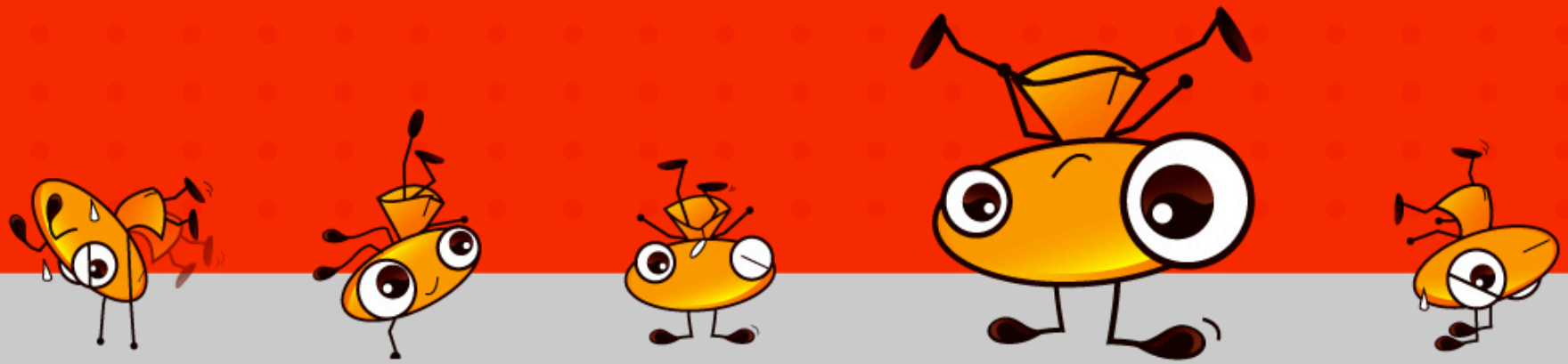
淘宝网
Taobao.com

# 淘宝云梯分布式计算平台整体架构

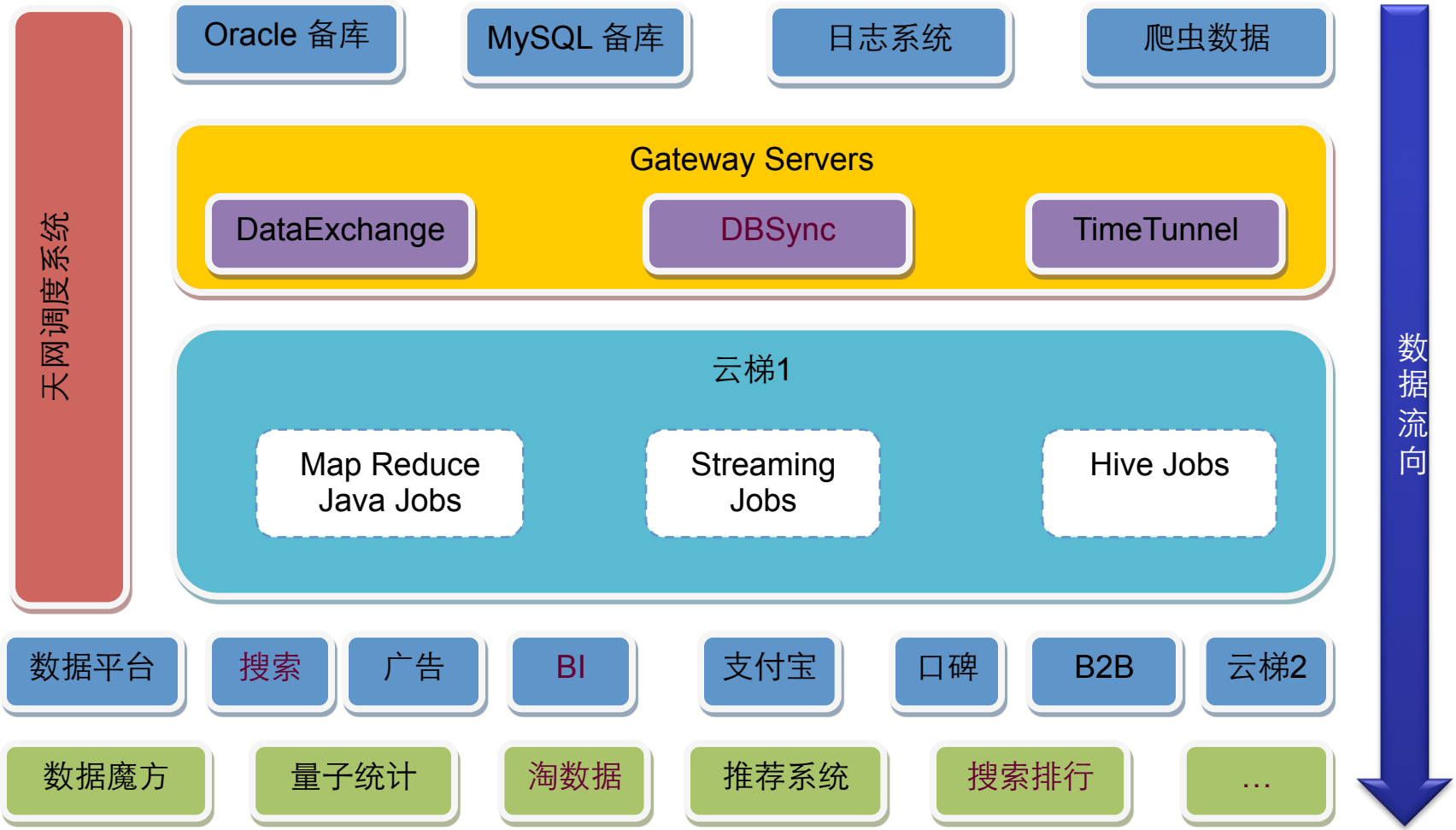
淘宝数据平台与产品部
云铮



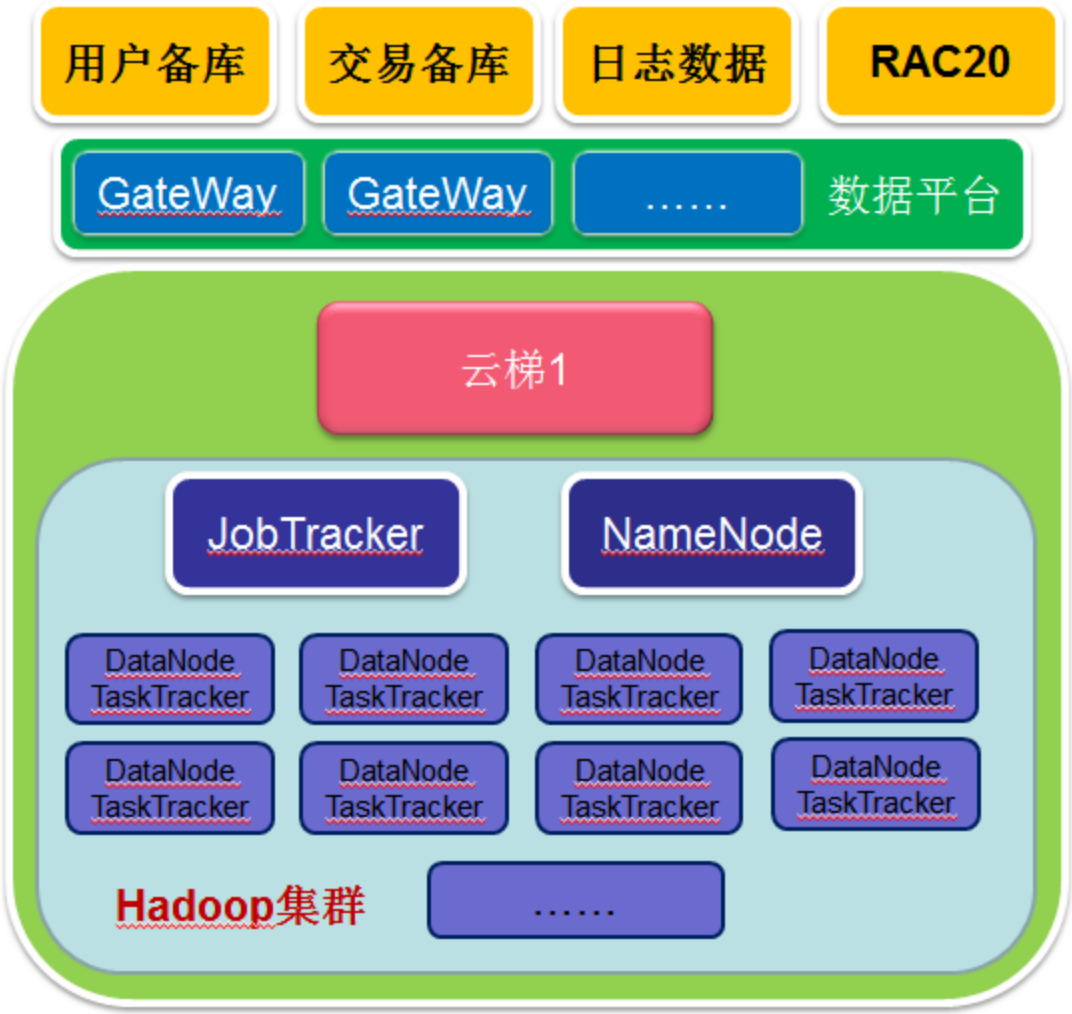
# 目录

# 目录

# 系统整体架构

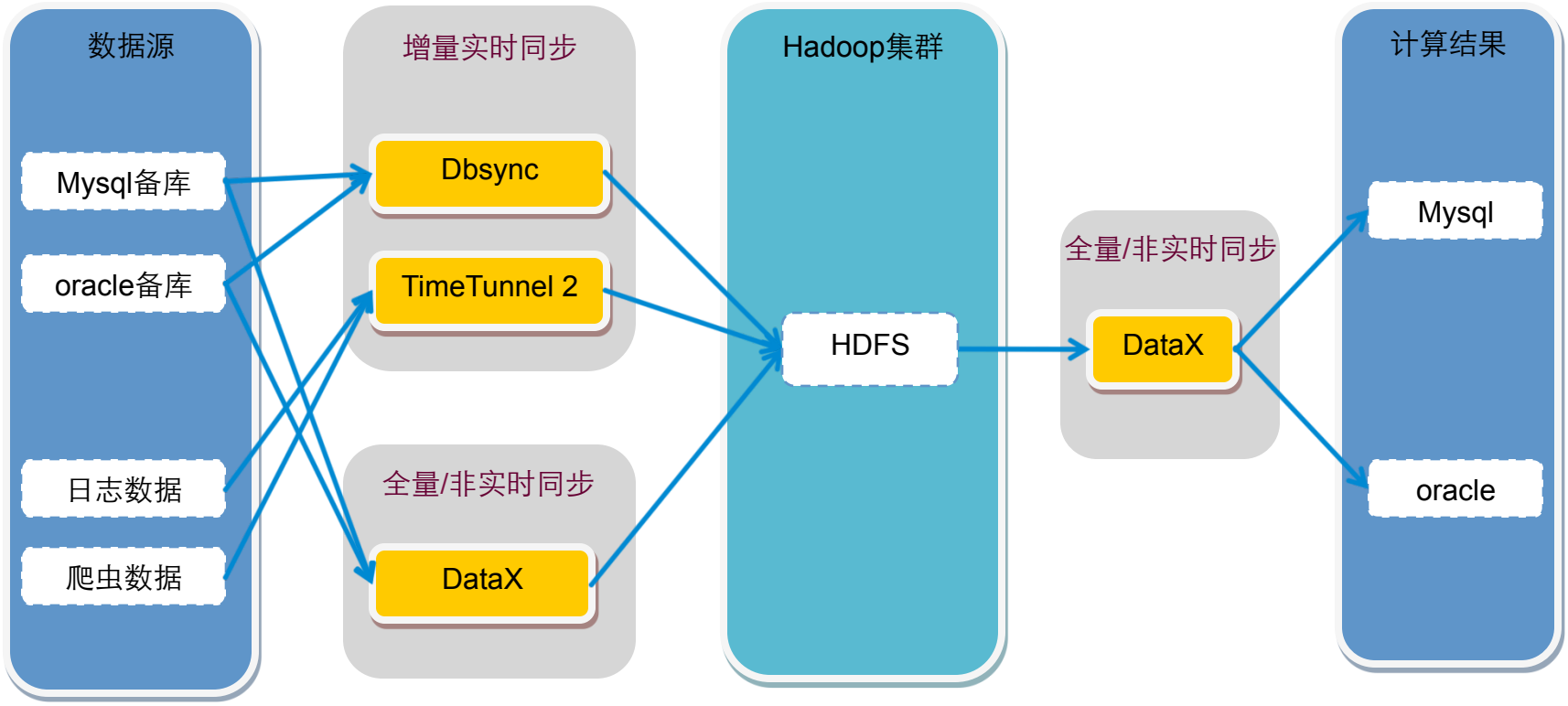

Oracle 备库
MySQL 备库
日志系统
爬虫数据
天网调度系统
Gateway Servers
DataExchange
DBSync
TimeTunnel
云梯1
Map Reduce Java Jobs
Streaming Jobs
Hive Jobs
数据流向
数据平台
搜索
广告
BI
支付宝
口碑
B2B
云梯2
数据魔方
量子统计
淘数据
推荐系统
搜索排行
…

# 淘宝云计算介绍


用户备库
交易备库
日志数据
RAC20
GateWay
GateWay
……
数据平台
云梯1
JobTracker
NameNode
DataNode
TaskTracker
DataNode
TaskTracker
DataNode
TaskTracker
DataNode
TaskTracker
DataNode
TaskTracker
DataNode
TaskTracker
DataNode
TaskTracker
DataNode
TaskTracker
Hadoop集群
……

