实时计算

数据交换平台 和仲

About me

- 花名：和仲
- 姓名：强琦
- 个人介绍：浙大机器学习方向毕业，后一直从事搜索引擎的研发工作。去年转至CDO数据交换平台，从事大数据平台的研发工作。
- 微博：阿里和仲

大纲

一、实时计算的范畴。

二、流计算服务化平台galaxy介绍。

三、多维度即时计算平台garuda介绍。

四、持续计算简介。

五、总结与展望。

实时特点

- Ad-hoc computing: 计算不可枚举，计算在query时发生。在线实时。这里的实时侧重query的实时计算。（数据的实时计算）
- Stream computing: 计算可枚举，计算在数据发生变化时发生。离线实时。这里的实时侧重实时数据的处理。（实时数据的计算）
- Continuous Computing: 计算可加（增量），大数据集的在线复杂实时计算。(实时数据的实时计算)

流计算

Stream computing特点

- 流（stream）：由业务产生的有向（渠道）无界的数据流。
  - 不可控：到达时机，相关数据顺序，质量（残缺），only once，规模，上游不可控（业务改变，渠道）
  - 时效性要求：容错方案，体系架构
  - 体系缺失：数据质量
- 处理粒度最小：对架构影响决定性
- 处理算子对全局状态影响不同：有状态，无状态；幂等，顺序相关（偏序，全序）
- （多）输出性质不同：action，state(大多数节点为commit点，少数为commit点)

难点

- 几对矛盾
  - 吞吐与响应时间（batch）
  - 实时性与数据通道不可控
  - 非幂等处理与数据通道不可控（架构复杂度）
  - 实时性与业务存储的压力
  - 精度与成本
  - 恢复成本与运行时成本
  - 易用性与表达能力
- 数据的不可控带来的架构复杂性
- 流计算的服务化
- 侵入式带来的风险
- 雪崩与抖动

galaxy

- 主要目标
  - 低开发成本：SQL;打通元数据等
  - 优化：框架，hint体系
  - 模式；并行DAG；MRM；Checkpoint；命令流
  - 服务化：调度，隔离，雪崩与抖动
  - 体系化*
- 准确度
  - 可接受不准确；精确；只多不少；*保序

三个层次

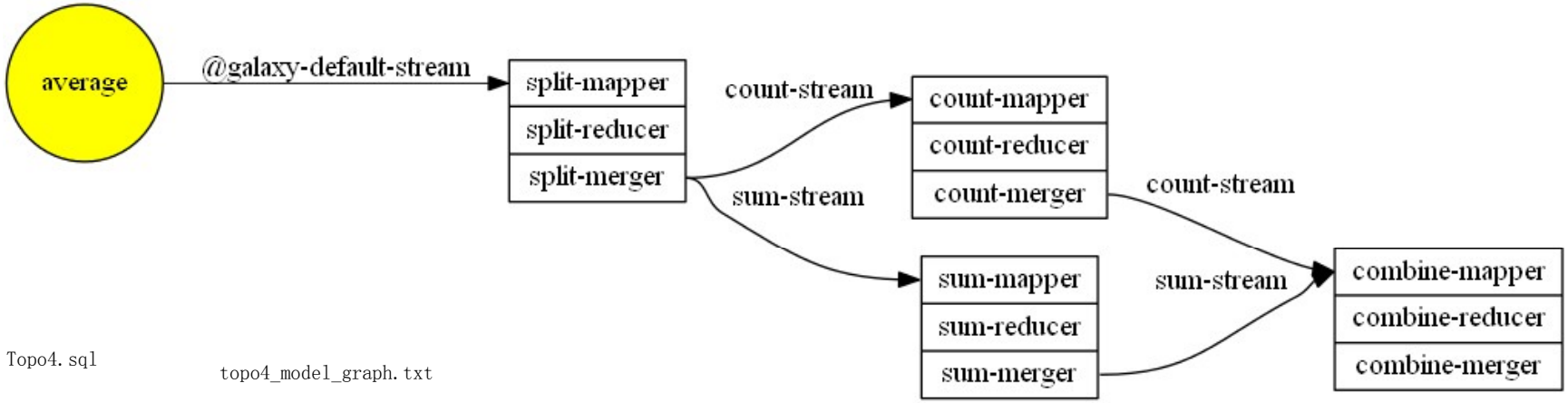
- SQL
  - Udf
  - Udaf
  - Udtf
  - window
  - 算子层
- 语义层
  - Map
  - Reduce
  - Merge(Rollback)
- sourceCode

SQL

- SQL
  - CREATE STREAM stream_name
  - CREATE DIM TABLE dim_name
  - CREATE CACHE TABLE AS SELECT [ALL|[col1[udf(col2),…]]] from DIMTABLE WHERE conditions WITH(cache_parameter=value[,……])
  - CREATE WINDOW[]
  - CREATE RESULT TABLE result_name
  - CREATE TMP TABLE tmp_tablename
  - SELECT [* | expression] [[AS] output_name] [, ...] [FROM from_item [alias] with [window(…)] [[left|full outer] join ...] on join_condition] [WHERE condition] [GROUP BY [group_expr [, ...]]|] [[UNION ALL] select] [TOP N by expression[ASC|DESC] [,……]] With(select_parameter=value[,……])
  - UDF，UDAF，UDTF

