HUAWEI

华为云EI

智能流计算：FLINK/SPARK在实时流计算中的应用

常见开源流计算框架


APACHE
STORM™

Alibaba
JStorm


HERON


Apache Apex™


APACHE
nifi


beam

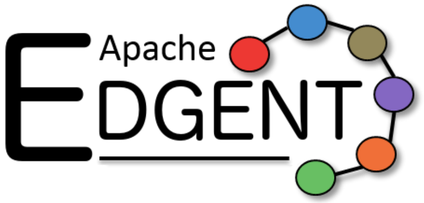

APACHE
Spark™


Flink


Apache Gearpump
Latest Release v0.8.4


APACHE
kafka™
A distributed streaming platform


akka

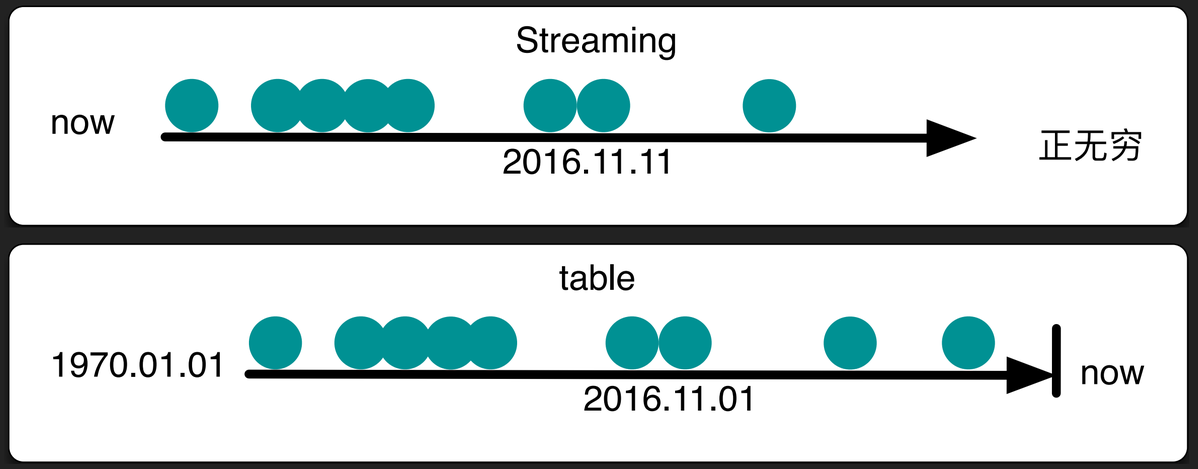

Apache
EDGENT


samza


Apache Flume™


apache
Ignite™

什么是流计算



实时处理当下正在发生的流数据，
逐条进行大数据分析或算法运算。

流计算的印象：

1. 乱序
2. 内存计算
3. 流速不定，数据倾斜
4. long running，永不结束
5. 基于消息事件的逐条处理
6. 可靠快照...