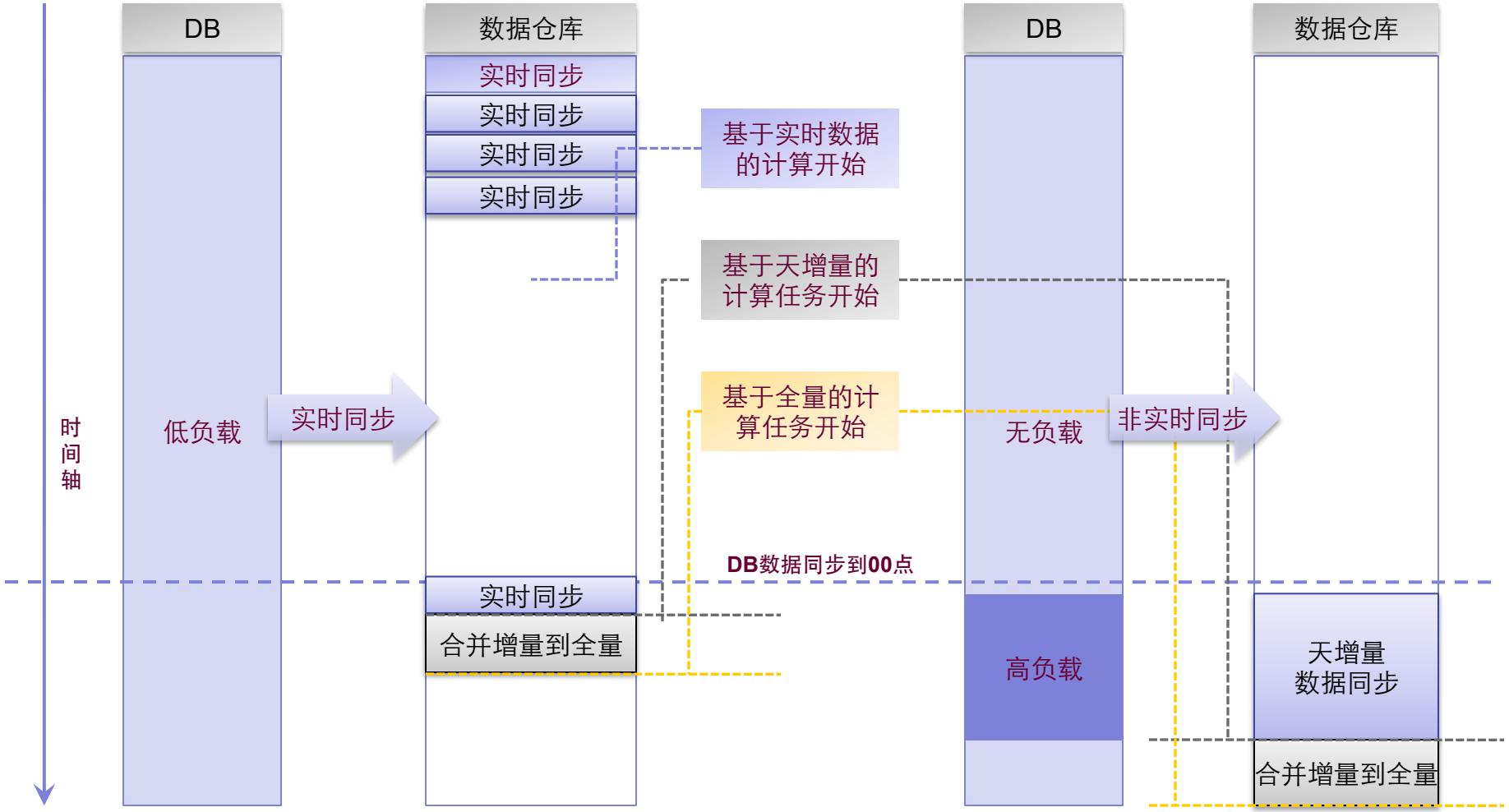
# 目录

# 数据同步方案——概览

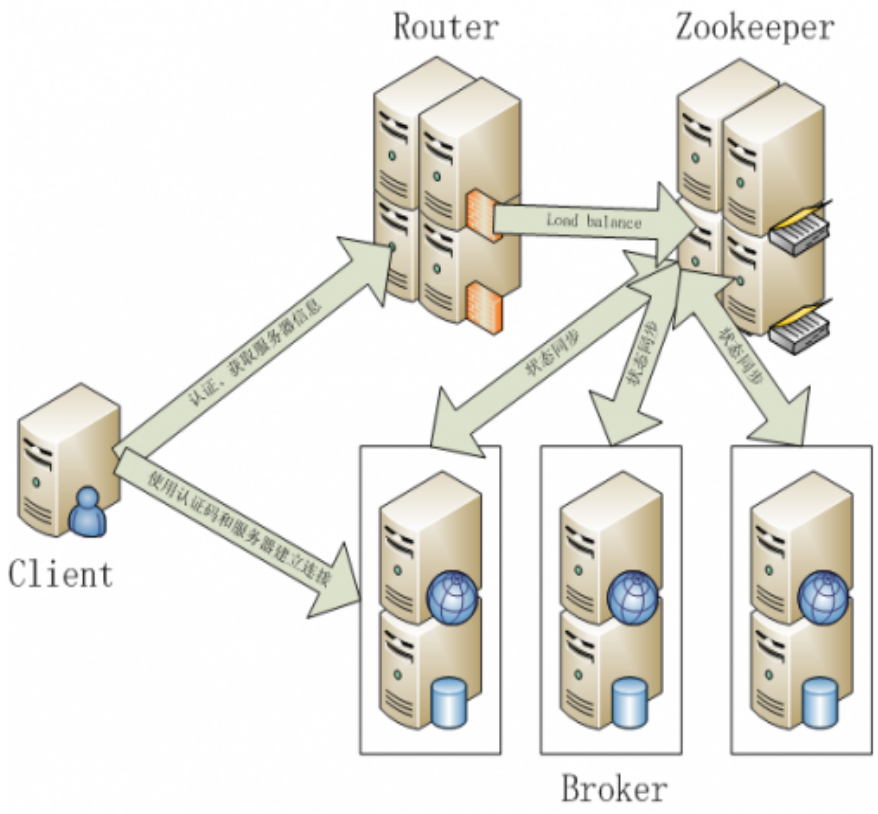



MySQL NGINX TimeTunnel taobao DW basis dataX hadoop

# 数据同步方案—— 实时同步VS非实时同步



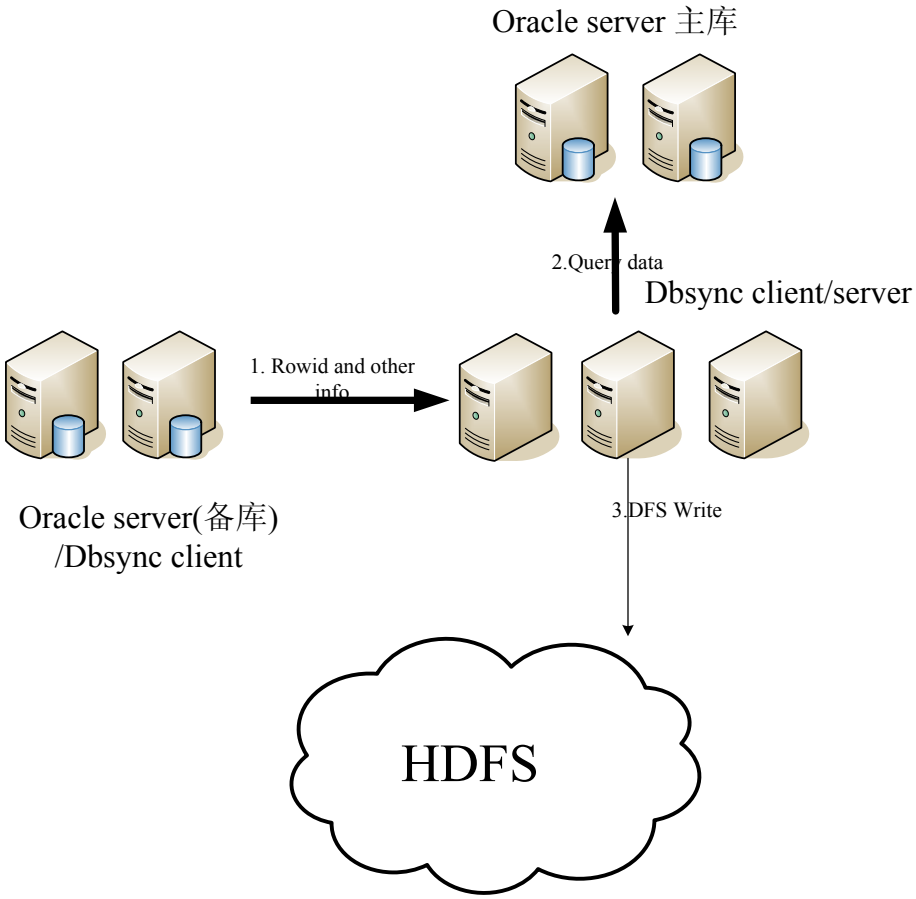
# 数据同步方案—— TimeTunnel2 介绍




TimeTunnel是一个实时数据传输平台，TimeTunnel的主要功能就是实时完成海量数据的交换，因此TimeTunnel的业务逻辑主 要也就有两个：一个是发布数据，将数据发送到TimeTunnel；一个是订阅数据，从TimeTunnel读取自己关心的数据。 TimeTunnel作为一个实时数据传输平台具有以下特点：

**高效性**：单点1k数据可以到4万TPS
**高可靠性**：M-S模式时保证数据不丢失
**高可用性**：单点故障不影响整个集群服务
**顺序性**：当没有故障发生时，保证所有传输都是顺序的，或者说一次连接内的传输是顺序的。

# 数据同步方案—— Dbsync 介绍



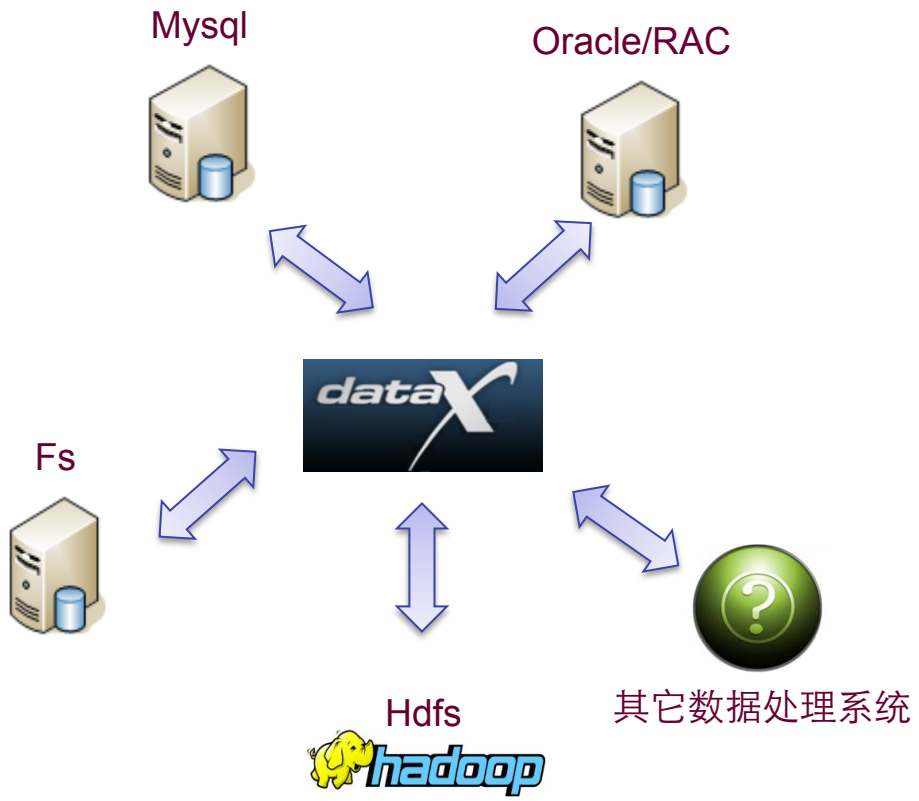

dbsync是一个用于同步服务库数据到HDFS的产品，通过分析数据库服务器的log文件来提取相应的数据库动作，进而达到数据库到HADOOP的数据同步，供相关部门提取增量数据。