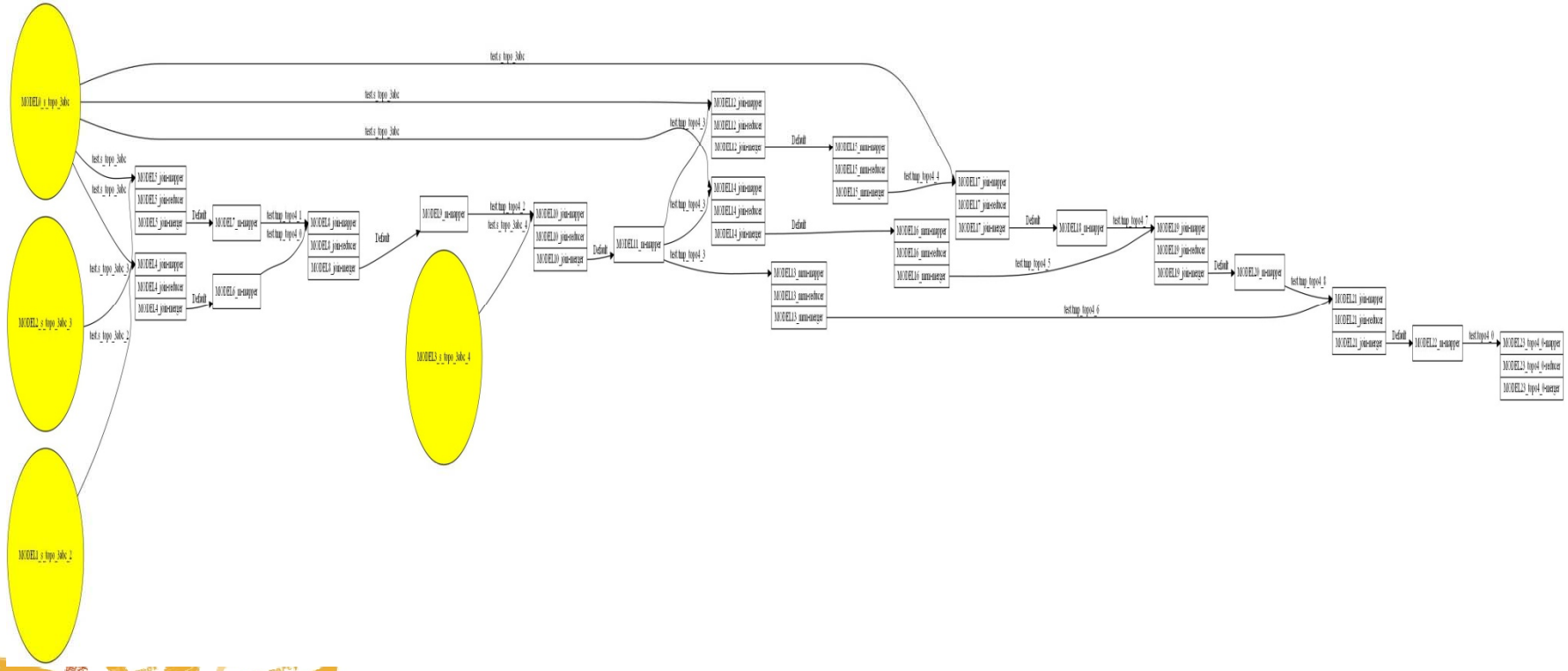
MRM

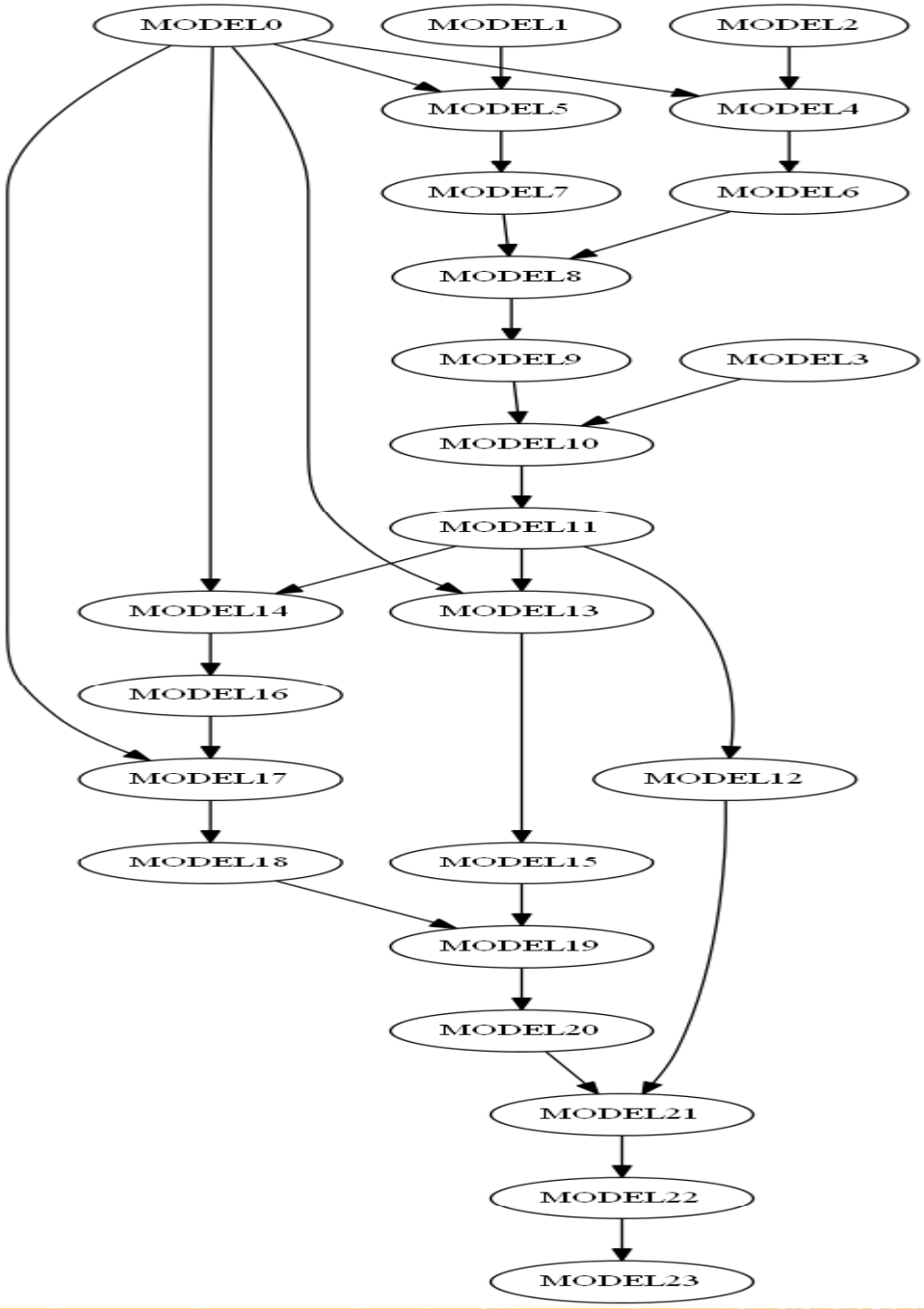
- Map
  - map
- Reduce
  - reduce
- Merge
  - merge
  - rollback
- mapOnly, mapreduce, mrm
- DAG



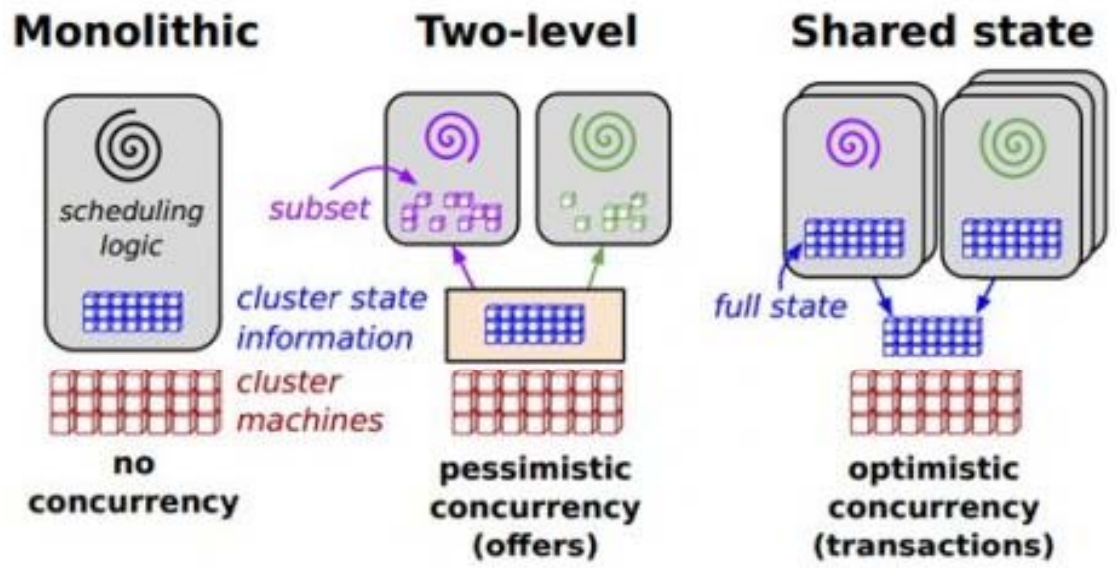
Topo4.sql

topo4_model_graph.txt




MODEL0
MODEL1
MODEL2
MODEL5
MODEL4
MODEL7
MODEL6
MODEL8
MODEL9
MODEL3
MODEL10
MODEL11
MODEL14
MODEL13
MODEL16
MODEL17
MODEL12
MODEL18
MODEL15
MODEL19
MODEL20
MODEL21
MODEL22
MODEL23

调度



- Galaxy的调度
  - 包租公模式
  - 二层

流计算与全量计算

- Case

```
t1 = select a, sum(b) as b' from t0
group by a;
t2 = select count(a) from t1 group by
b' /10;
```

-

	载体	生命周期	容错监控	单个计算单元	DAG
全量	进程	Partition数据处理完，进程"退出"	进程	重	串行
流计算	进程，线程	Keep alive	数据	轻	并行

流计算与全量计算

- 准确度
  - 全量：无
  - 流计算：可接受不准确；精确；只多不少；*保序；
- 范围
  - 全量：LOCAL
  - 流计算：业务依赖数据；跨批中间结果；业务结果
- 上下游
  - 全量：all ready
  - 流计算：不可控
- 流量复用
  - 全量：不迫切
  - 流计算：迫切
- ...

系统级

- 控制命令
  - 打通维表；打通天网
  - gentleClose
  - Tick
  - Yy
  - …
- State的屏蔽
- 抖动与雪崩
  - Replay, replay，reCcomputing
- 屏蔽TID逻辑&负流
- Memtable的引入
- Checkpoint的引入
  - 迁移
  - 集群扩容
  - 提升吞吐
  - 二个约束：
    - Cp(i)一定在cp(i+1)之前完成；
    - Cp(i)一定会在本cp前的最后一个batch结束后才进行。