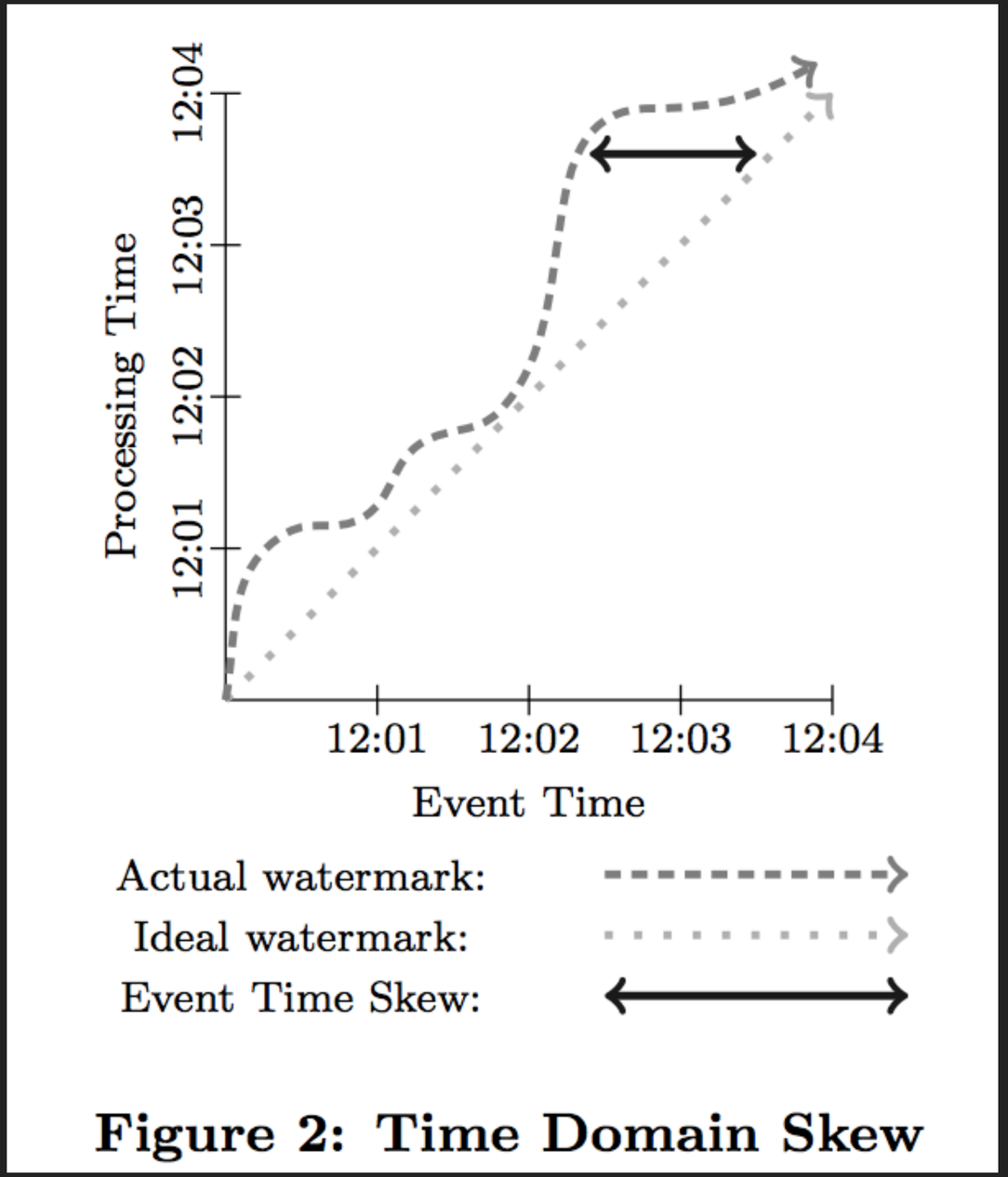


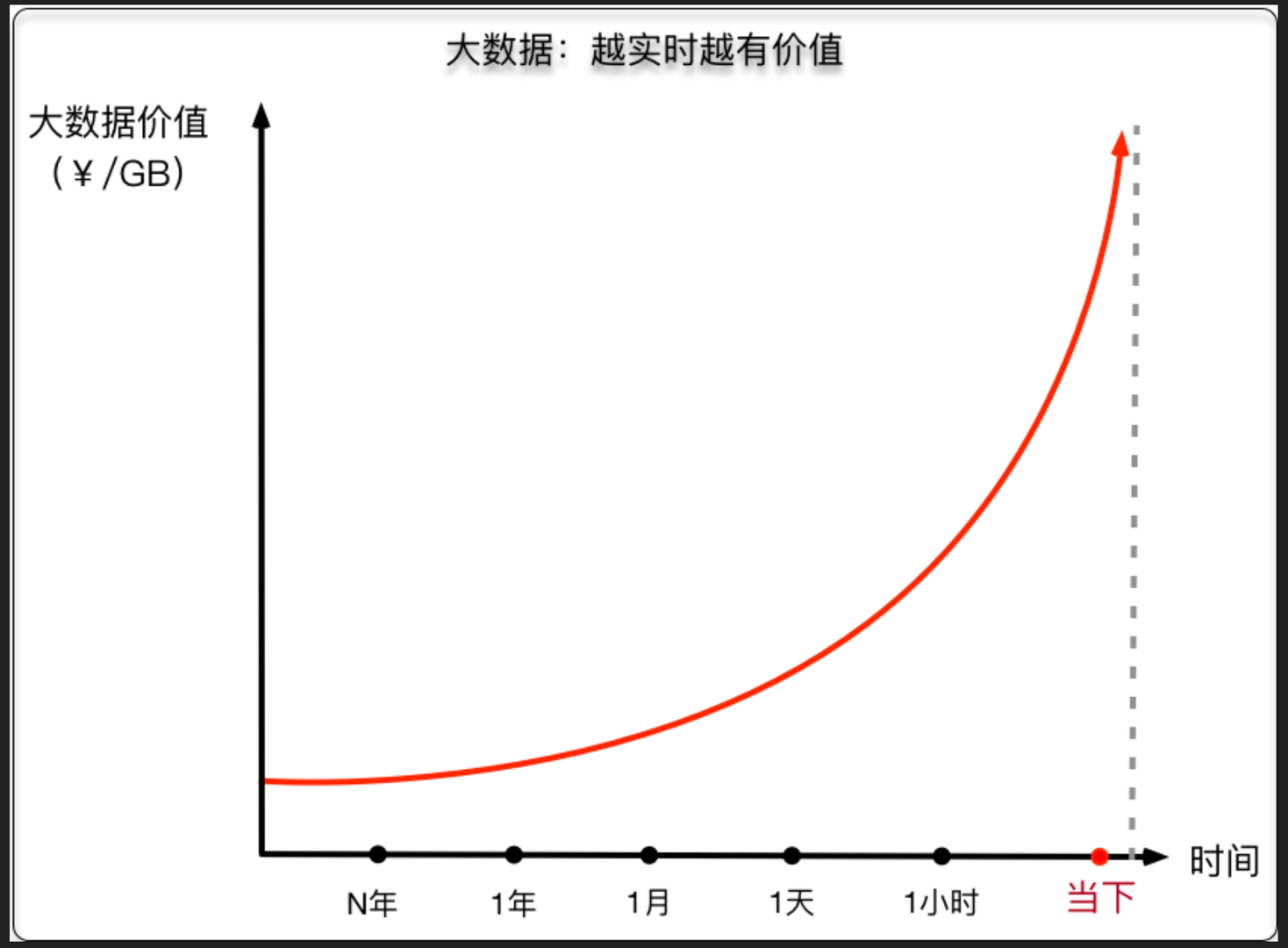
Figure 2: Time Domain Skew

大数据计算：越实时越有价值

新技术的影响：加速度-颠覆
人的耐心程度：加速度-降低
大数据的增速：加速度-增长

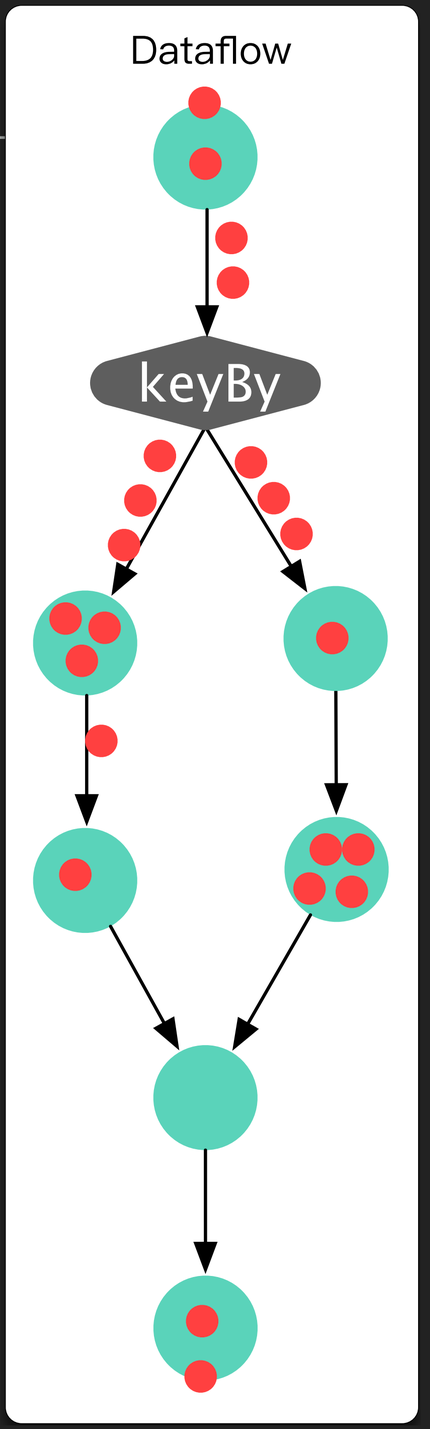
大数据是基础，计算产生增值
实时流计算：最大限度挖掘数据价值

适用场景：
实时推荐(商品/广告) 、实时监控大盘、打车、