Dbsync实时同步性能

| 记录大小 | 速度 |
|---|---|
| 2K | 4M/s |
| 9K | 10M/s |

应用场景
数据量 800G
00:10分备库打开：
非实时同步完成时间0:55
实时同步完成时间0:25

# 数据同步方案—— DataX 介绍




•DataX是一个在异构的数据容器之间交换数据的工具。用于在任意的数据处理系统(RDBMS/NoSql/FS）之间交换数据。

•Framework+plugin，Framework处理了高速数据交换的大部分问题，插件提供对数据处理系统的访问。

•运行模式： stand-alone / on hadoop

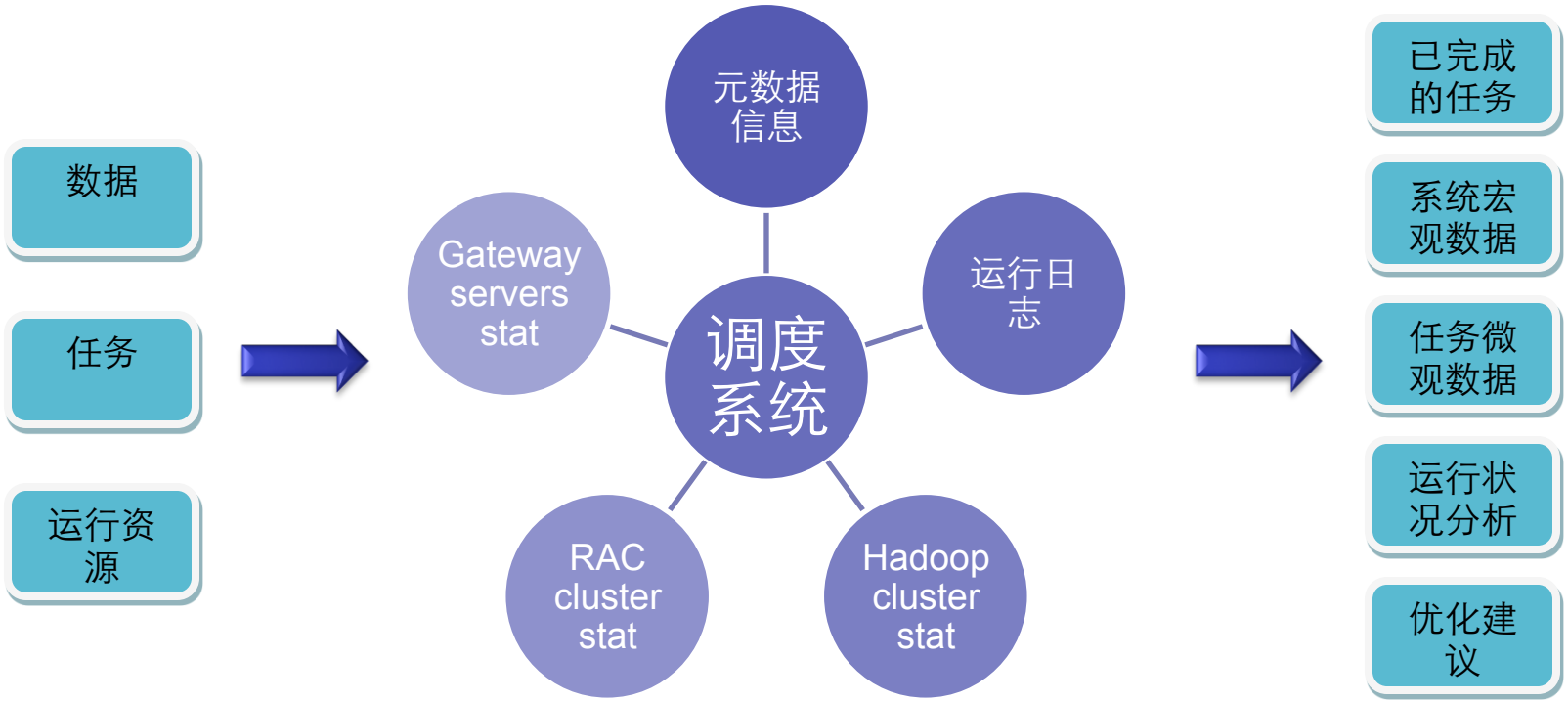
•Webui + cui 基于元数据的高效配置,例子:
表A sharding为32个库，1024张表，配置时间<1 min

DataX部分性能数据:

| 数据输出方向 | Speed（M/s） |
|---|---|
| mysqll->hdfs | 29.9 |
| oracleloader->hdfs | 31.9 |
| hdfs->oracle | 18.1 |

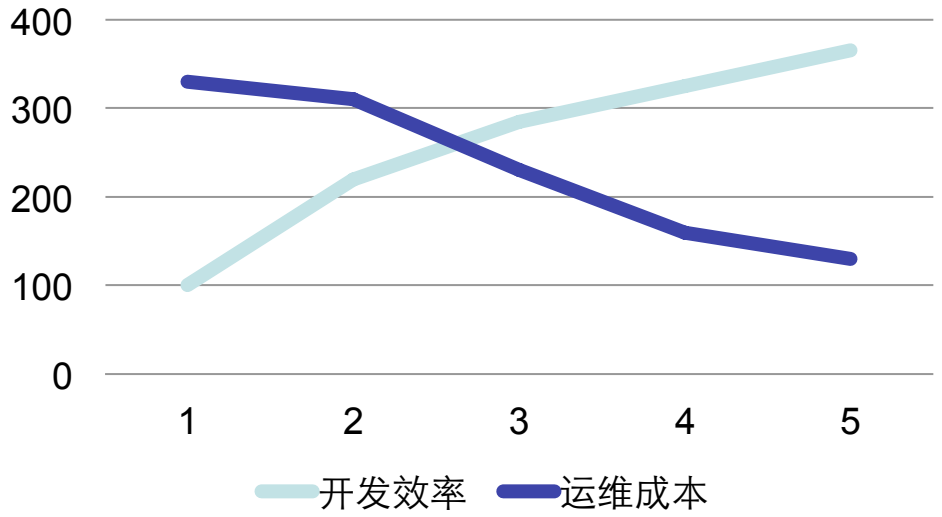
# 目录

# 调度系统


数据
任务
运行资源
元数据信息
Gateway servers stat
运行日志
调度系统
RAC cluster stat
Hadoop cluster stat
已完成的任务
系统宏观数据
任务微观数据
运行状况分析
优化建议

# 调度系统——生产率银弹

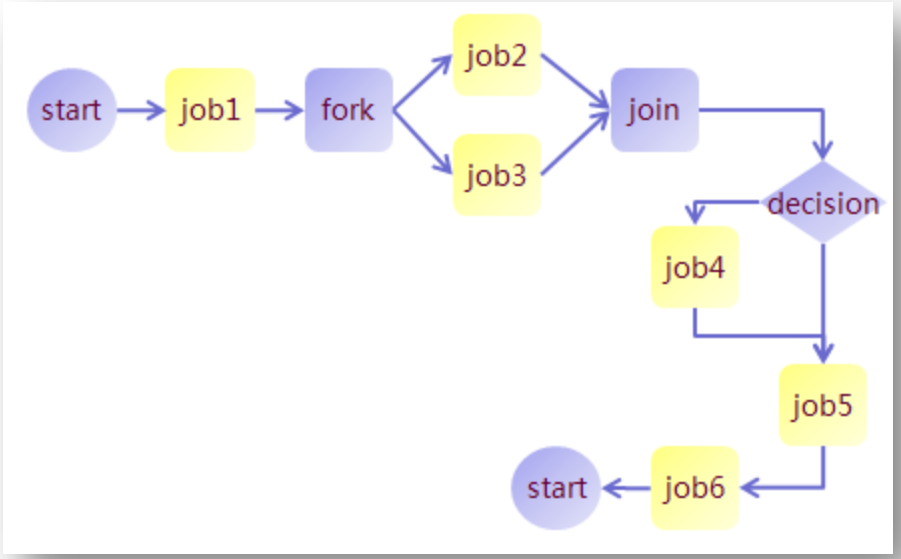
- 自动部署
- 智能调度
- 运维平台
- 监控告警
- Web UI
- 异构平台支持



# 调度系统——模块/子系统


Workflow/Data Pipelines 管理
• Flow Control
• Time trigger
• Data trigger
业务运维
• 智能重跑
• 补历史数据
• 查看日志/源代码
• 日志/代码搜索
• 业务分组
智能调度
• 异构系统支持
• 优先级设置
• Dynamic LB/ Failover
• 自动重跑
• 资源分组
监控/告警/分析
• 运行信息收集
• 系统瓶颈分析
• 关键路径分析

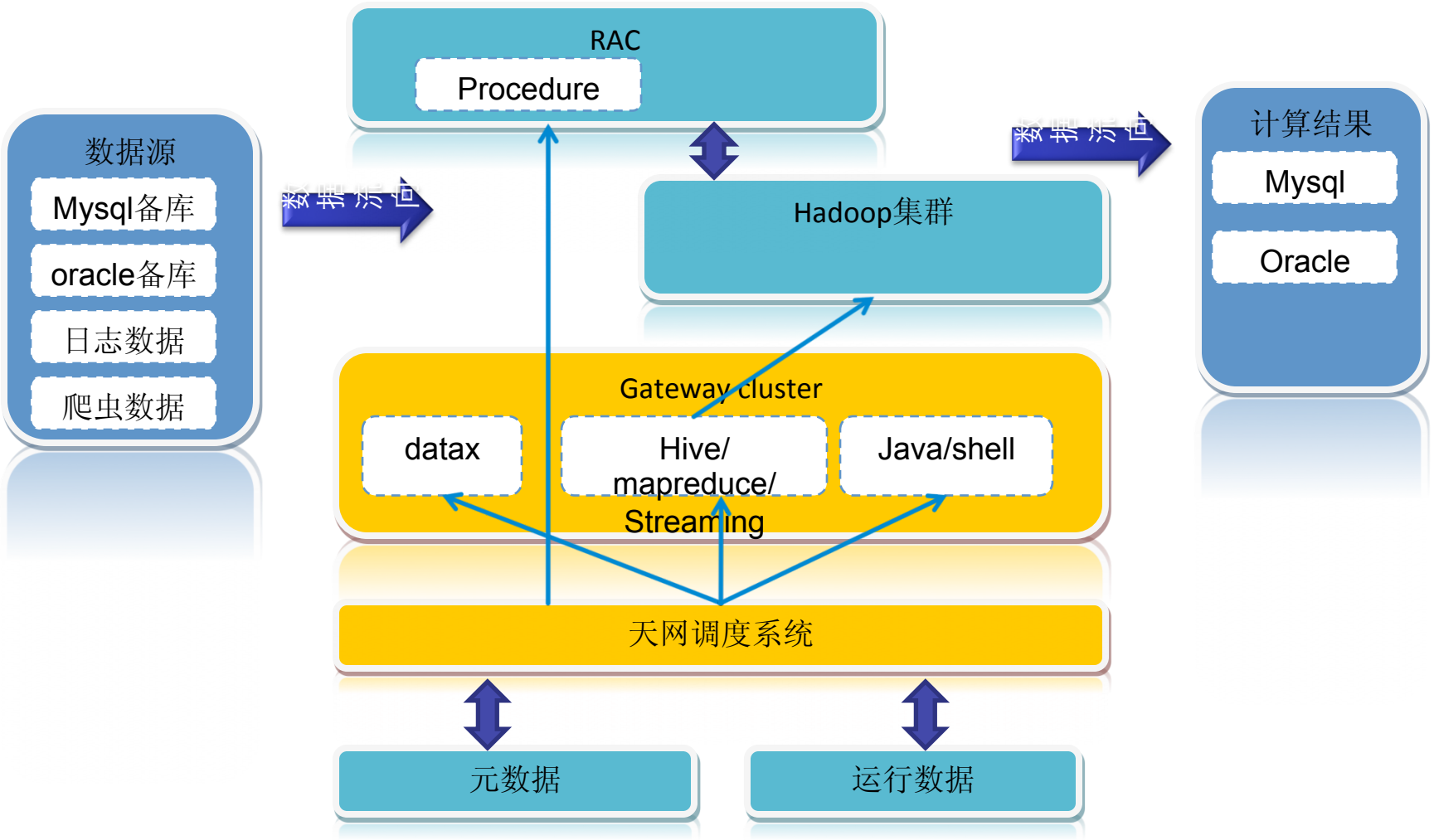
# 调度系统——任务触发方式




Flow control/Data Trigger

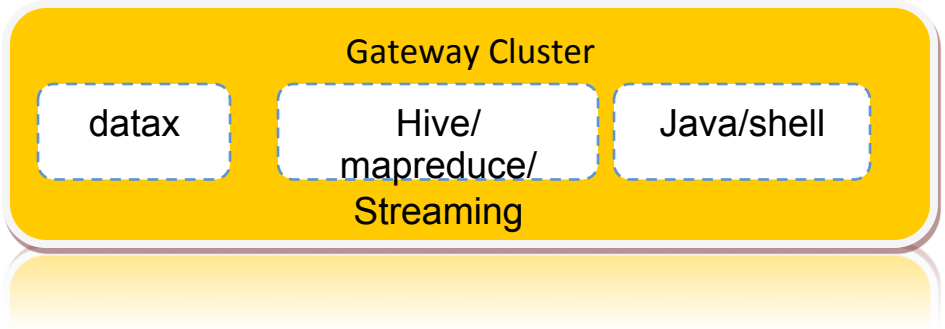
Time Trigger

# 调度系统——调度方式


RAC
Procedure
数据源
Mysql备库
oracle备库
日志数据
爬虫数据
Hadoop集群
计算结果
Mysql
Oracle
Gateway cluster
datax
Hive/
mapreduce/
Streaming
Java/shell
天网调度系统
元数据
运行数据

# 调度系统——什么是Gateway?

- Gateway:参与天网调度的资源
- 功能:
  - 数据同步(dataX, DBSync,TimeTunnel2…)
  - 数据上传/下载(hadoop fs –put/get/getmerge)
  - 日志收集
  - Hive sql语句提交运行
  - MapReduce程序提交运行
  - 集群间数据同步(hadoop distcp)



# 调度系统—— Gateway规模及规划

- 用于生产的Gateway约30台，由天网调度统一进行任务分发，并行控制。
- 数据同步(dataX, DBSync,TimeTunnel2…)
- 数据上传/下载(hadoop fs –put/get/getmerge)
- 日志收集
- Hive sql语句提交运行
- MapReduce程序提交运行
- 集群间数据同步(hadoop distcp)