checkPoint

几对矛盾

吞吐与响应时间（batch）

实时性与数据通道不可控

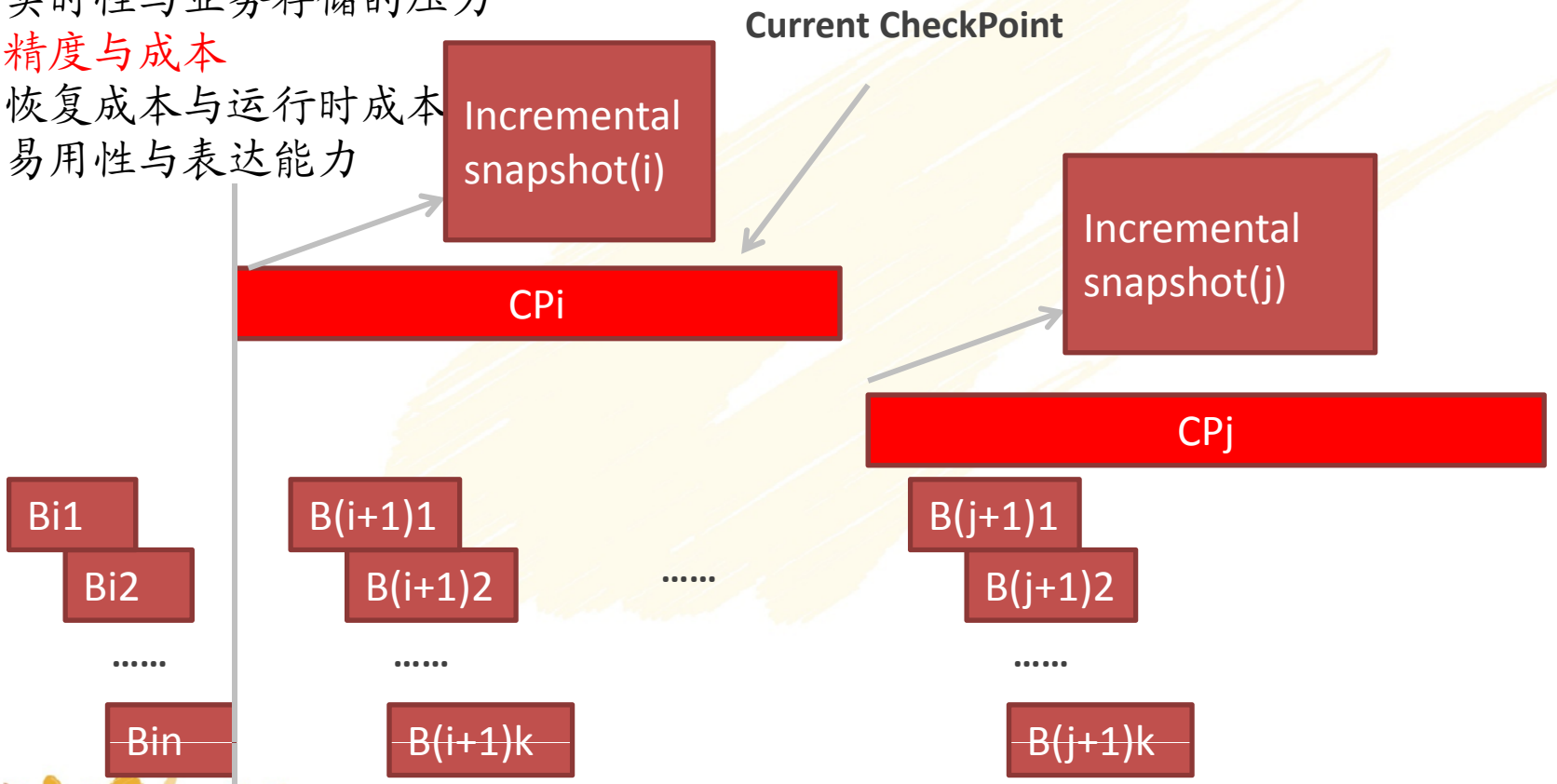
非幂等处理与数据通道不可控（架构复杂度）

实时性与业务存储的压力

精度与成本

恢复成本与运行时成本

易用性与表达能力



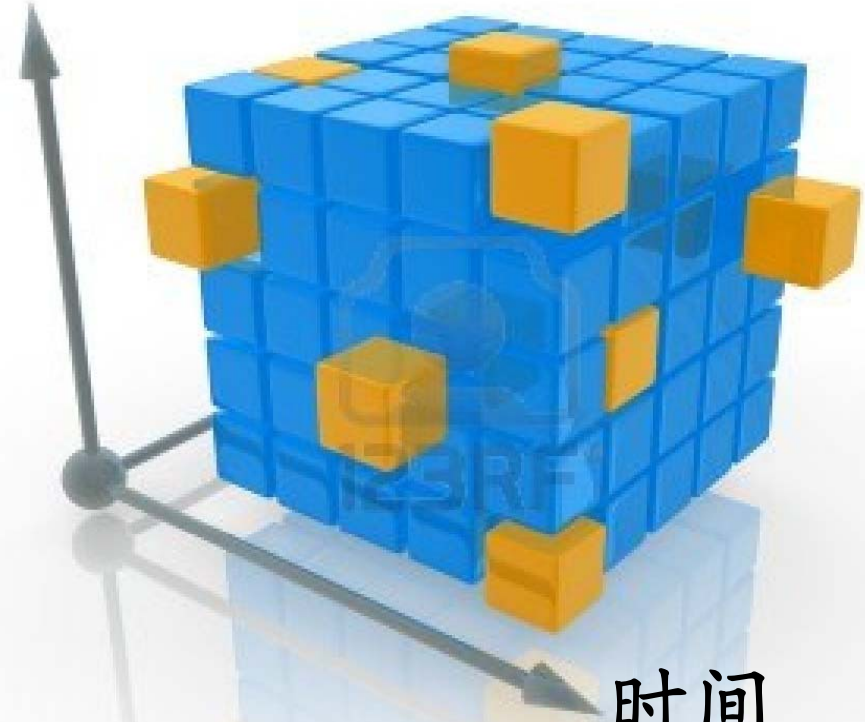
其它

- API
- Stg, dev, pre, prd
- 回流服务
- 复用流量
- Metric
- 集成测试框架
- 告警
- 计量
- 审计
- 安全
- 调度*
- 隔离*

Garuda

即时计算场景

....