金融风控、异常检测、交通、物流、外卖、车联网、智慧城市...



DATAFLOW编程模型

- infinite stream data
- window：分配 / 合并
- trigger and incremental processing

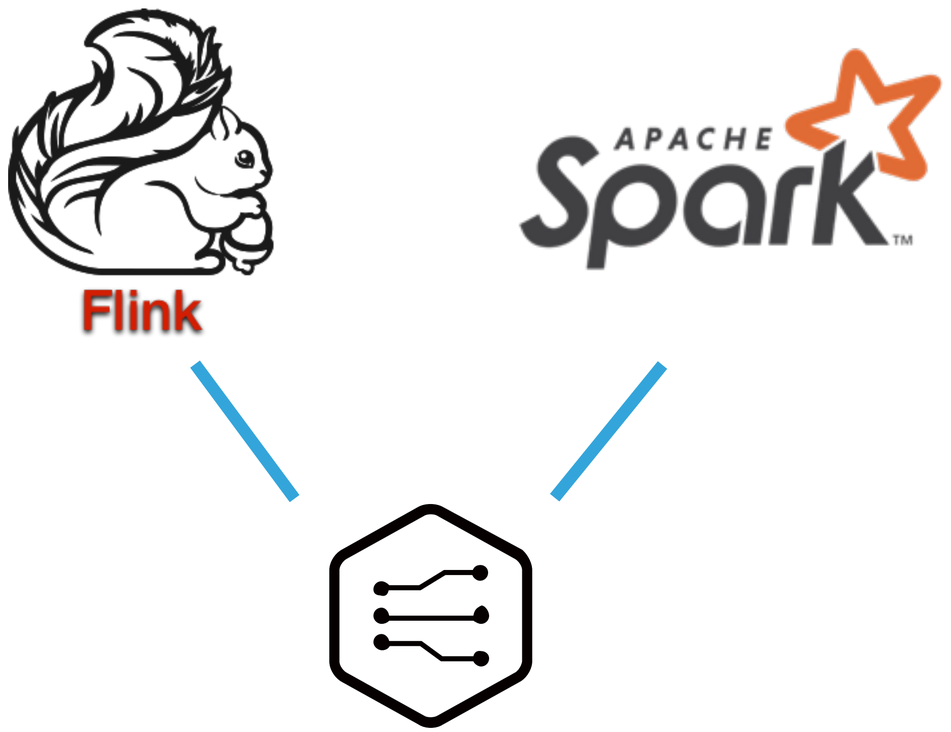


http://people.csail.mit.edu/matei/courses/2015/6.S897/readings/google-dataflow.pdf