# 调度系统——gateway standardization

天网调度系统

智能调度

- fail over
- dynamic LB

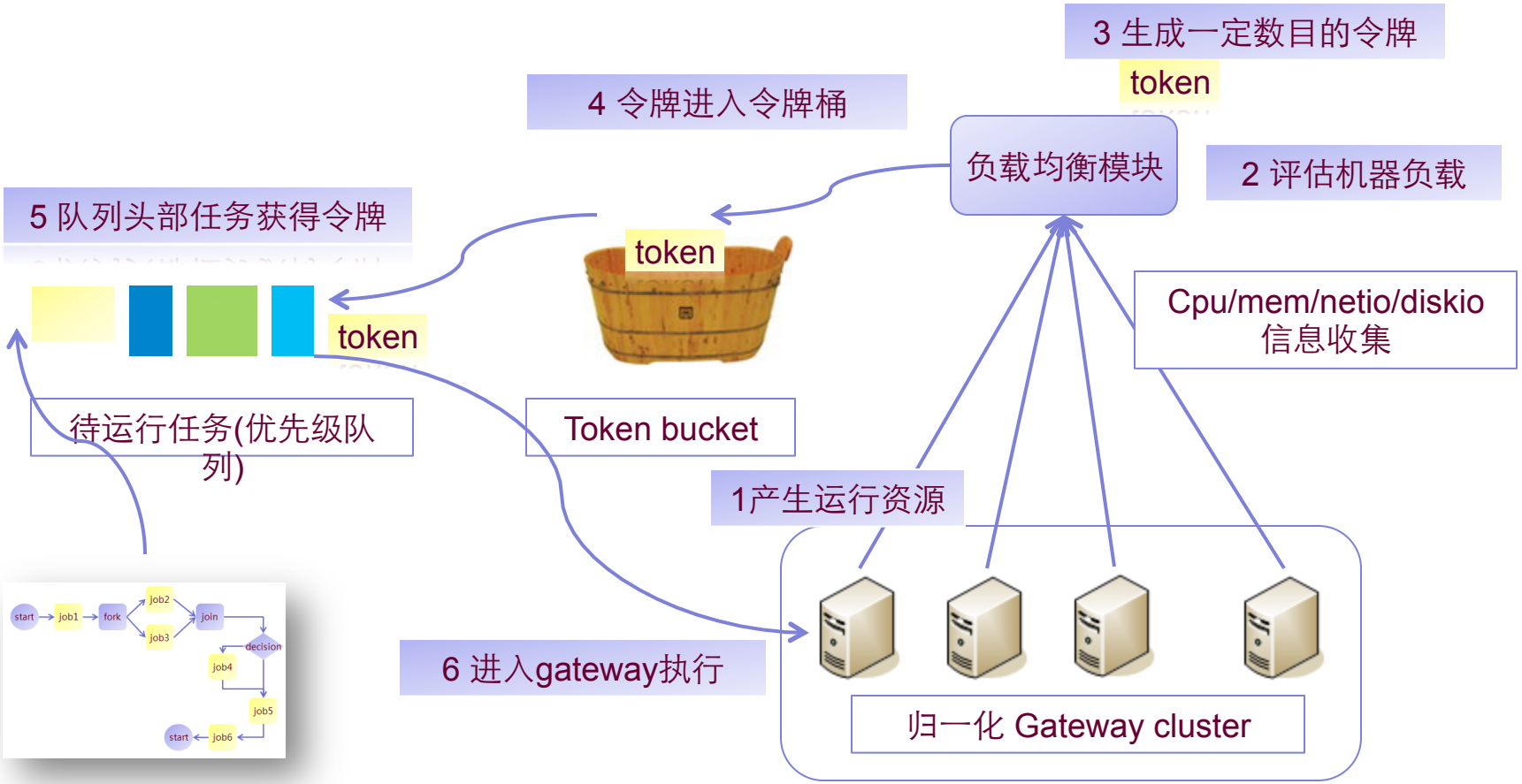
Standardized Gateway server cluster

- 标准化配置
- 统一包管理
- 批量管理工具
- 标准化运维

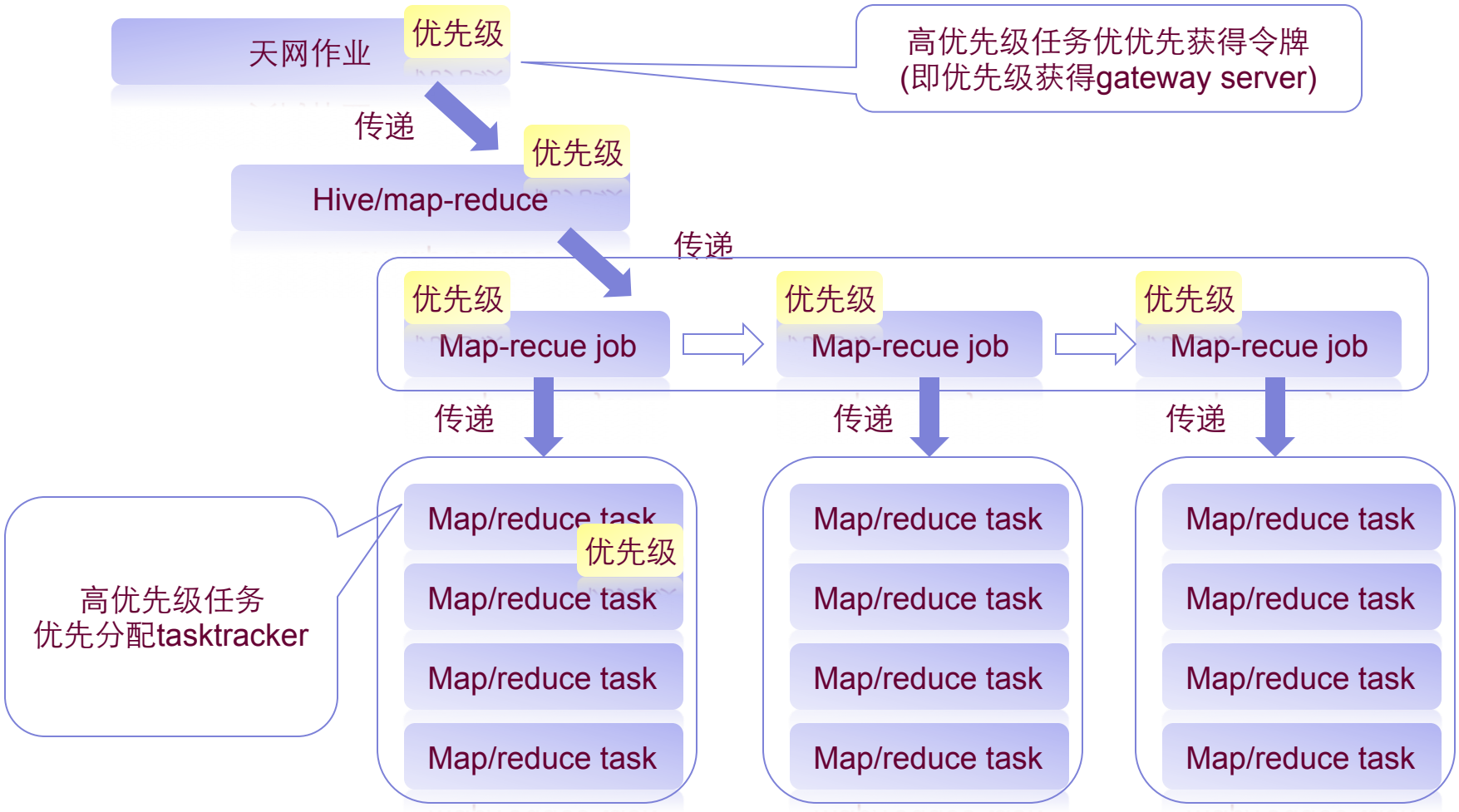
benefit

- high-available
- high-usage
- Lower maintenance cost per machine

# 调度系统——Dynamic LB实现



# 调度系统——优先级策略(实现)



# 调度系统——优先级策略(意义)

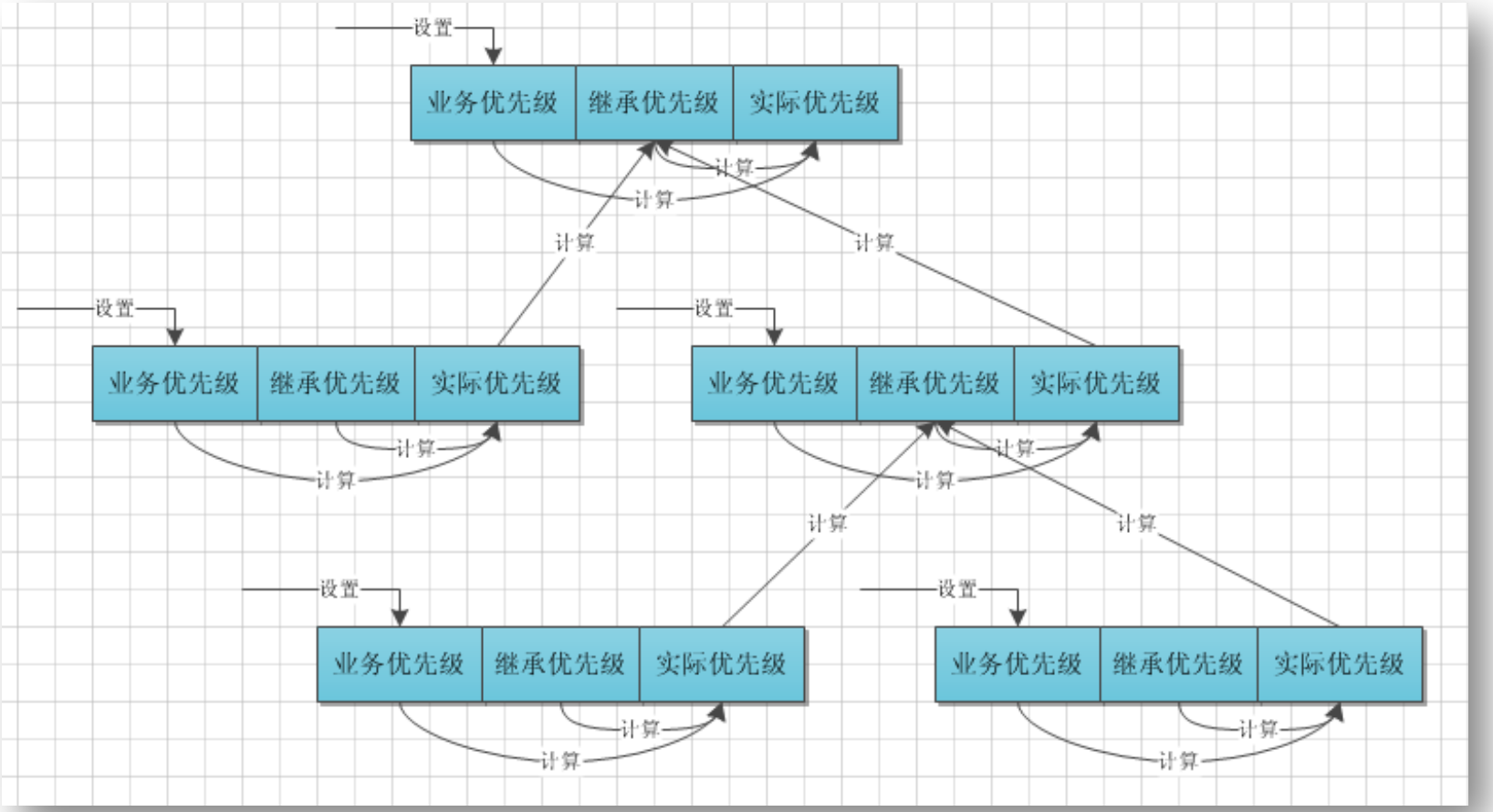
## 没有优先级

| 客户重要程度 | ETL进度 | 客户反应 |
| --- | --- | --- |
| 高 | ETL jobs |  |
| | ETL jobs |  |
| | ETL jobs |  |
| | ETL jobs |  |
| 低 | ETL jobs |  |