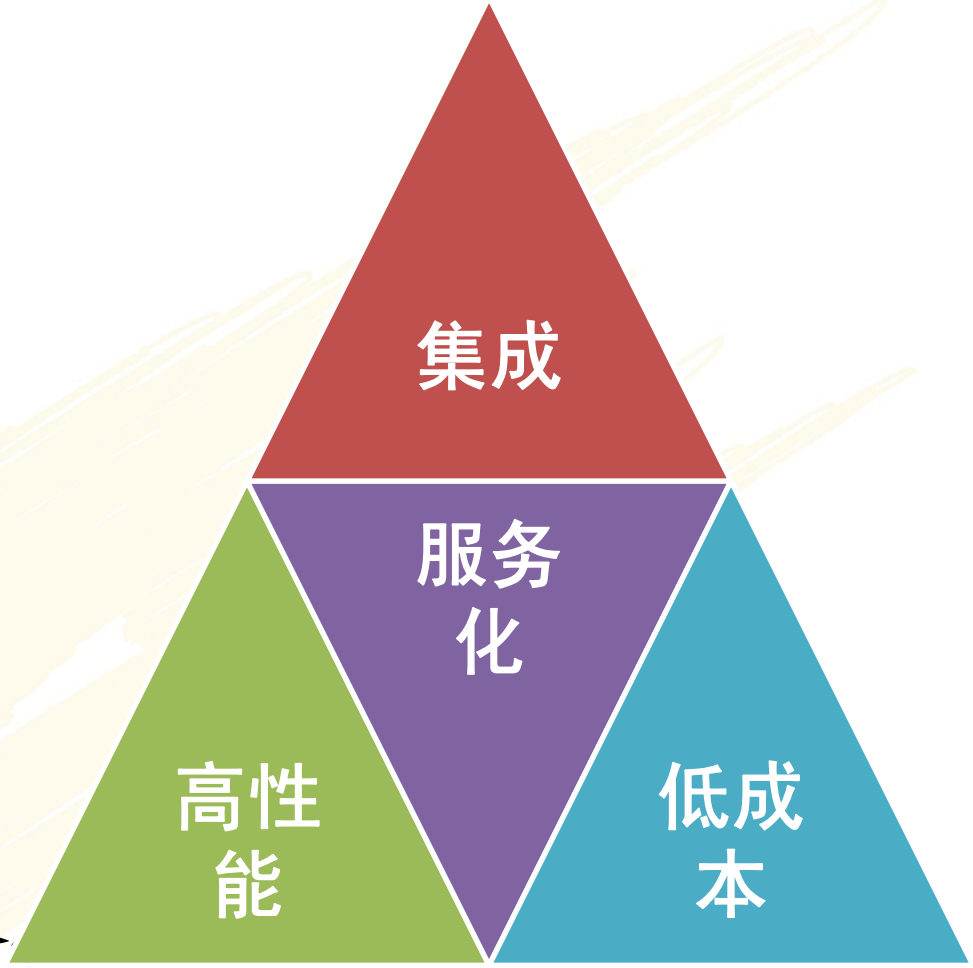
时间
省份
地市
性别
年龄

PV/UV
热门频道/栏目

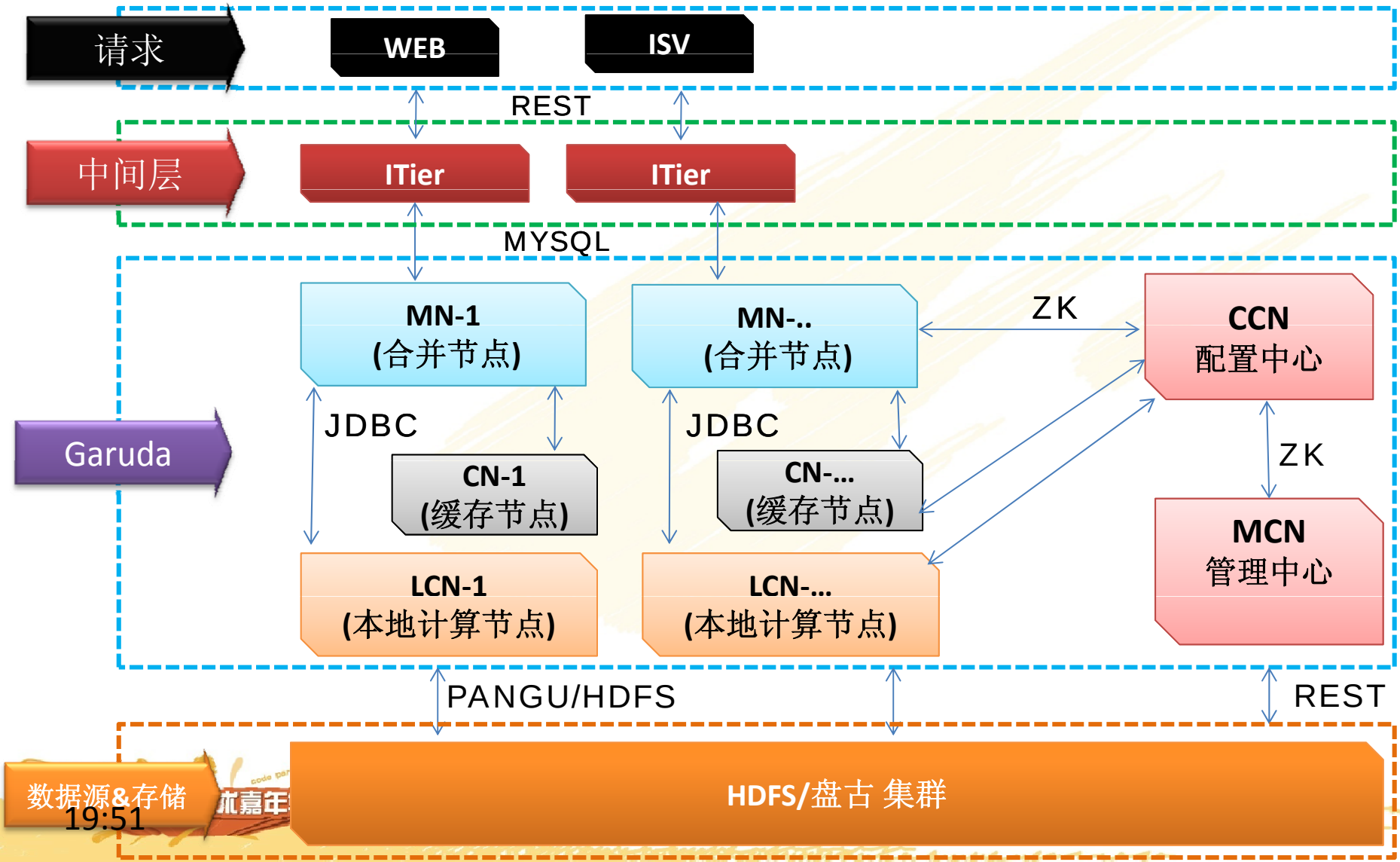
garuda

- 功能
  - 任意维度组合查询、统计
  - 支持1000万级数据导出
  - 支持大表Join
  - 支持contains(oracle intersect)*
  - 支持长周期历史数据*(年)
  - 多分区模型(Range/Hash/List)
- 其它
  - SQL/MySQL Protocol(case when/if/like/UDF/……)
  - 高并发/低延时
  - 高可用/过载保护
  - 服务化离线Build

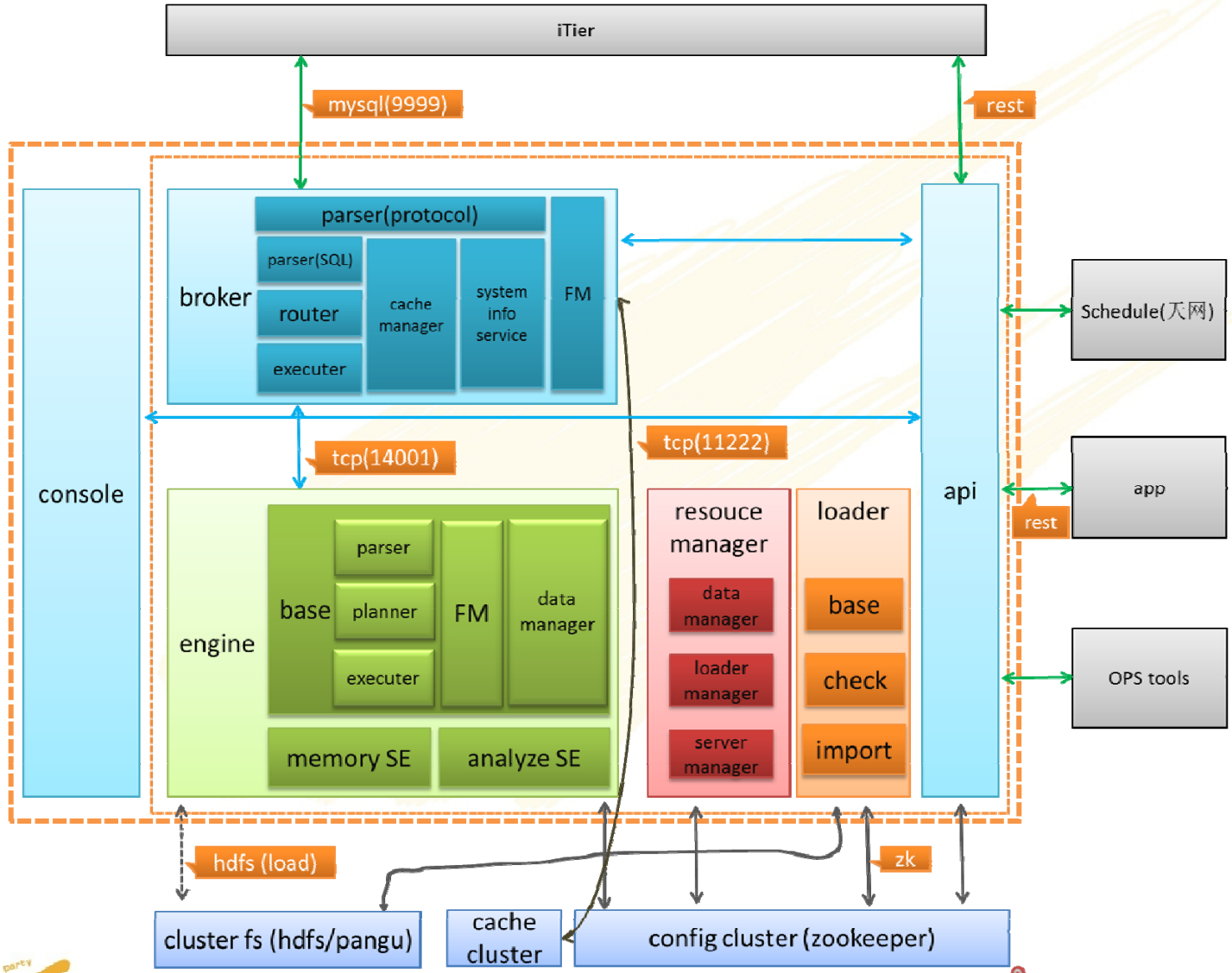
- 功能:
  - 实时数据源
  - 稀疏列
  - 完整子查询
  - 存储过程
- 性能&其它
  - SSD 支持
  - 离线建索引（飞天）
  - 计量/安全/API/服务
  - 低成本



