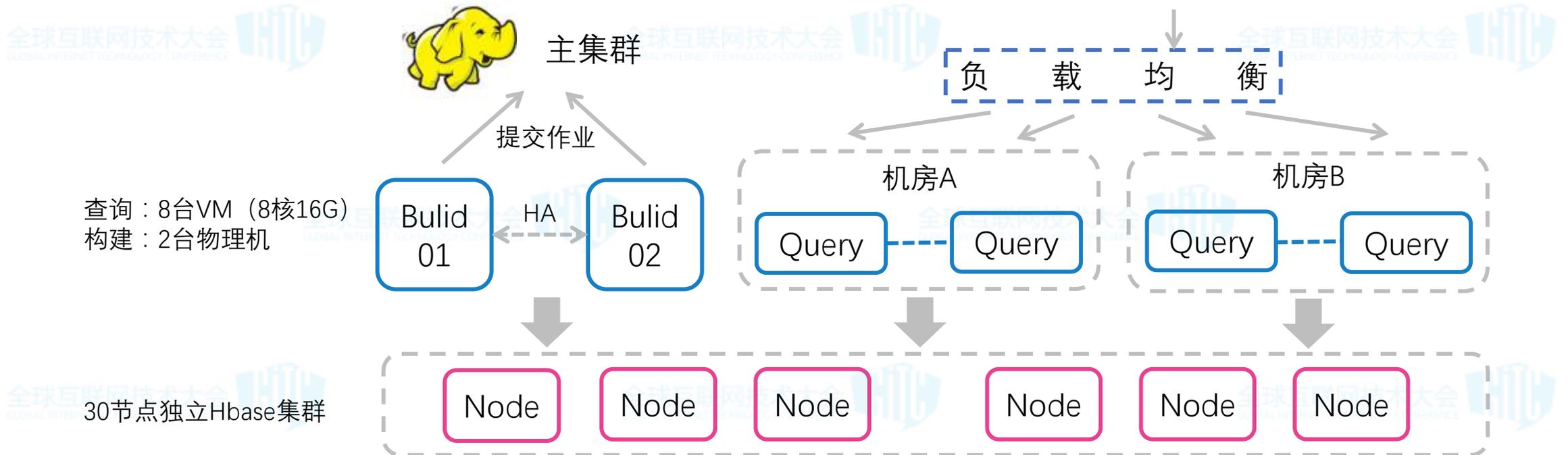
最大的O2O公司Kylin应用架构（外卖数仓）

214[↑] Cube

数据量：2853亿 Cube存储：59TB

TP50: 87ms TP99: 1266ms

50w^{次/天} 查询量





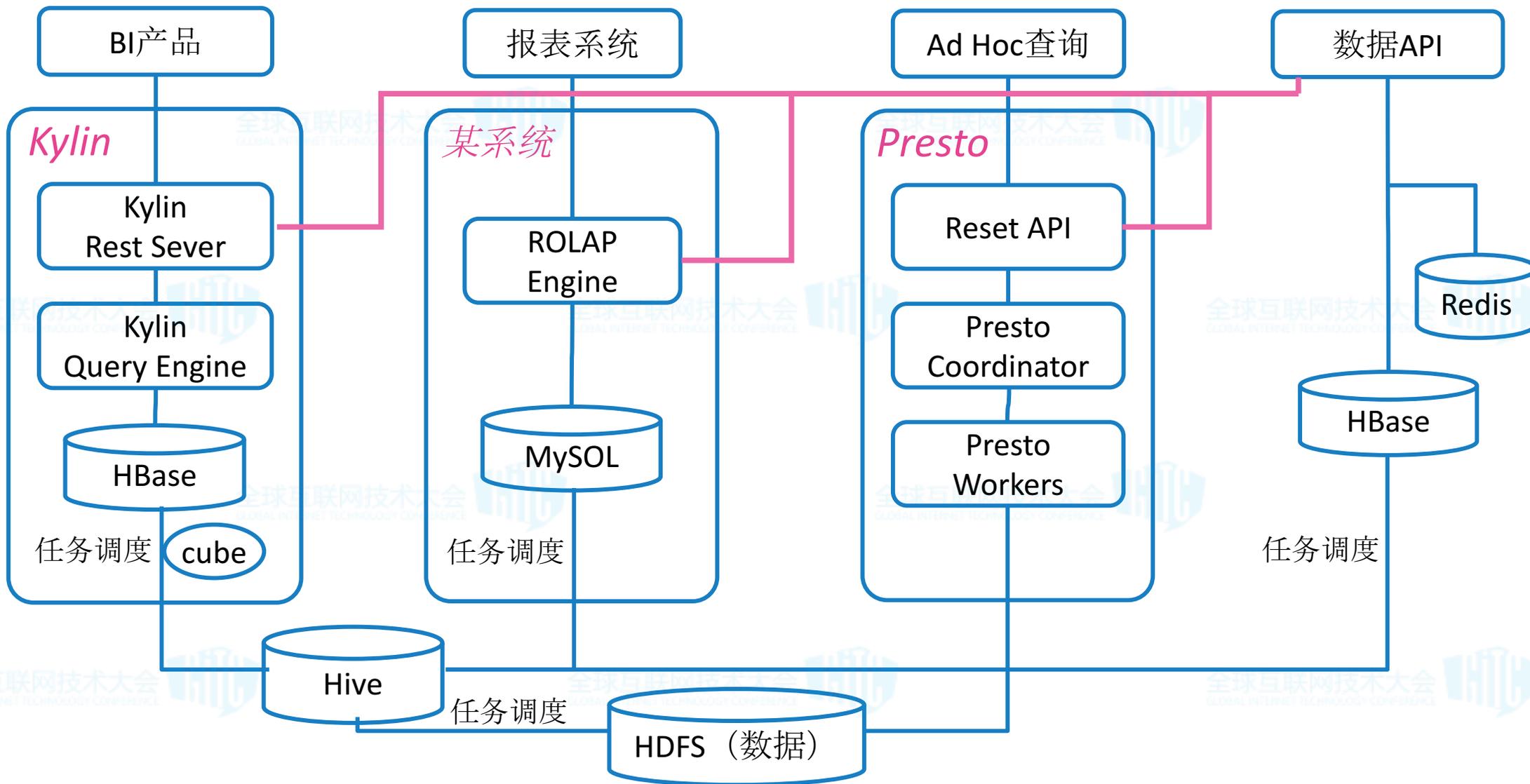
Kylin带来能力和效率的提升

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



对比项	接入Kylin之前	接入Kylin之后	对比结果
数据开发流程	需要开发事实层、聚合层、主题层和维度表ETL	只需要开发事实层、维度表、配置Kylin	<ol style="list-style-type: none">ETL开发效率提升3倍维护成本只需要改Kylin配置，不需要修改ETL
存储方式	Hive中与计算结果推送到MySQL	存储在HBase	<ol style="list-style-type: none">支持结果数据量级增大都使用JDBC查询数据结果数据传输提高1倍
多维数据处理	Hive中cube或者grouping sets	Kylin UI配置	维度变化只需要改Kylin配置，不需要修改ETL
精确去重	Hive中count distinct	基于bitmap的精确去重	<ol style="list-style-type: none">节省计算资源支持跨天去重
转化率&过户留存	Hive中通过join过滤	通过UDAF处理bitmap	<ol style="list-style-type: none">开发效率提升不需要预计算，节省资源