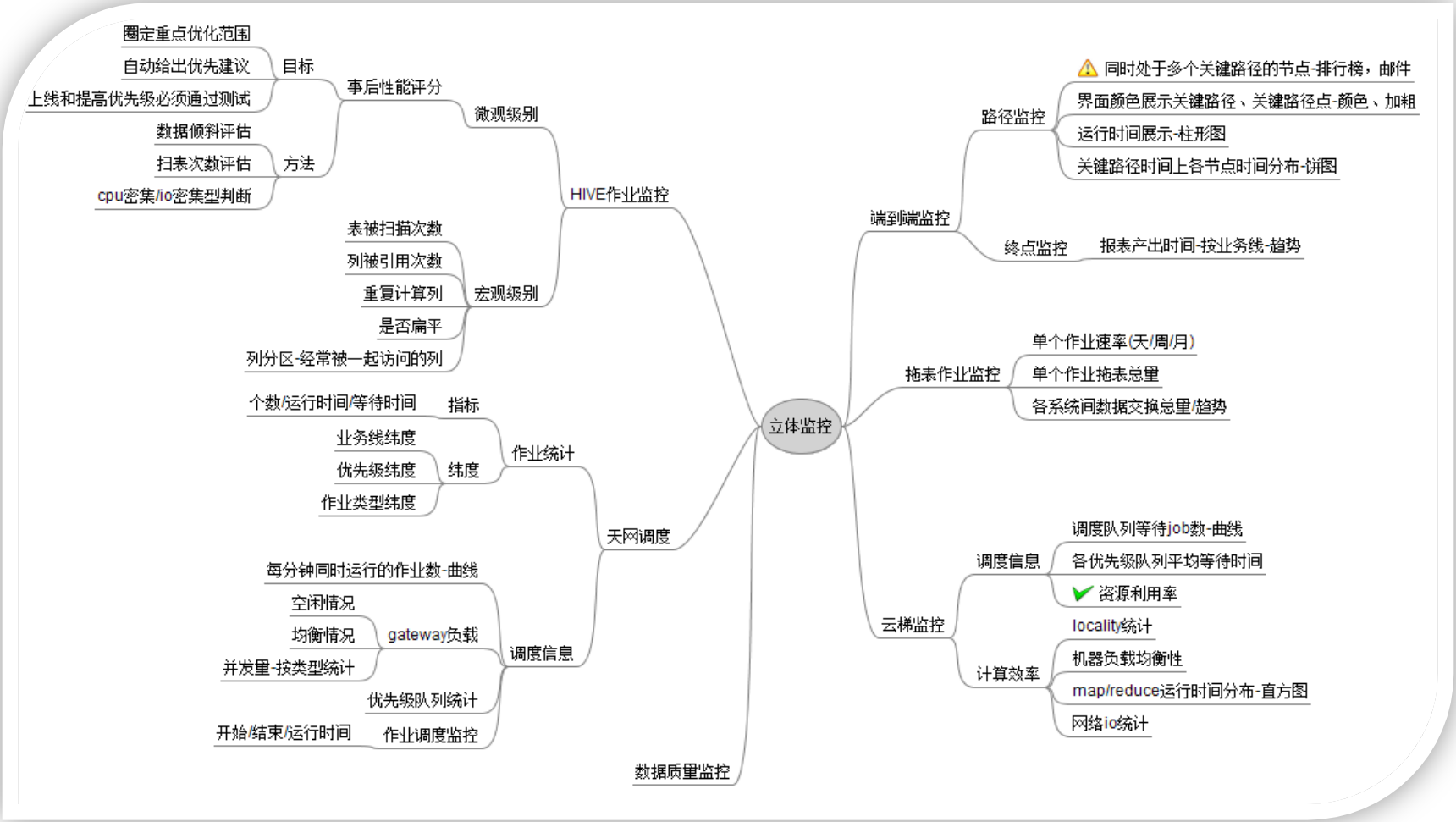
## 有优先级

| 客户重要程度 | ETL进度 | 客户反应 |
| --- | --- | --- |
| 高 | ETL jobs |  |
| | ETL jobs |  |
| | ETL jobs |  |
| | ETL jobs |  |
| 低 | ETL jobs |  |

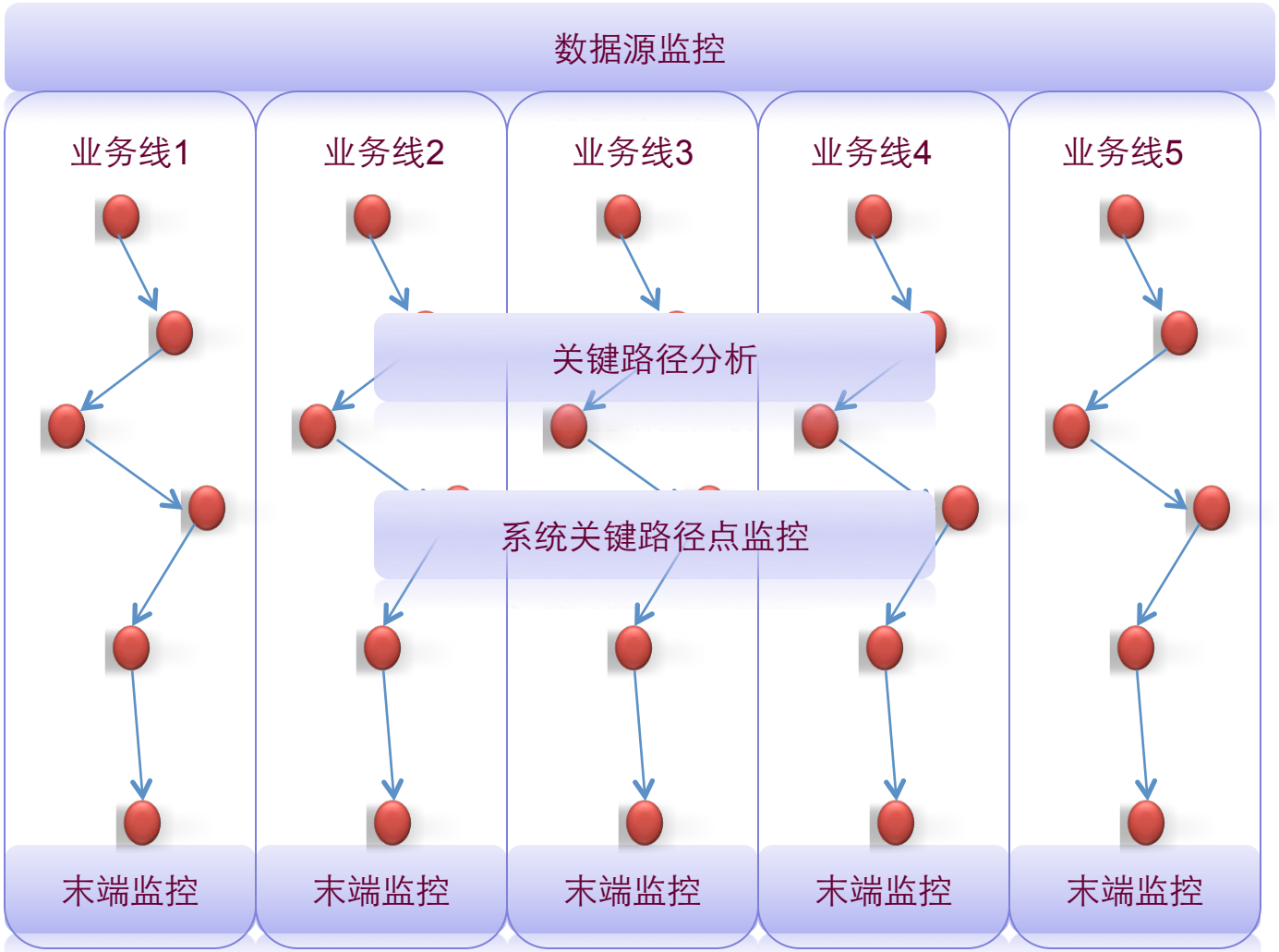
# 调度系统——优先级策略(DAG继承算法)



# 调度系统——监控全景


立体监控
HIVE作业监控
微观级别
事后性能评分
目标
圈定重点优化范围
自动给出优先建议
上线和提高优先级必须通过测试
方法
数据倾斜评估
扫表次数评估
cpu密集/io密集型判断
宏观级别
表被扫描次数
列被引用次数
重复计算列
是否扁平
列分区-经常被一起访问的列
天网调度
作业统计
指标
个数/运行时间/等待时间
纬度
业务线纬度
优先级纬度
作业类型纬度
调度信息
每分钟同时运行的作业数-曲线
gateway负载
空闲情况
均衡情况
并发量-按类型统计
优先级队列统计
作业调度监控
开始/结束/运行时间
数据质量监控
端到端监控
路径监控
同时处于多个关键路径的节点-排行榜，邮件
界面颜色展示关键路径、关键路径点-颜色、加粗
运行时间展示-柱形图
关键路径时间上各节点时间分布-饼图
终点监控
报表产出时间-按业务线-趋势
拖表作业监控
单个作业速率(天/周/月)
单个作业拖表总量
各系统间数据交换总量/趋势
云梯监控
调度信息
调度队列等待job数-曲线
各优先级队列平均等待时间
资源利用率
计算效率
locality统计
机器负载均衡性
map/reduce运行时间分布-直方图
网络io统计

# 监控重点——系统边界和主干

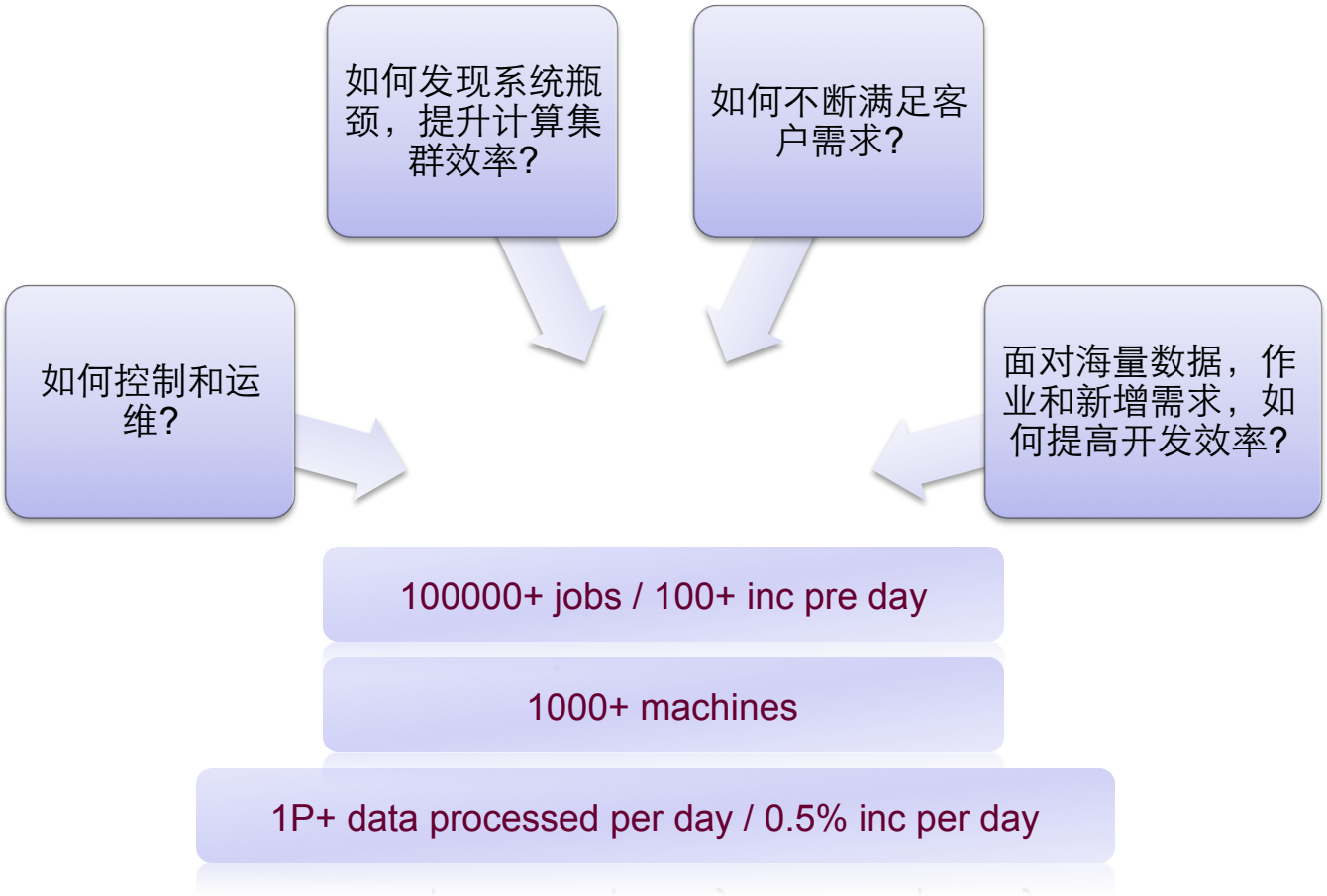


淘宝网 Taobao.com

淘我喜欢!

# 目录

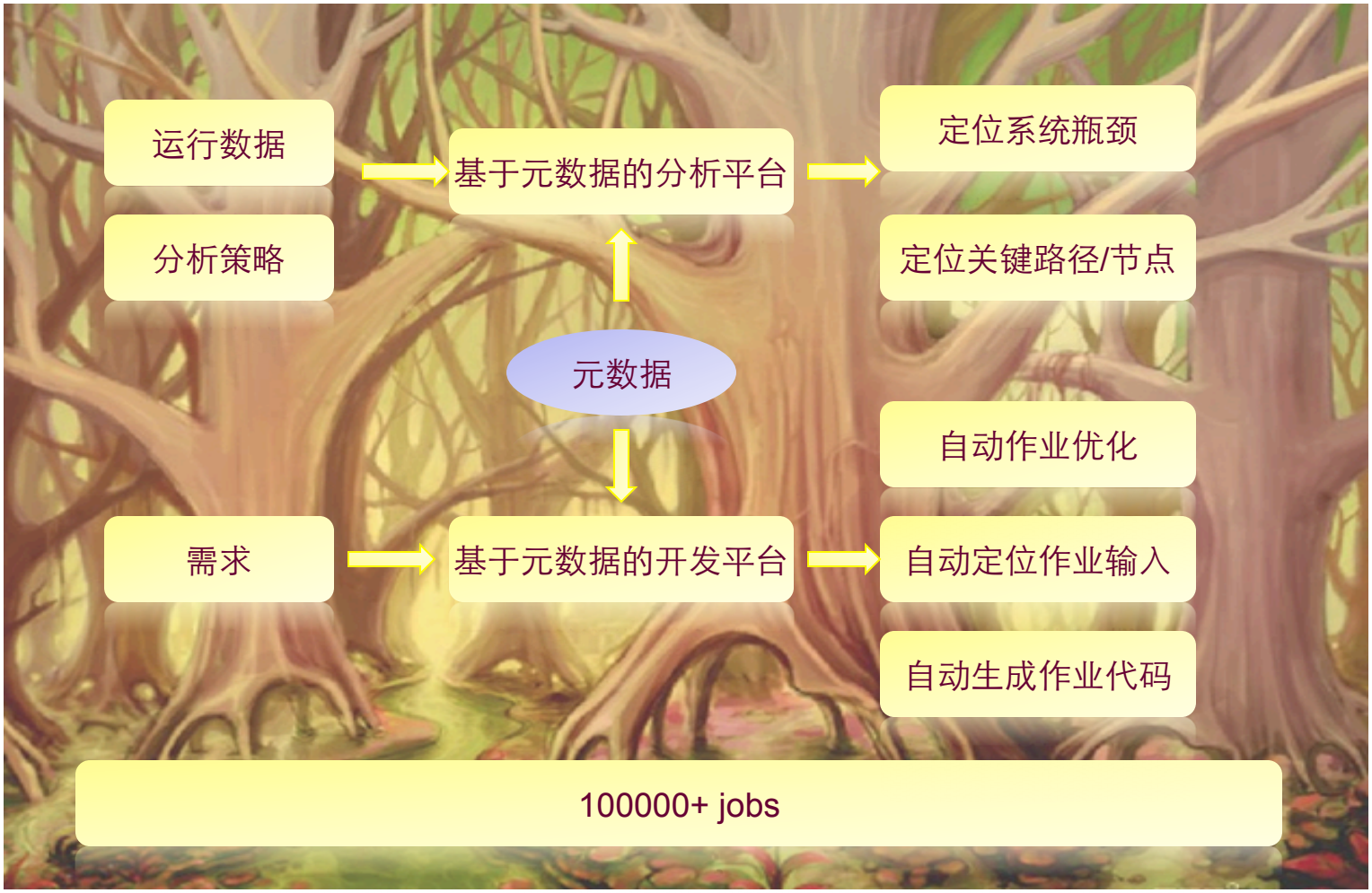
# 问题


如何发现系统瓶颈，提升计算集群效率?
如何不断满足客户需求?
如何控制和运维?
面对海量数据，作业和新增需求，如何提高开发效率?
100000+ jobs / 100+ inc pre day
1000+ machines
1P+ data processed per day / 0.5% inc per day