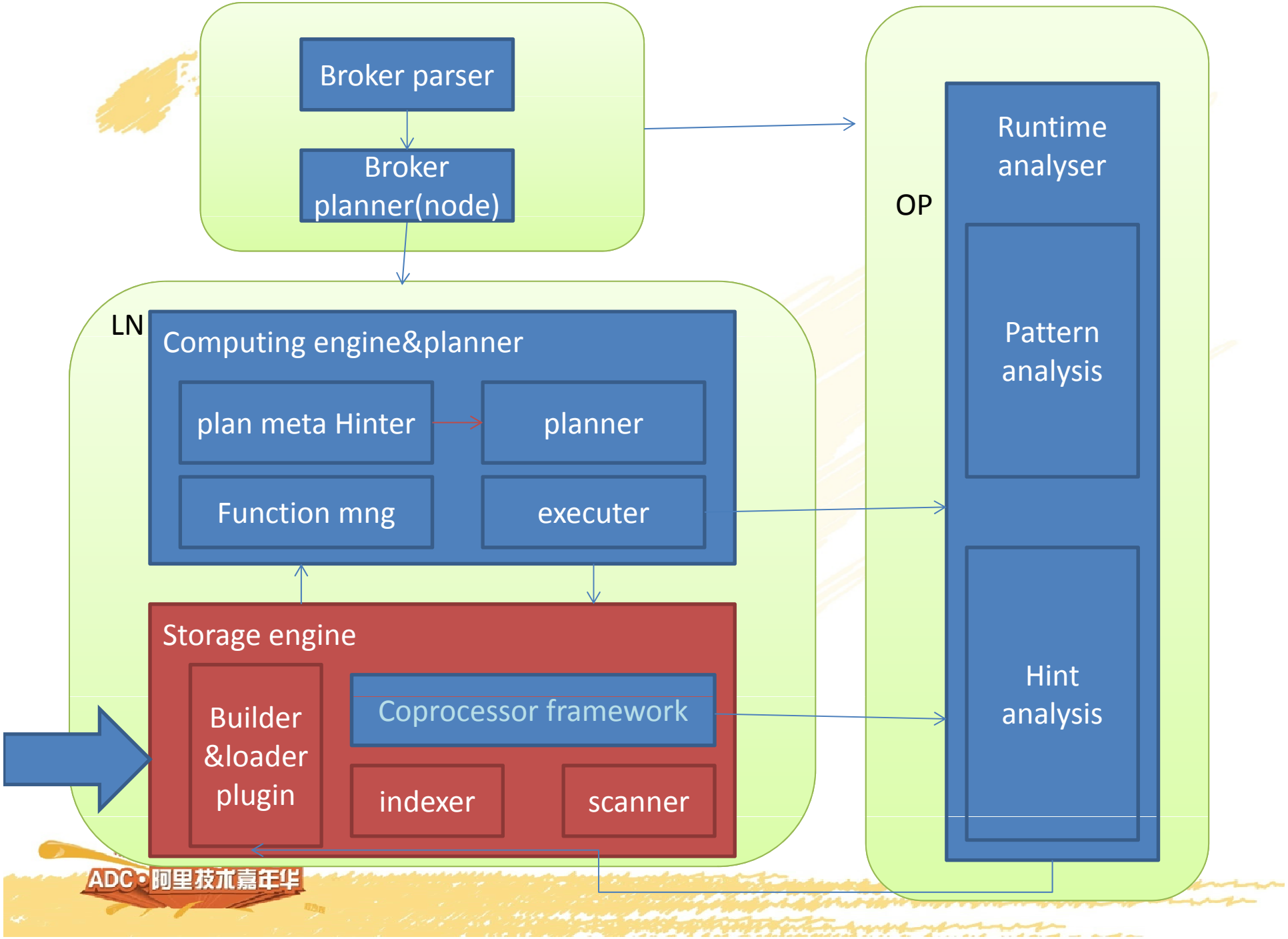
架构总览


请求
WEB
ISV
REST
中间层
ITier
ITier
MYSQL
MN-1
(合并节点)
MN-..
(合并节点)
ZK
CCN
配置中心
JDBC
JDBC
Garuda
CN-1
(缓存节点)
CN-...
(缓存节点)
ZK
MCN
管理中心
LCN-1
(本地计算节点)
LCN-...
(本地计算节点)
PANGU/HDFS
REST
数据源&存储
HDFS/盘古 集群

架构总览



SE:Storage Engine,CBO:Cost Based Optimizer,FM:Fluid Manager


Broker parser
Broker planner(node)
OP
Runtime analyser
Pattern analysis
Hint analysis
LN
Computing engine&planner
plan meta Hinter
planner
Function mng
executer
Storage engine
Builder &loader plugin
Coprocessor framework
indexer
scanner

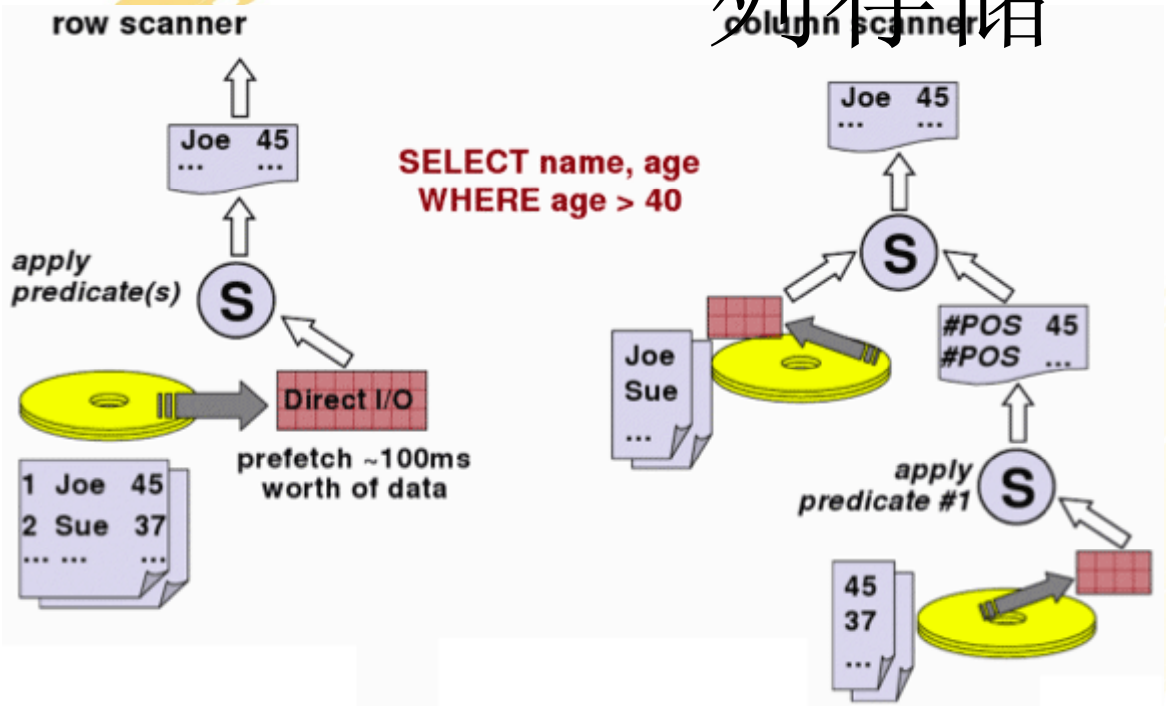
核心要点

随机访问	• 内存(稳定高频数据) • SSD
业务分区 本地计算	• 避免网络交互 • 降低时延 • 提高并发
压缩	• 减少内存占用 • 降低磁盘IO
缓存	• 提高并发
Failover 加载	• 服务稳定性 • 提高可用性

关键特性

- 列存储
- 分区
- 本地计算
- 大表Join
- 缓存
- 压缩
- 资源调度 &可用性
- 过载保护

列存储



存储

- 更少扫描(只选择需要的列)/顺序访问
- 稀疏列

压缩

- 内存压缩 38%
- 磁盘压缩 6%

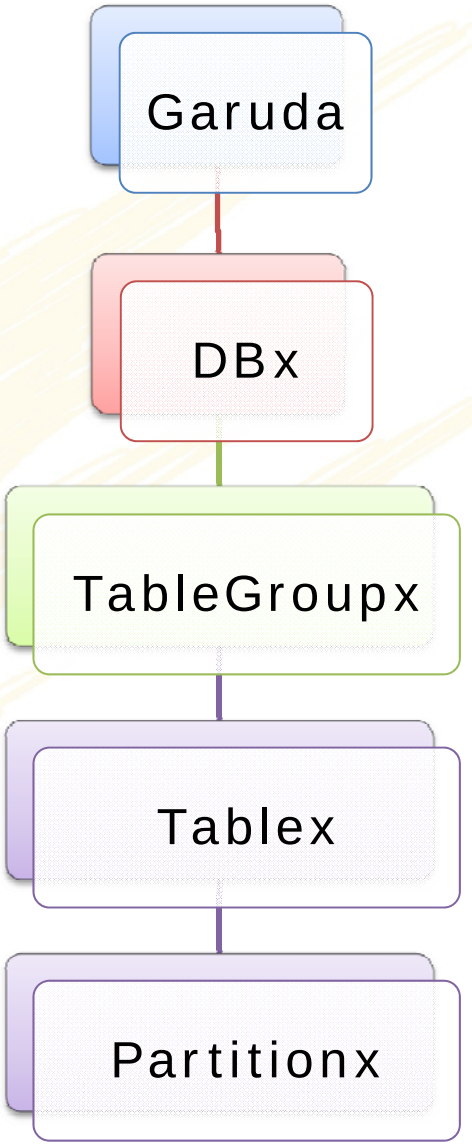
分区

分区

- 本地计算
- 减少网络交互

分组

- 便于Join



本地计算-Distinct

- Count(distinct user_id)
- ….