https://www.oreilly.com/ideas/the-world-beyond-batch-streaming-101

https://www.oreilly.com/ideas/the-world-beyond-batch-streaming-102

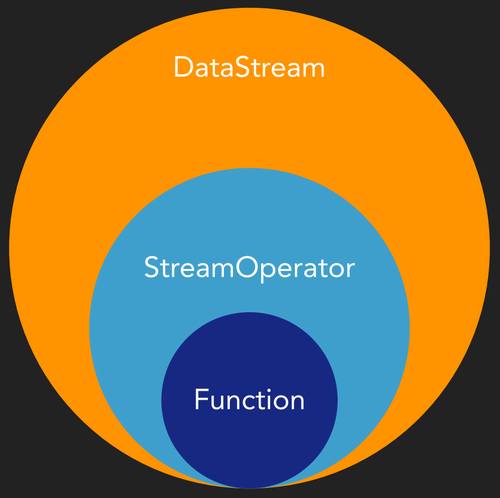
我们的选择：FLINK + SPARK



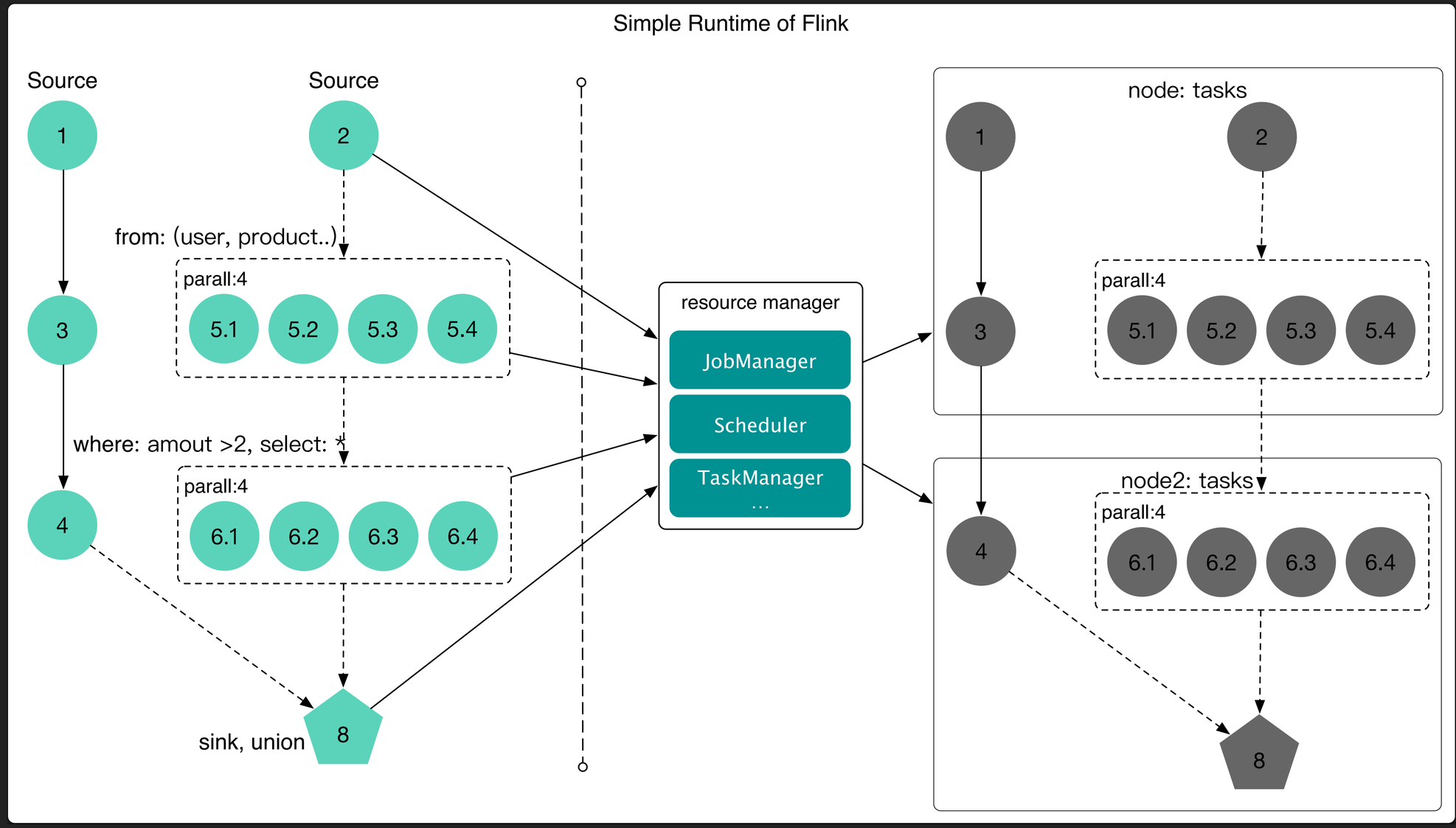
FLINK: 运行时概要

API或函数的调用链，由逻辑上的算子表示，经过逻辑优化，调度到物理节点上执行。

算子启动顺序为：dataflow中最后一个节点先起，依次向上启动；算子之间通过netty通信，由网络buffer实现自然反压。



apply()