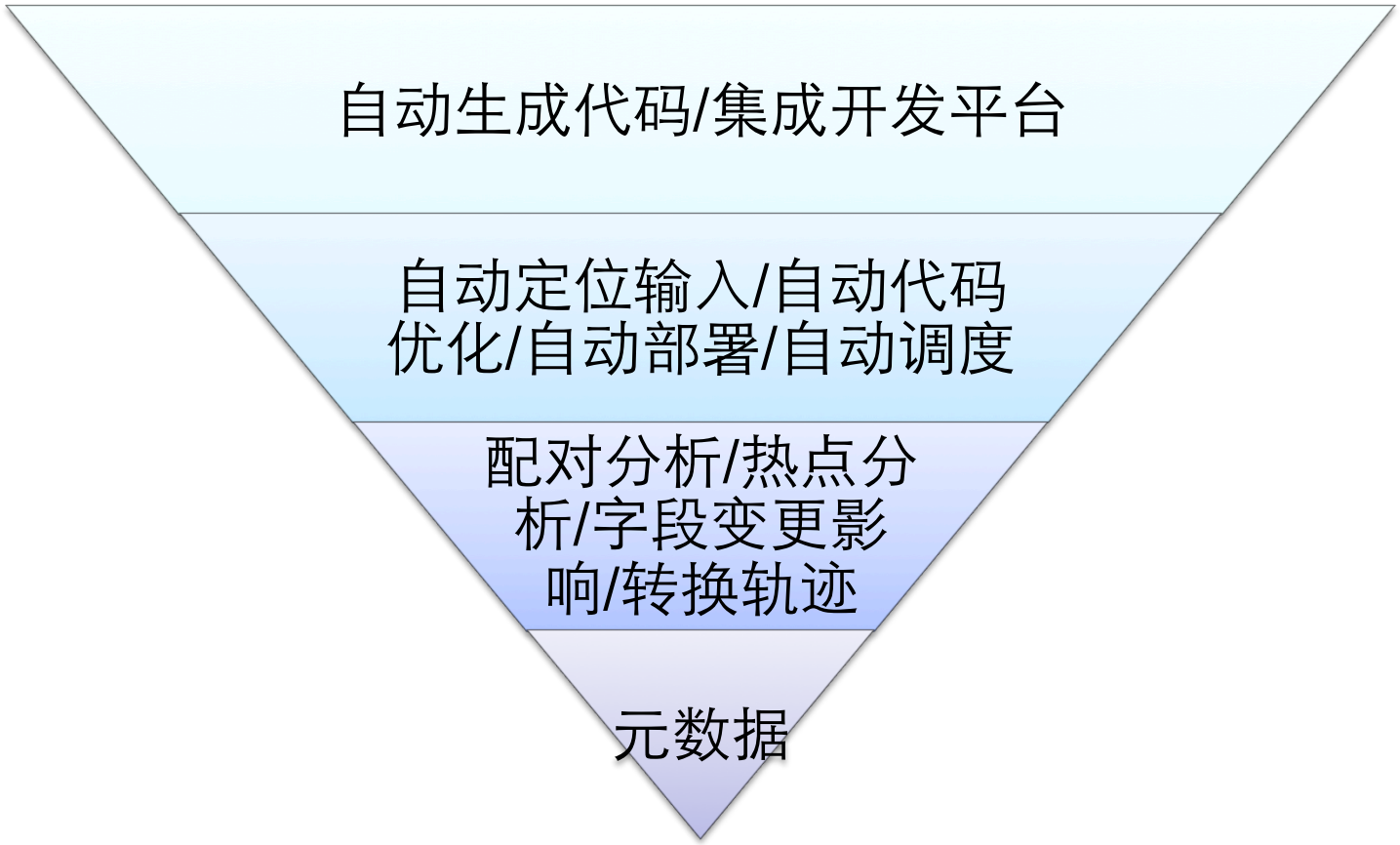
# 问题

面对上面的问题
靠经验丰富的架构师?
还是靠智能的分析系统?

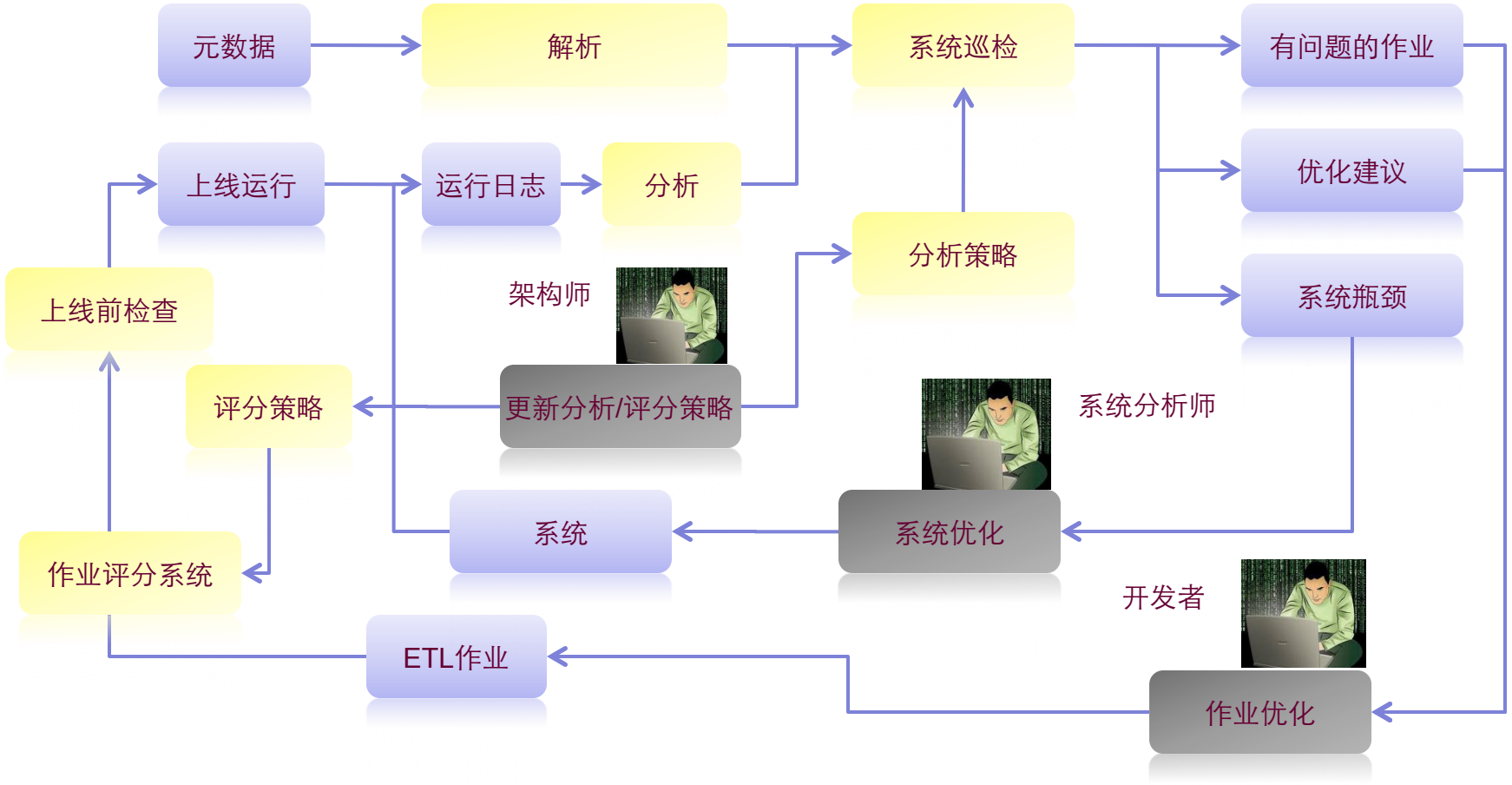
# 挖掘元数据金矿


运行数据
分析策略
基于元数据的分析平台
定位系统瓶颈
定位关键路径/节点
元数据
自动作业优化
需求
基于元数据的开发平台
自动定位作业输入
自动生成作业代码
100000+ jobs