1	0	1	0	1	0	0	0	0	1
1	2	3	4	5	6	7	8	9	10

1）user_id 全局编码保持有序

2）计算每个分区bitset

3）bitset压缩

4）bitset求交 || 求和

本地计算-top k

- Select type_id, count(id) from t group by type_id order by count(id) desc limit 10
- 精确度换latency、concurrent

Num of top-k	Size of output	Num of guaranteed	Guarantee	Precision
10	10	10	1	1
25	25	20	0.8	0.84
50	50	46	0.92	0.98
75	75	72	0.96	0.987
100	100	98	0.98	0.99

本地计算-其它

- 地理查询
- 多值列
- **AVG**
  - Sum/Count
- **Order by rand() limit 10**
  - rand(cardinality*limit+2)

大表Join

- TableGroup:

  分区Join

- 附属表（支持M:N）：

  存储：主表内存位置+
  自身内存位置

  加载：主表增加虚拟列

- 附加索引（支持M:N）：

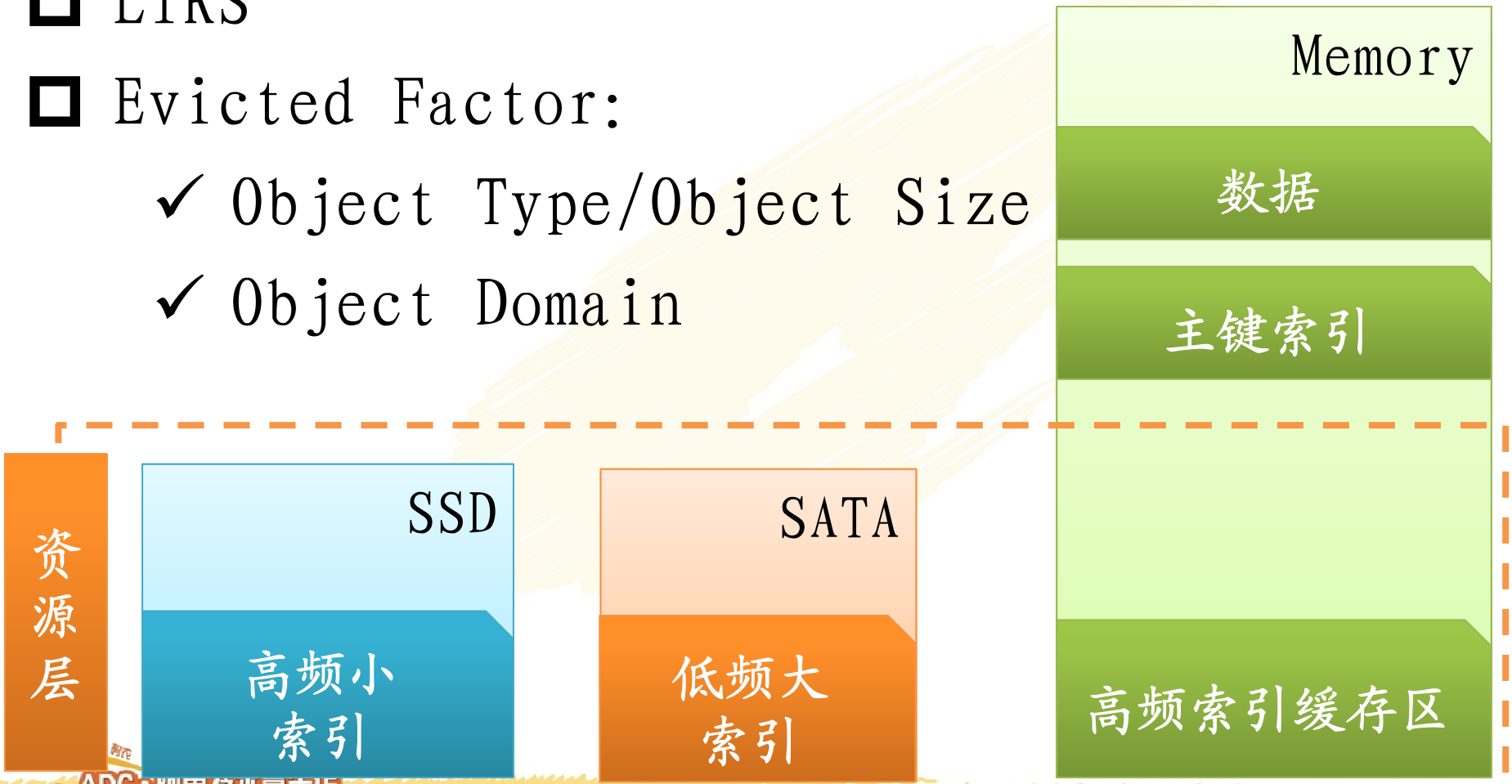
  存储：主表内存位置

  只能用来定位和count

大表Join

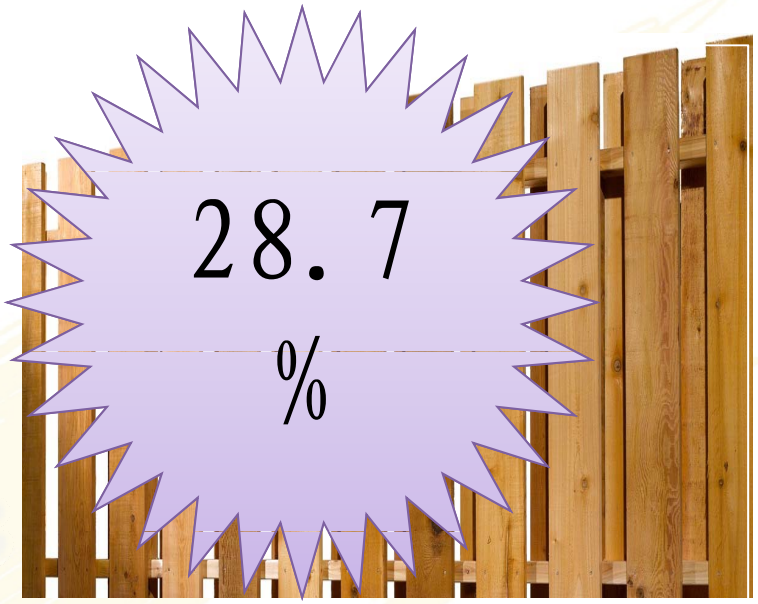
缓存

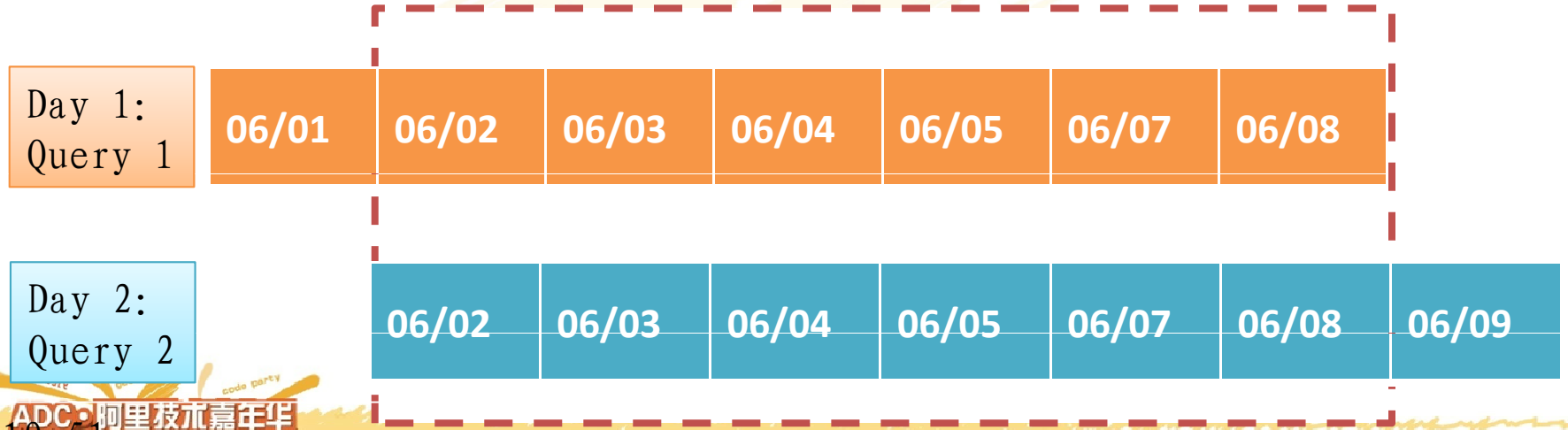
- 本地节点缓存:
  - LIRS
  - Evicted Factor:
    - ✓ Object Type/Object Size
    - ✓ Object Domain