FLINK内核运行时分析（大图详见附件）

transformations

	StreamTransformation	StreamOperator	input: StreamTransformation
0	OneInputTransformation	map # StreamMap	SourceTransformation
1	OneInputTransformation	flatMap # StreatFlatMap	0: OneInputTransformation
2	SinkTransformation	print # StreamSink	1: SinkTransformation

- 第一步：逻辑关系解析
- API包含：Function，StreamTransformation，返回类型为DataStream
- API，输入：fun: T => R包装成相应Function，输出：DataStream/WindowStream
- context把所有的StreamTransformation保存到列表中
- DataStream构成调用链
- 惰性执行：在物理资源上实际执行时，反序列化fun，隐式调用scala的apply()

- 第一阶段产物：StreamTransformation

FLINK内核运行时分析（大图详见附件）

- 第三步：构建StreamGraph，DAG
- 1. 入口：env.execute()，根据配置进入本地环境或分布式环境，执行应用程序
- 2. StreamTransformation包含上下游调用关系，构建StreamNode：入边-计算-出边 关系列表
- 3. 生成StreamGraph，逻辑执行图

StreamNode ## graph

	field	type
	id	int
	transformationId	String
	operatorName	string
	maxParallelism	int
	bufferTimeout	long
	slotSharingGroup	String
	statePartitioner1	KeySelector
	statePartitioner2	KeySelector
	stateKeySerializer	TypeSerializer
	operator	StreamOperator
	inEdges	List<StreamEdge>
	outEdges	List<StreamEdge>
	operator	StreamOperator
	jobVertexClass	Class
	inputFormat	InputFormat

StreamGraph

	field	type
	streamNodes	Map[Integer, StreamNode]
	sources	Set[Integer]
	sinks	Set[Integer]
	virtualSelectNodes	Map[Integer, Tuple[Integer,List[String]]]
	stateBackend	AbstractStateBackend
	virtualPartitionNodes	Map[Integer, Tuple2[Integer, StreamPartitioner]]

FLINK内核运行时分析（大图详见附件）

- 第二步：构建ExecutionGraph
- 1. StreamGraph转换成JobGraph
- 2. Standalone/mesos/yarn/k8s客户端提交JobGraph
- 3. 连通JobManager（独立启动集群）
- 4. 上传JobGraph及其依赖，反序列化
- 5. ExecutionJobVertex就是可执行task

executionGraph

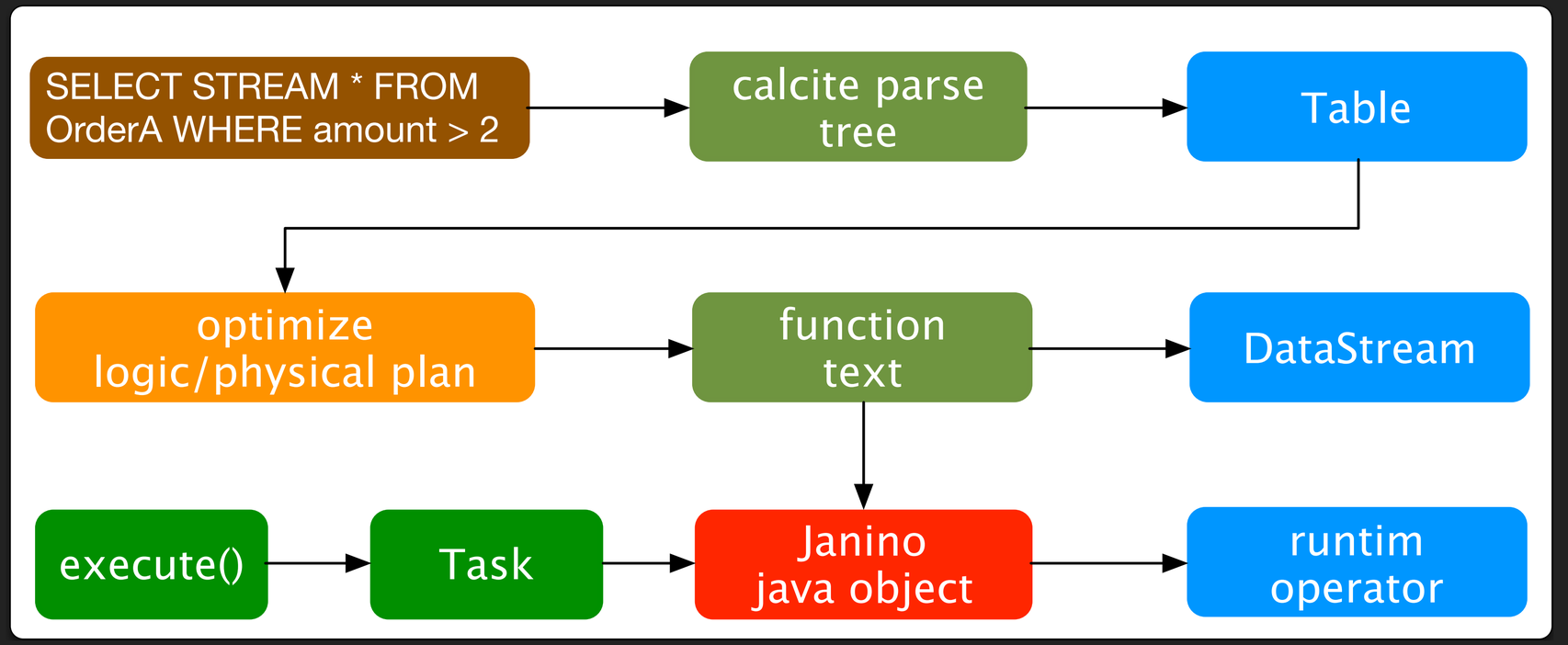
ExecutionGraph		
	field	type
	jobInformation	JobInformation
	ioExecutor	Executor # Thread pool
	currentExecutions	ConcurrentHashMap<ExecutionAttemptID, Execution>
	slotProvider	SlotProvider
	restartStrategy	RestartStrategy
	userClassLoader	ClassLoader
	futureExecutor	Executor # Thread pool
	checkpointStatsTracker	CheckpointStatsTracker
	checkpointCoordinator	CheckpointCoordinator
	scheduleMode	LAZY_FROM_SOURCES | EAGER
	tasks	ConcurrentHashMap<JobVertexID, ExecutionJobVertex>
	verticesInCreationOrder	List<ExecutionJobVertex>
	jobStatusListeners	List<JobStatusListener>