最大的房地产中介公司Kylin应用架构



某公司外卖业务流量分析的选型之路



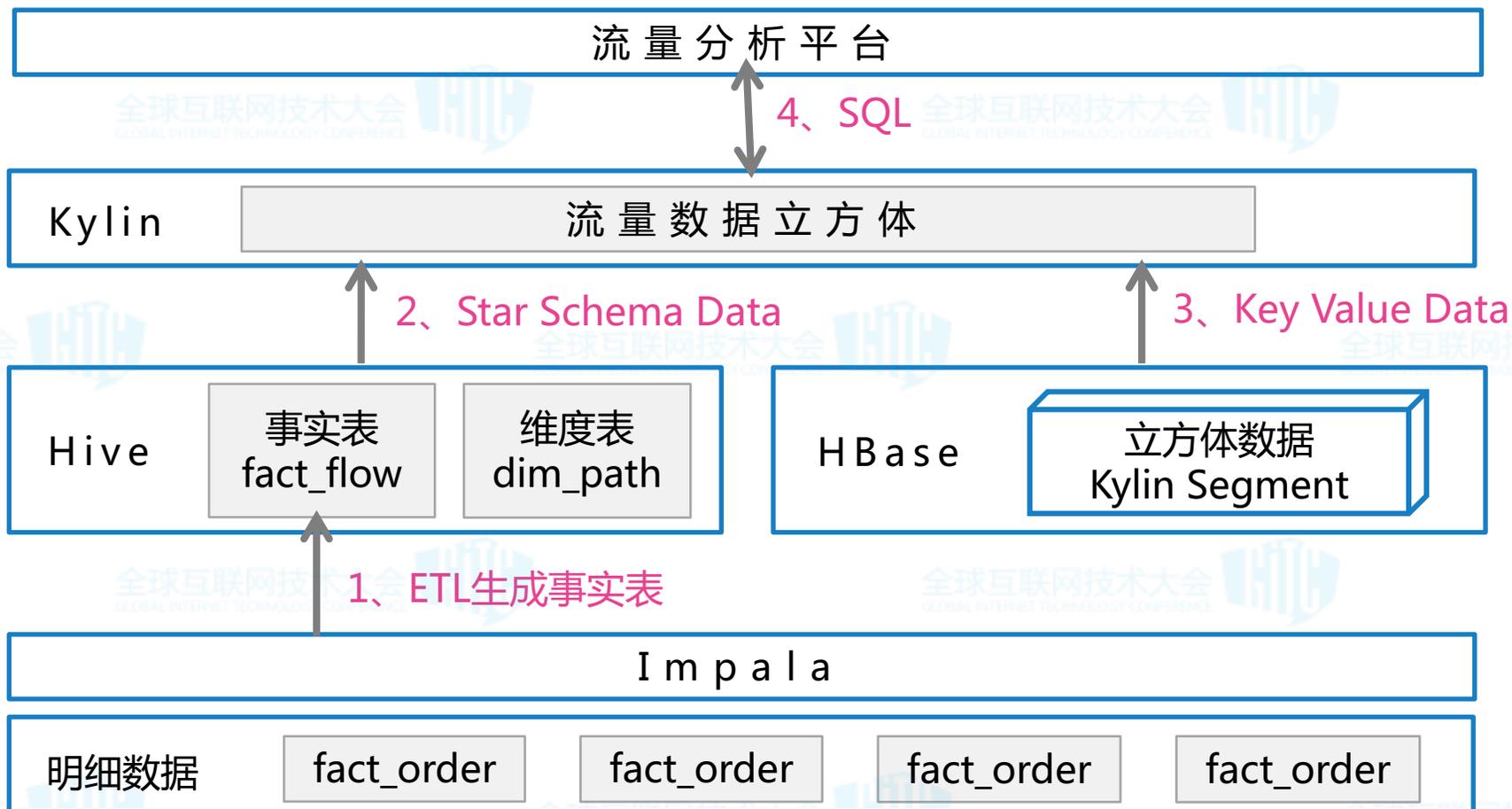
V S



- 非实时分析为主，V1.5后支持流式构建
- MapReduce、Spark计算
- 支持SQL和JDBC
- 支持精准的count distinct
- 支持历史数据回溯
- 维度膨胀

- 实时分析为主
- 基于内存计算
- 支持第三方SQL引擎
- 支持不精准的count distinct
- 历史数据不易回溯
- 无维度膨胀

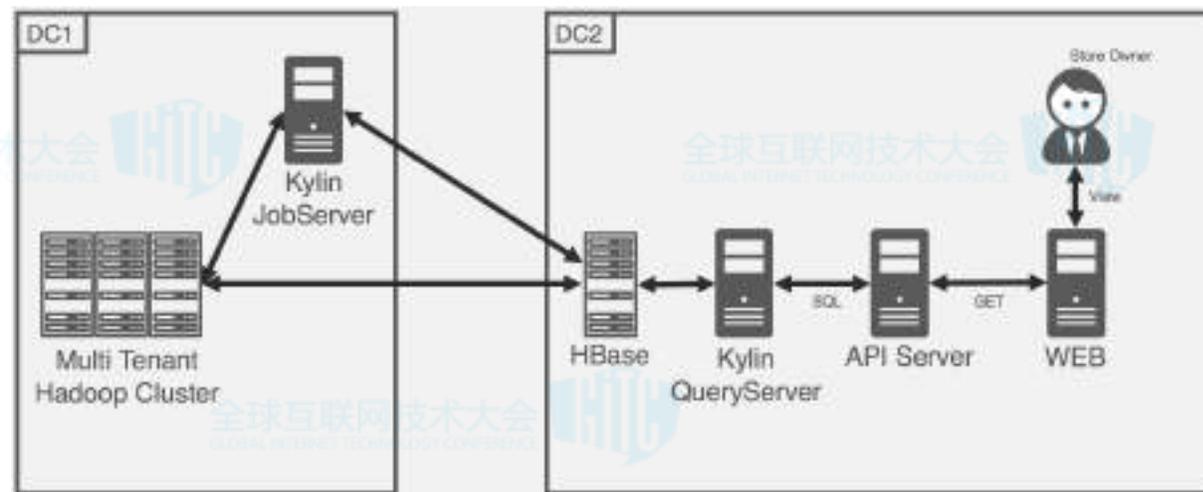
最终落地的应用架构



Yahoo日本的多数据中心部署

- 为商家提供报表统计服务
 - 包括点击率、展示率、销售情况
- 以前使用Impala作为数据后端
 - 需要大概60s才能展示Web UI
- 使用Kylin之后
 - 大部分case平均延迟小于1s
- 多数据中心
 - 美国数据中心构建索引，日本数据中心查询

数据本地化，索引全球部署





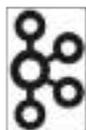
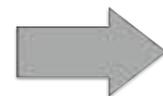
Kylin未来发展的一些展望



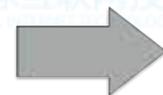
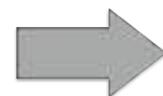
- 完全基于Spark的Cubing，更快，更敏捷
- 更多Data Source
- 将Pushdown进行到底
- Cloud Friendly (Azure HDI/Amazon EMR ready)
- Container Friendly



关于Kyligence



Apache Kafka



We are hiring!

K y l i g e n c e

Apache Kylin

dev@kylin.apache.org

Twitter: @ApacheKylin
<http://kylin.apache.org>



WeChat: ApacheKylin

Kyligence Inc

info@kyligence.io

Twitter: @Kyligence
<http://kyligence.io>



WeChat: Kyligence

谢谢!



刘一鸣
billyliu@apache.org