缓存

- Master节点缓存:
  - LIRS
  - SQL cardinality
  - Partition result

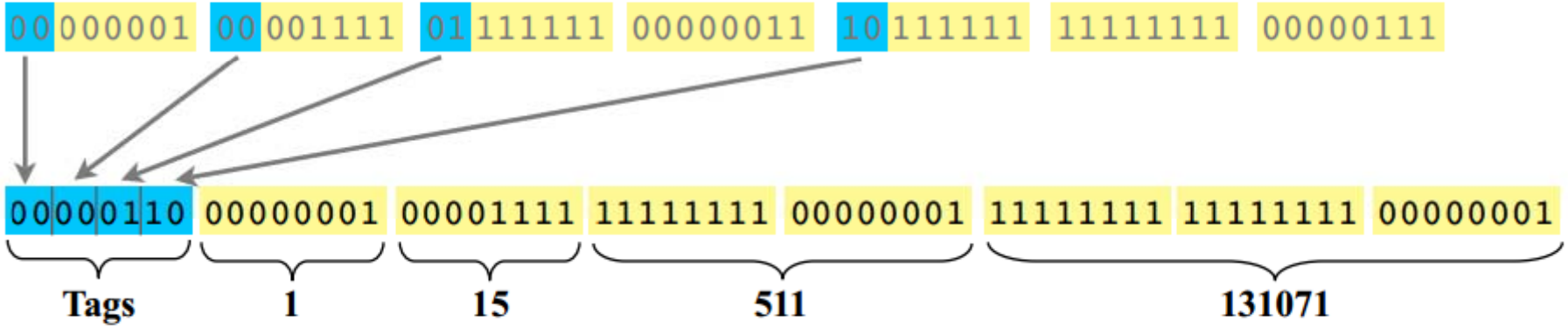




压缩

- 内存压缩
  - Group Varint Encoding (压缩比: 37.5%)
- 磁盘压缩
  - Lz4 (2.08%, 330MB/s, 915MB/s)
- 稀疏列&业务&字符串

- Idea: encode groups of 4 values in 5-17 bytes
  - Pull out 4 2-bit binary lengths into single byte prefix



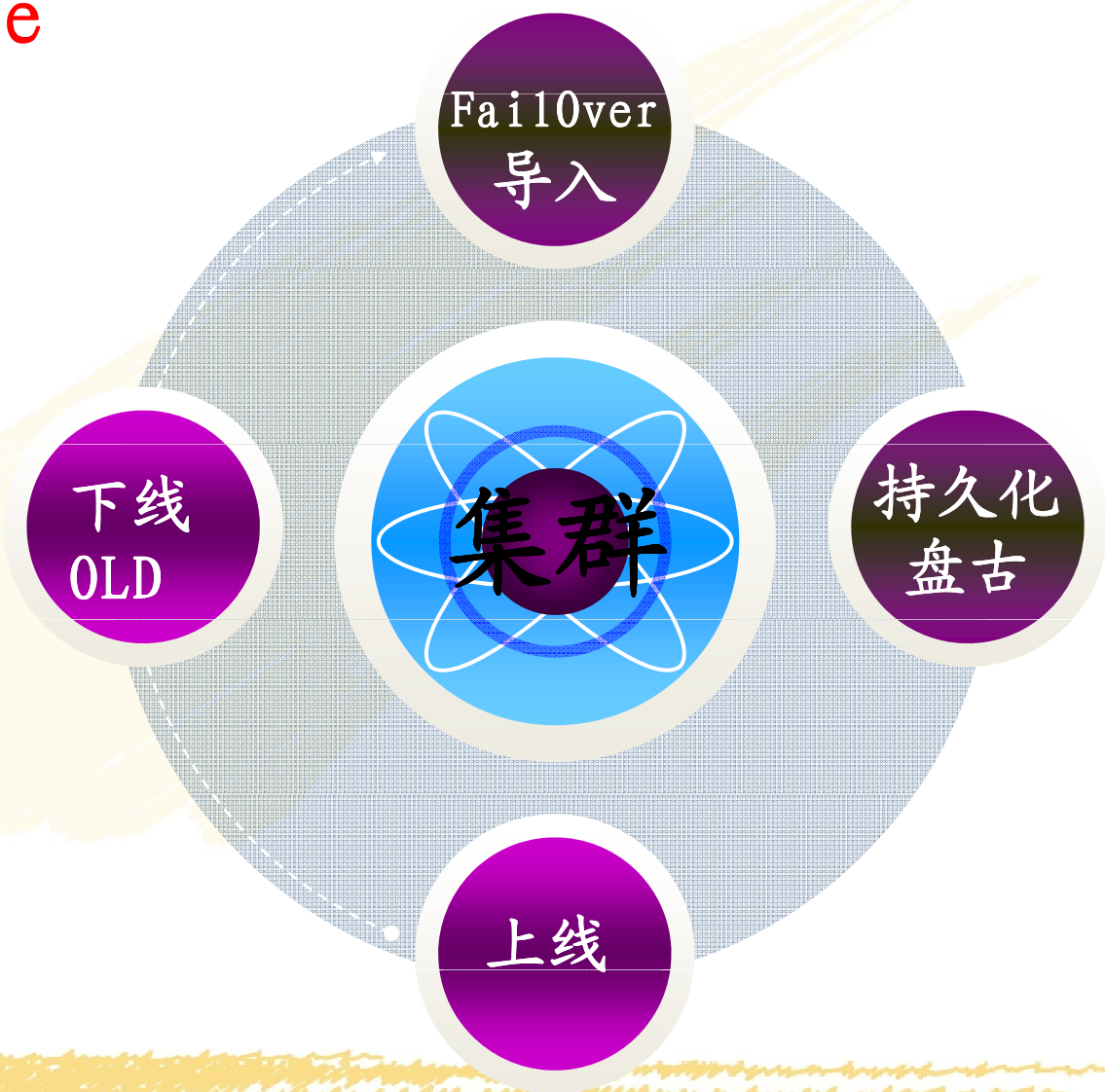
资源管理调度

- 动态规划算法
- Monitor 服务器分布式锁（主/备）
- 参数:
  - 可用内存、可用磁盘（Buffer阈值）
  - 每个表占用的内存、磁盘
  - 最小可用实例数
  - 最小Failover机器数
  - 每个分区最小可用份数（在线上集群）
  - 每个表最多保留分区数（Rotate）
  - 超时设置(上线/下线/导入 超时)
  - 表组信息
  - 节点分组
  - 滚动升级
  - 上线模型（单表上线/整体上线）
  - 资源隔离
  - ….

可用性

- Failover Rotate
- 资源虚拟化(T4)

- Heartbeat
- 双机房
- 任务分布式锁
- 任务持久化
- 任务跟踪JobID
- 执行时间监控



过载保护

- ✓ 全局保护
- ✓ 局部保护（UserID）

案例-指数&魔方

最近30天(2012-11-05 至 2012-12-04) 购买 羊毛衫 的主流人群是 "女性白领，中等消费的初级买家，年龄在25岁-29岁，江苏、浙江和广东共占29%"

人群定位

性别

男 27%　　女 73%

消费层级

低 1%　偏低 8%　中等 57%　偏高 30%　高 3%

买家等级

新手买家 27%　初级买家 48%　中等买家 15%　资深买家 8%　骨灰级买家 3%

身份

白领 53%　　学生 7%

年龄

0　25　30　35　40　50　60

18%　25%　22%　15%　17%　3%　0%

相关品牌　相关商品　**相关属性**　+展开宝贝

名称	销量趋势	热销指数	倾向指数	人群均价
1 通勤 纯色 修身型		208,420	100	￥135
2 通勤 修身型		40,659	24	￥148
3 通勤 其它图案 修身型		31,705	18	￥168
4 甜美 纯色 修身型		30,678	16	￥146
5 纯色 修身型		18,172	12	￥127
6 通勤 条纹 修身型		25,055	11	￥136
7 通勤 纯色 直筒型		34,549	9	￥142

19:51

案例-黄金策

- 黄金策

✓ 用户及关联信息>2,700,000,000

✓ Join表 11 张

✓ 列数>600

✓ AVG RT ¯ 300ms（单表）

✓ Join AVG RT ¯ 2000ms (2大表Join)