FLINK内核运行时分析（大图详见附件）

- 第4步：申请资源，执行任务（算子）
- 1. Slot、Execution、Task
- 2. 向RM申请资源slot槽位，slot对应逻辑和物理执行单元
- 3. slot申请后，在其中释放Execution，向TaskManager提交Task，run线程
- 4. Task包含AbstractInvokable（BatchTask、StreamTask..）,调用invoke函数
- 5. 反向启动相应的function

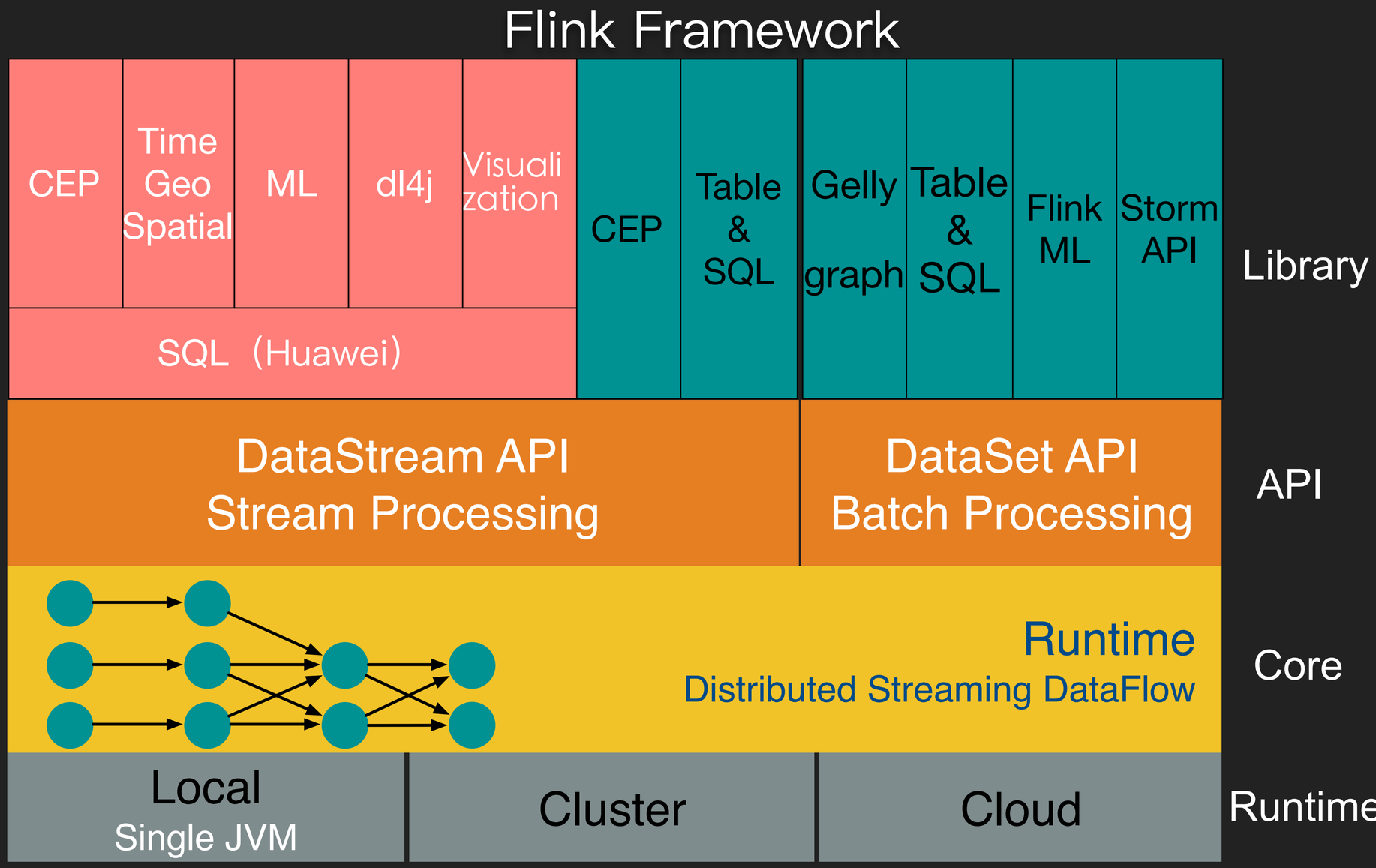
FLINK SQL运行时分析（大图详见附件）

- Flink SQL parse, optimize, codegen, pre-compile => DataStream



FLINK技术栈

- 完整Dataflow框架
- SQL优先
- 上层扩展