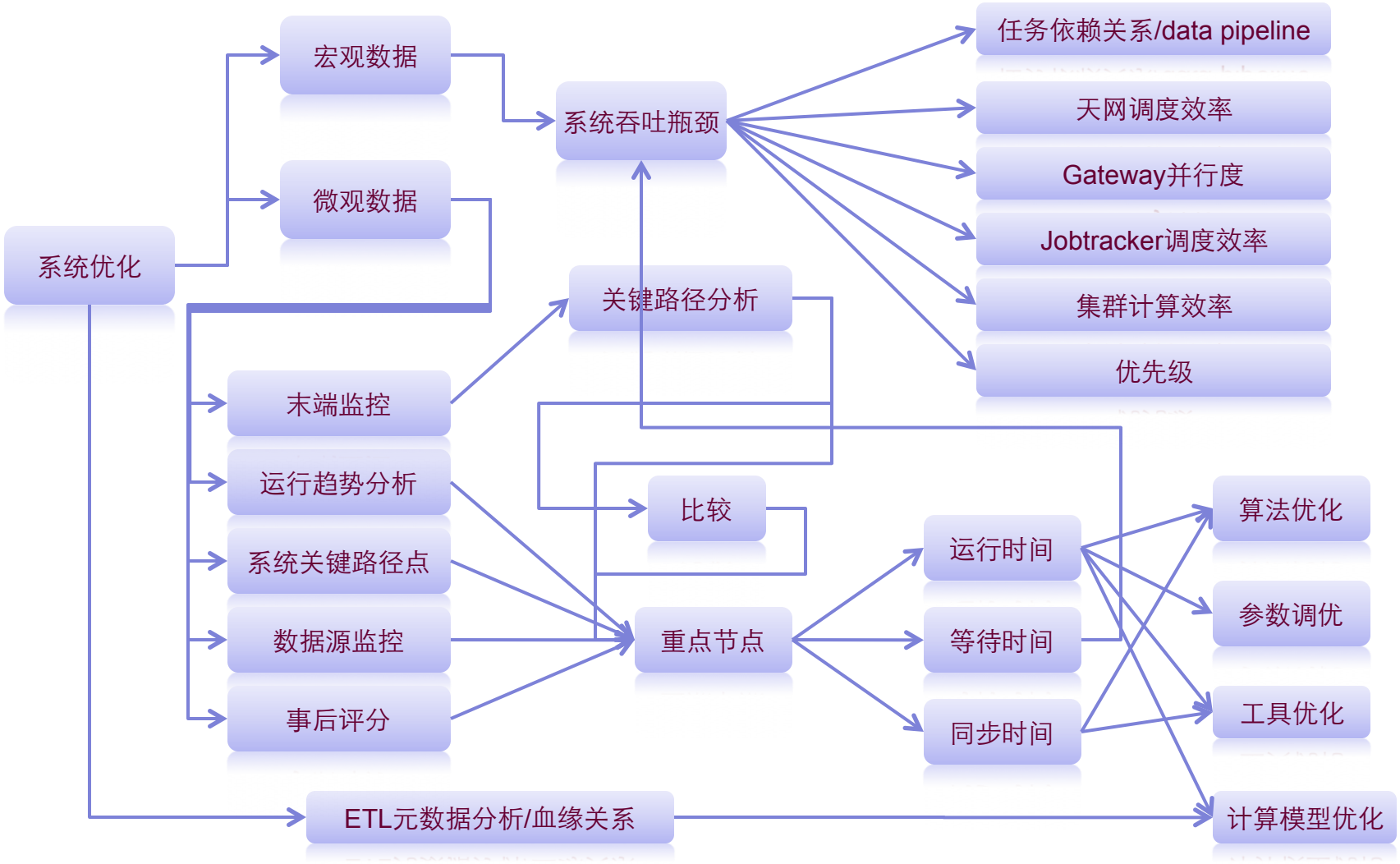
# 基于元数据的开发平台


自动生成代码/集成开发平台
自动定位输入/自动代码优化/自动部署/自动调度
配对分析/热点分析/字段变更影响/转换轨迹
元数据

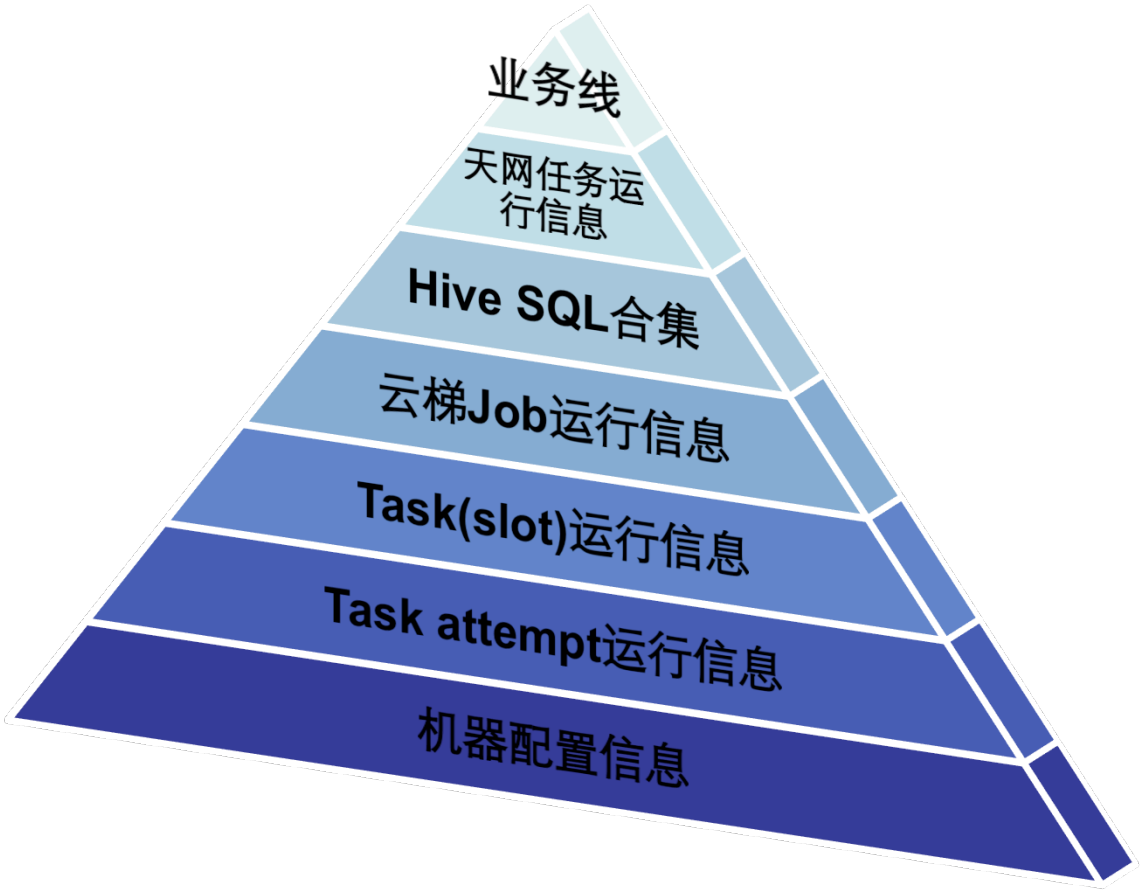
# 基于元数据的分析平台——运行分析系统



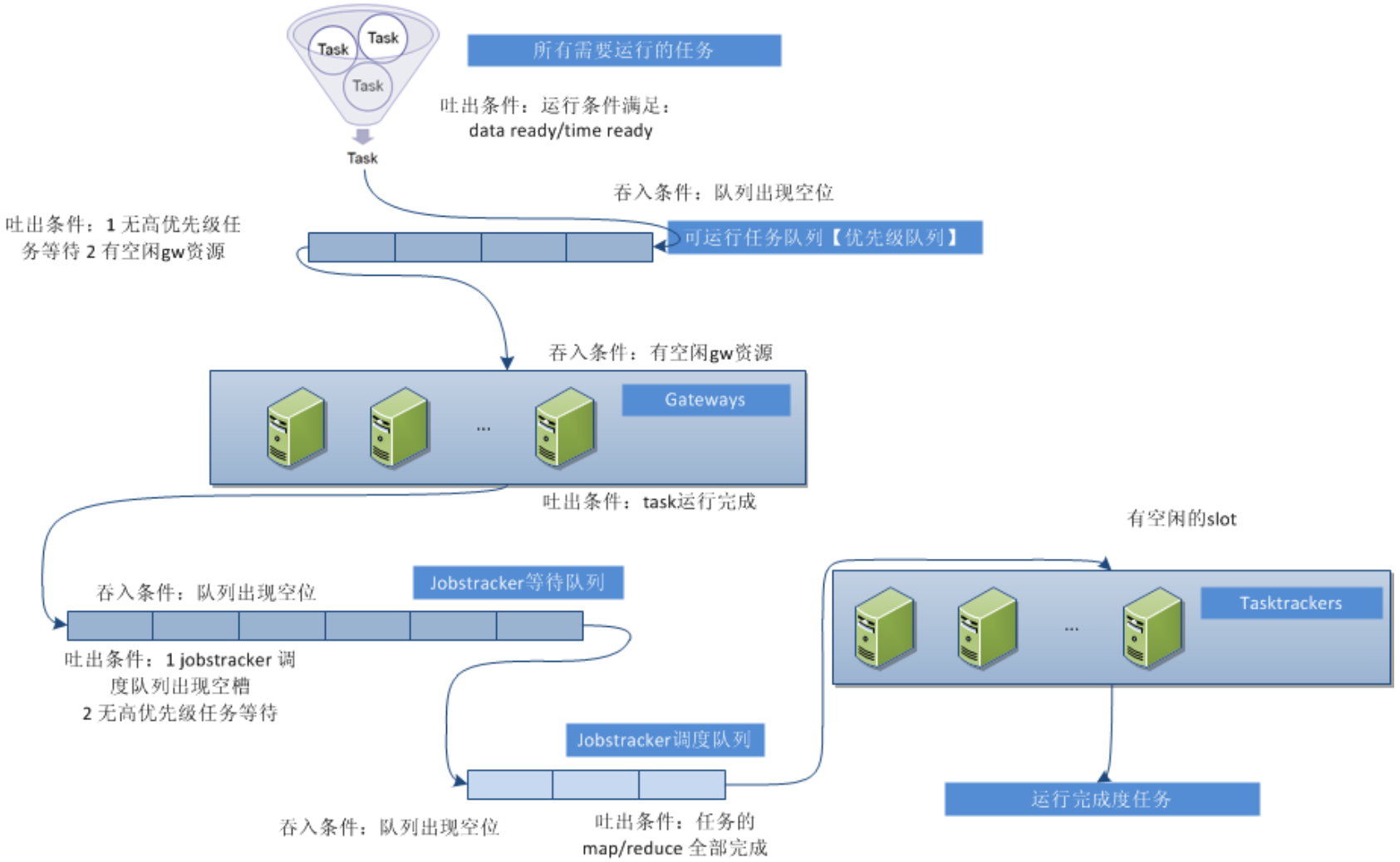
# 基于元数据的分析平台——分析策略概览



# 基于元数据的分析平台——运行数据收集



# 基于元数据的分析平台——宏观分析策略



# 基于元数据的分析平台——定位系统瓶颈

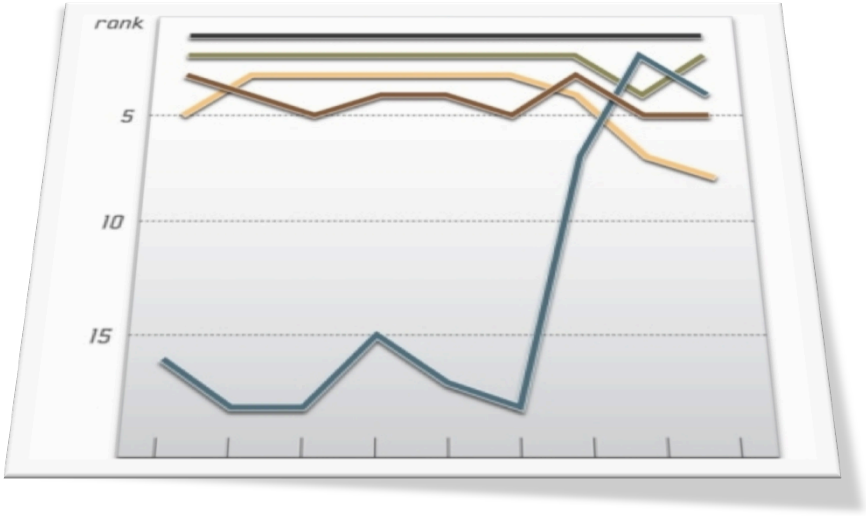
- 每个环节的吞吐能力都是动态变化的。
- 在某个特定时间区间内，整个系统的吞吐能力由吞吐能力最小的一个环节决定。
- 如果需要发现系统的短板，需要对每个环节的吞吐曲线绘制出来。
- 针对系统的短板进行重点优化。
- 对于吞吐能力抖动比较大的环节，需要在前面设置队列进行缓冲。