✓ QPS ¯ 10 (集群单副本)

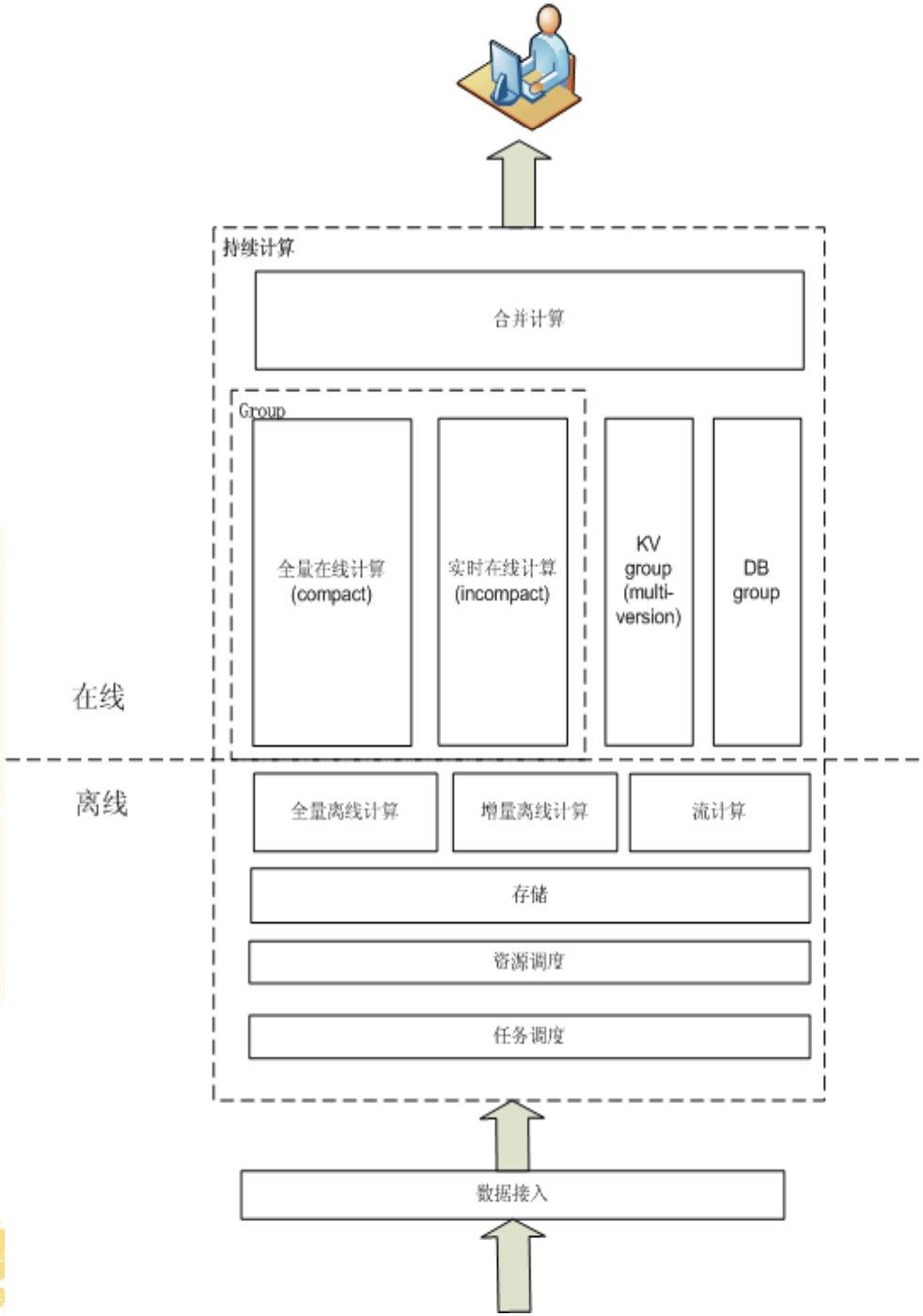
✓ Scan Row/Per Query > 4亿

持续计算

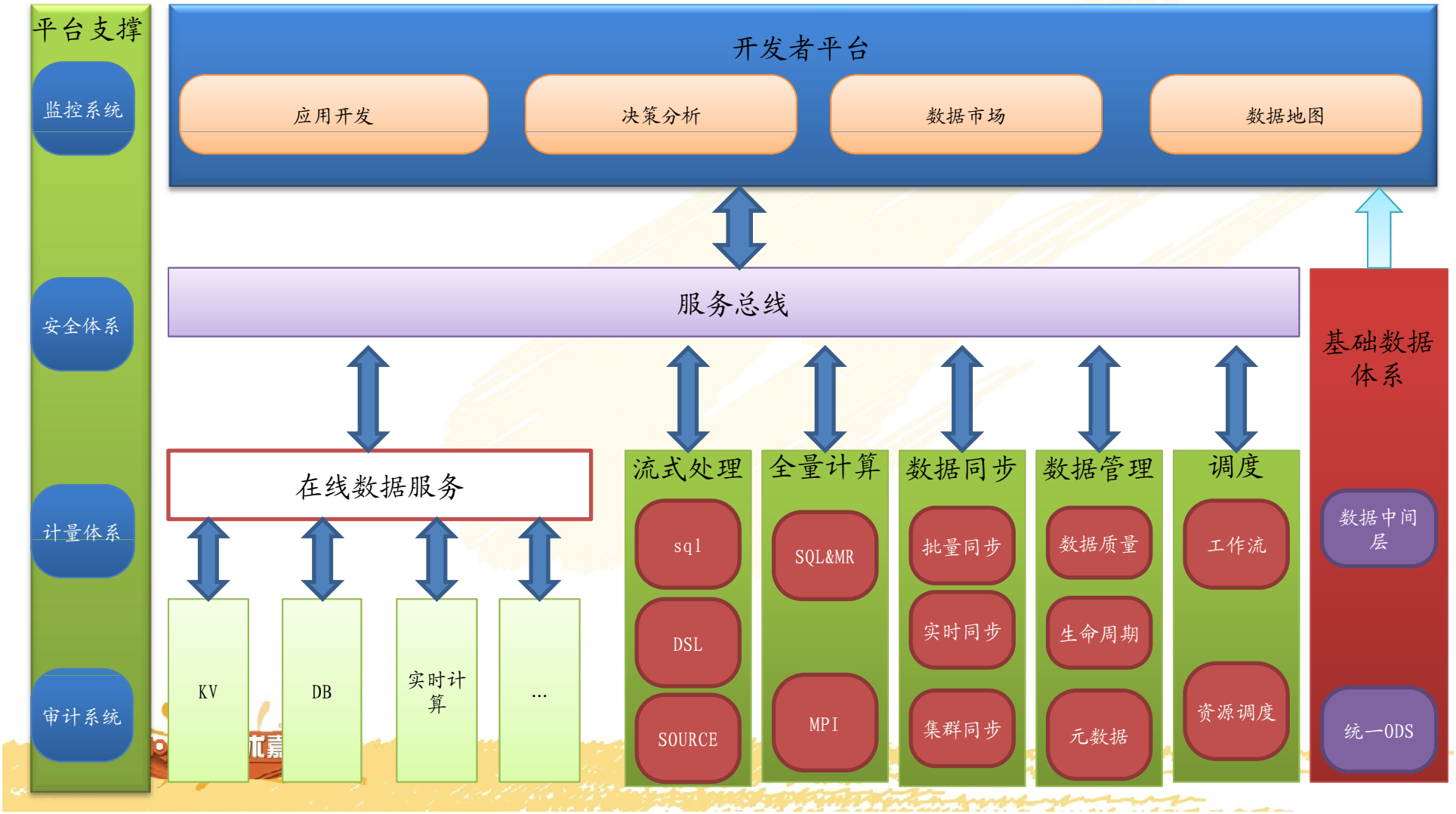
	批量	实时
冲击	Volume	Velocity
资源有利	累积	分摊
业务有利	覆盖	增量
延迟	高	低
成本	高	高
容错	相对简单	复杂
现有资源	多	少
计算	简单	复杂

持续计算

- Continuous Computing：计算可加（增量），大数据集的在线复杂实时计算。实时数据的实时计算
- 统一的离线逻辑
- 统一的在线接口



数据交换平台


平台支撑
监控系统
安全体系
计量体系
审计系统
开发者平台
应用开发
决策分析
数据市场
数据地图
服务总线
基础数据体系
数据中间层
统一ODS
在线数据服务
KV
DB
实时计算
...
流式处理
sql
DSL
SOURCE
全量计算
SQL&MR
MPI
数据同步
批量同步
实时同步
集群同步
数据管理
数据质量
生命周期
元数据
调度
工作流
资源调度

谢　谢