SPARK VS FLINK

- Flink：
  - dataflow模型
  - 丰富易用的Stream API
  - 功能完善：SQL、Table、CEP、ML、Graph
- Spark：
  - Structured Streaming时延缩短
  - 社区活跃
  - 生态完善

实时流计算未来可能如何发展

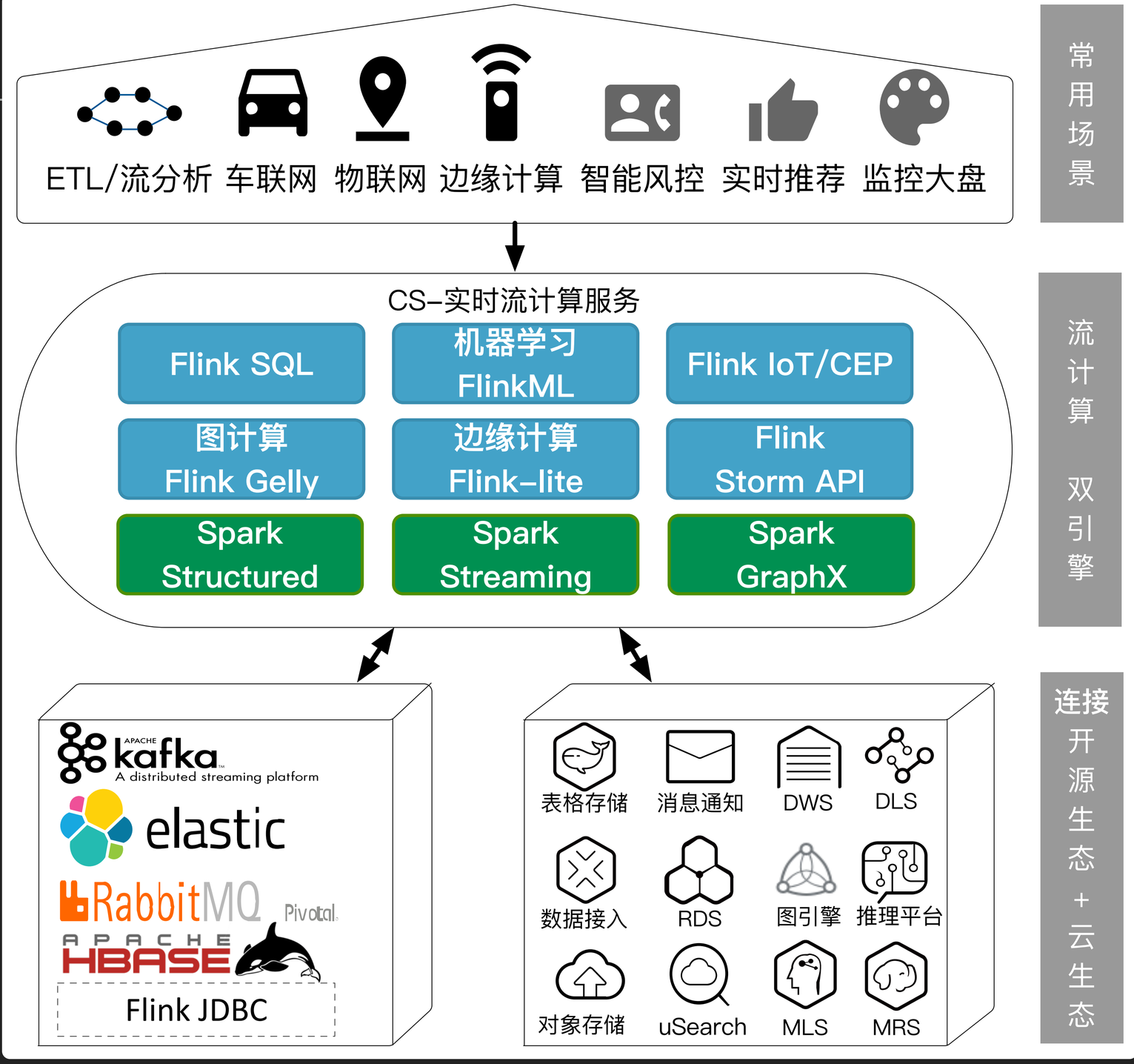
- StreamingML and StreamingAutoML
- StreamingPS with GPU support
- StreamingGraph
- FPGA speed up network and key SQL operator
- uniform StreamSQL DSL support spark/flink/kafka
- Streaming Storage with low latency