瓶颈定位方法：

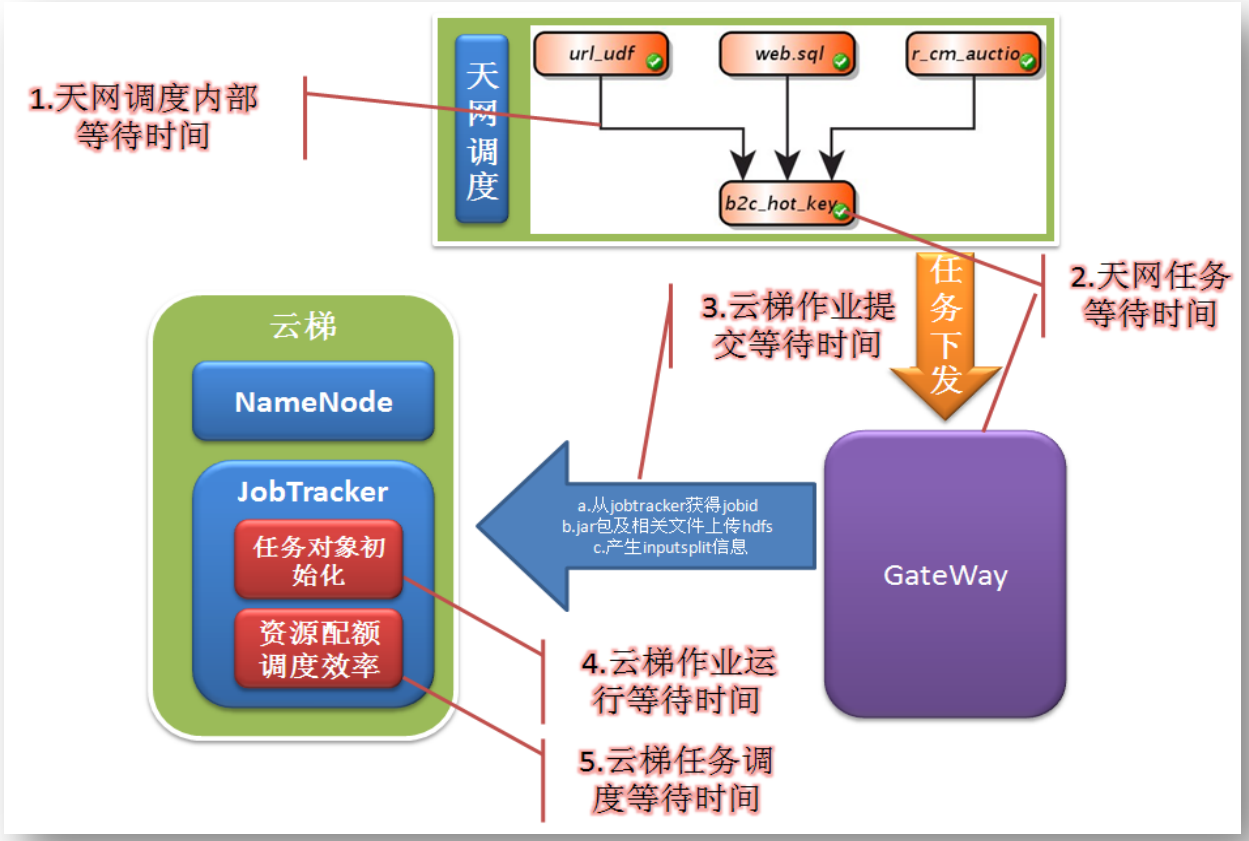
1 每个环节的吞吐曲线

2 两个环节之前缓冲队列的状态曲线

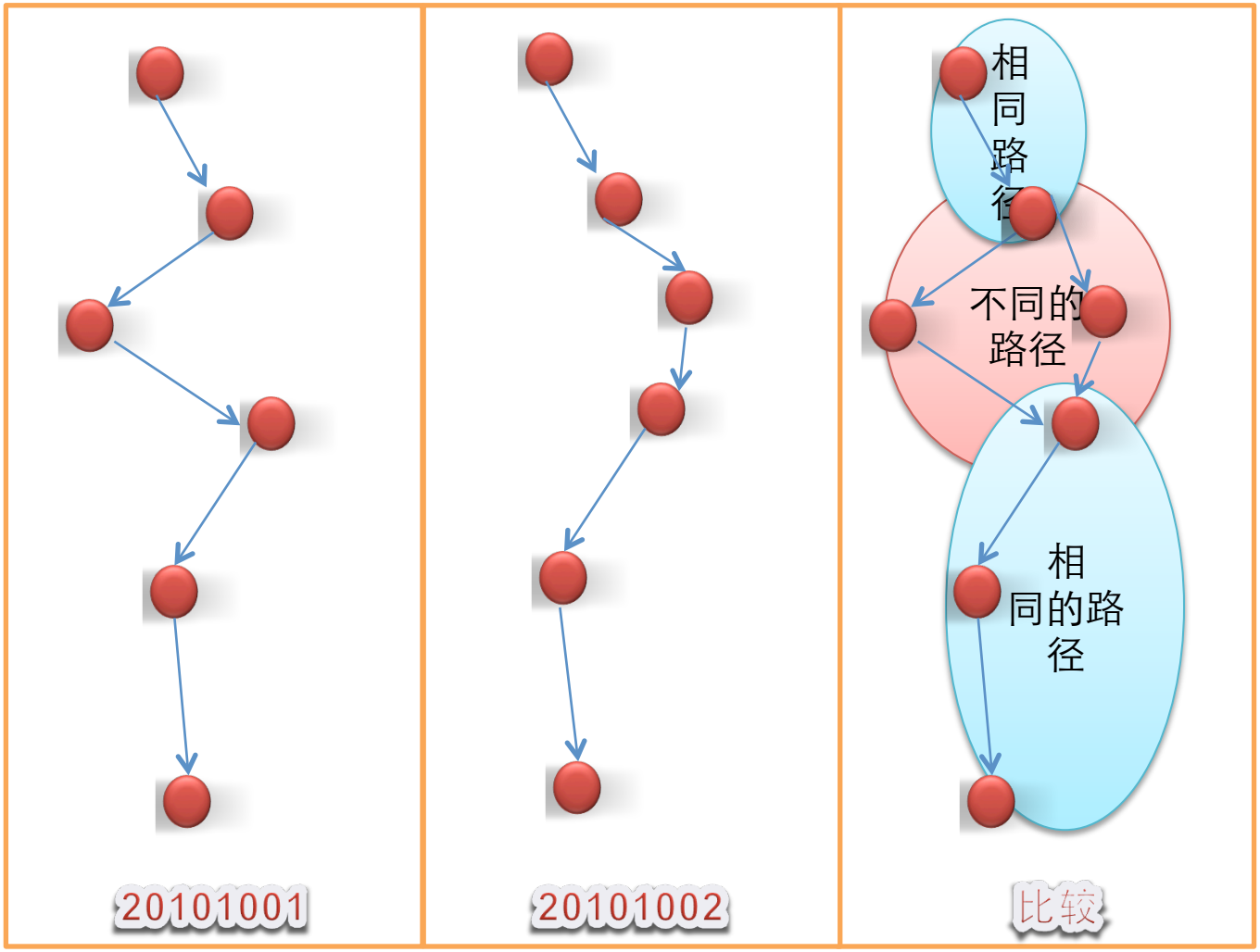
3 统一单位到task级别



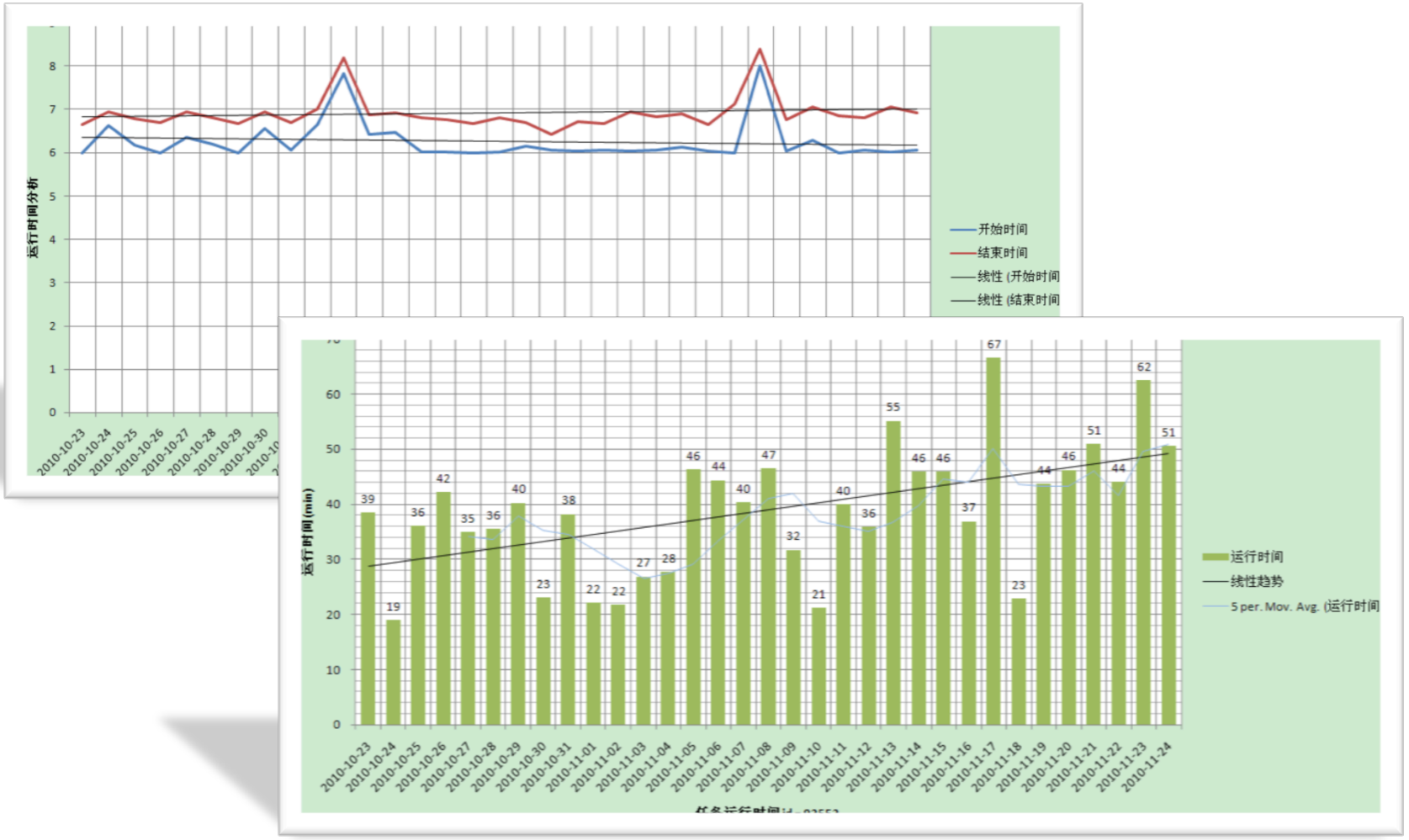
# 基于元数据的分析平台——任务等待时间



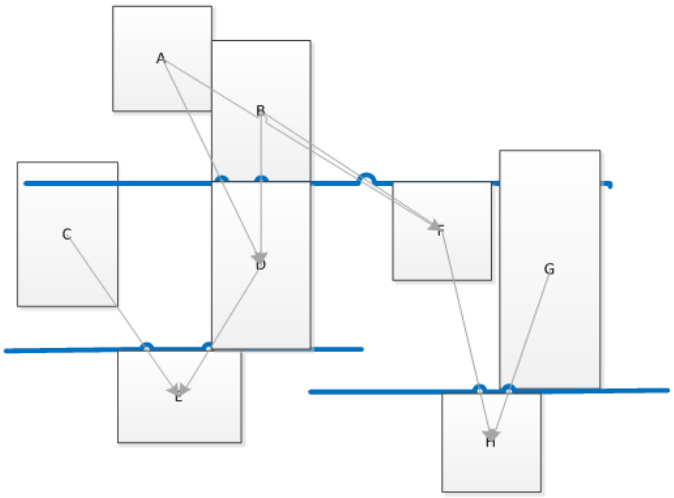
# 基于元数据的分析平台——关键路径分析

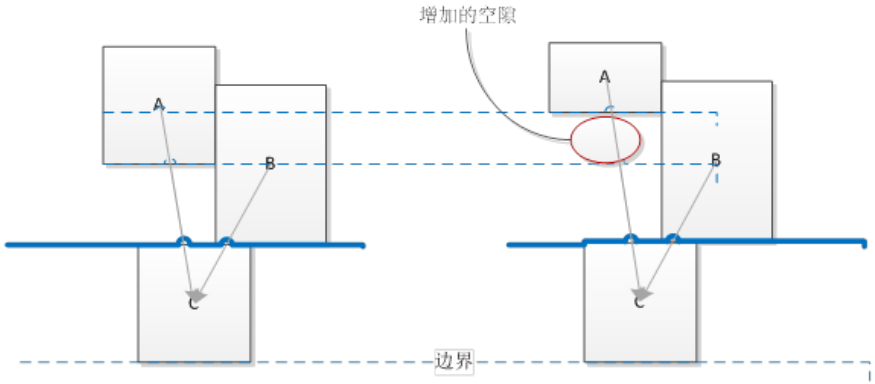


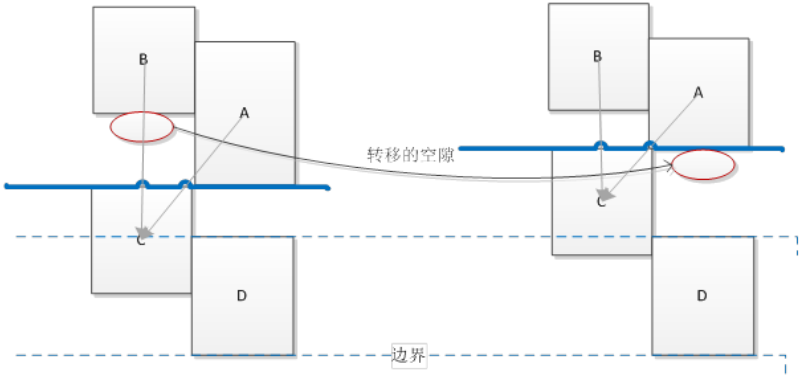
# 基于元数据的分析平台——任务运行趋势

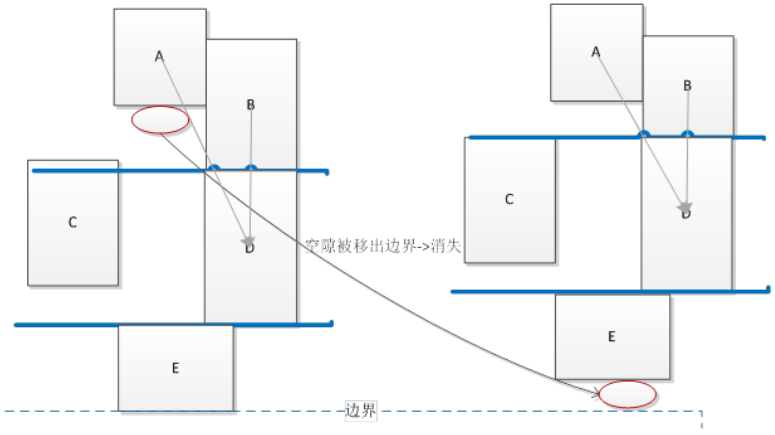


# 基于元数据的分析平台——调度紧凑度分析

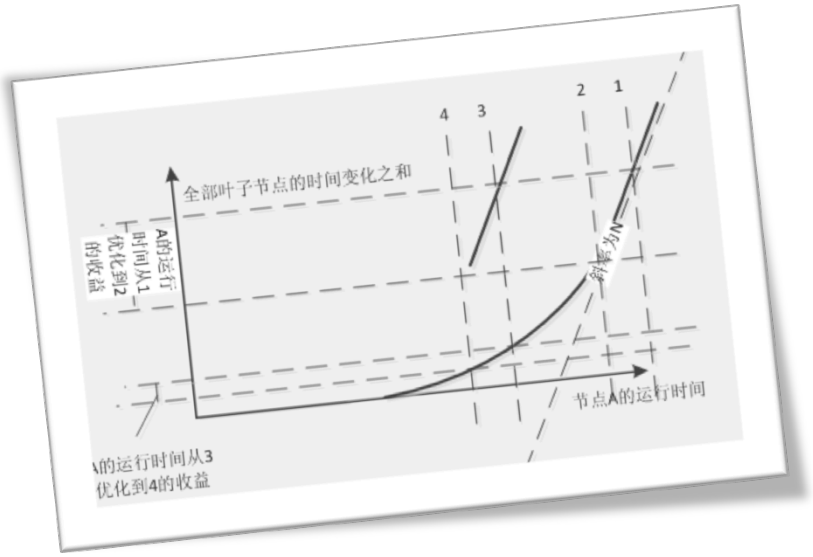


# 基于元数据的分析平台——最值得优化的任务




从关键路径的角度考虑，任务A运行时间变化，对系统对影响可以用下图中阴影的面积来计算,它取决于下面几个因素：

A 任务的当前运行时间——决定了当前位置的曲线斜率。
B 任务在几个叶子节点的关键路径上——决定了当前位置的曲线斜率。
C 其它关联任务的运行时间——决定A何时会离开某个关键路径，也就是决定了斜率的斜率。

最值得优化的任务：
以下三项评分中综合评分最高
1 运行时间长
2 同时处于多个关键路径
3 孔隙度大

# 谢谢大家

# Q&A