华为云在实时流计算方面的努力

实时流计算服务-CLOUDSTREAM-CS

- 实时流计算服务（Cloud Stream Service, 简称CS）提供实时处理流式大数据的全栈能力, 简单易用, 即时执行Stream SQL或自定义作业。无需关心计算集群, 无需学习编程技能。完全兼容Apache Flink和Spark API
- Spark, Flink二合一
- 不锁定客户，完全兼容开源

https://www.huaweicloud.com/product/cs.html



CLOUDSTREAM特点

易用

- StreamSQL编辑器
- StreamSQL可视化拖拽
- 在线调试
- 作业可视化监控

按需计费 & 包年包月

按实际使用量计费
用多少付费多少
1 SPU/小时 = 0.5元
包年包月有优惠

开箱即用

无需关心基础设施
即时执行业务作业

低延时 高吞吐

- 毫秒级时延
- 每秒处理百万消息

完整生态

- 开源生态
- 连通云存储和AI服务

安全可靠

首创完全托管的独享集群

- 物理隔离和安全策略
- 华为软件安全加固

特点

- 所见即所得
- SQL满足80%业务
- 强大的SQL特性
- SQL连接一切

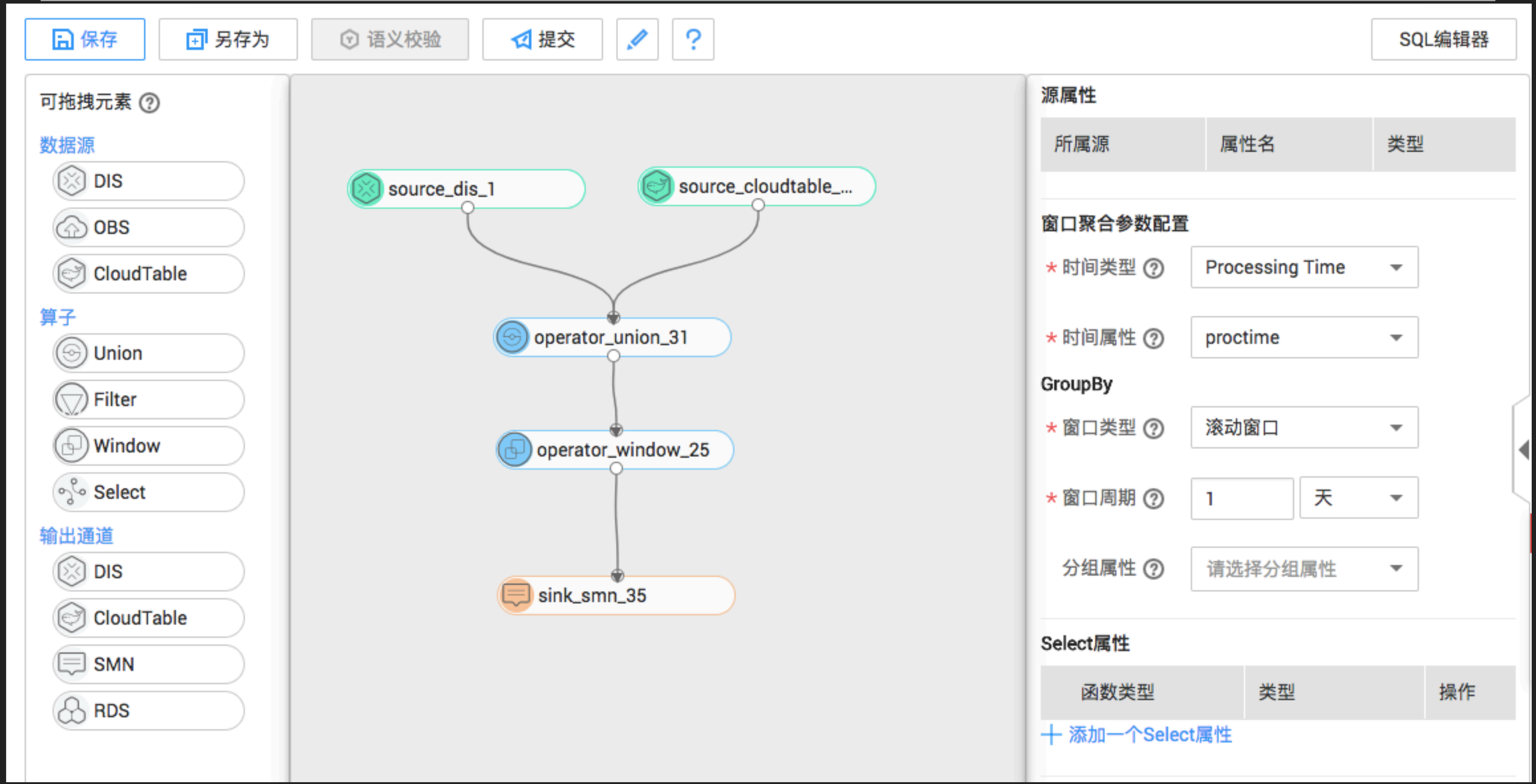
保存 另存为 语义校验 调试 提交 设为模板

```
  * encode:  结果编码方式，可以为csv或者json
  * field_delimiter: 当编码格式为csv时，属性之间的分隔符
  **/
CREATE SINK STREAM fake_licensed_car (
  car_license_number STRING,
  first_zone String,
  second_zone String
)
WITH (
  type = "dis",
  region = "cn-north-1",
  channel = "csoutput",
  partition_key = "car_license_number",
  encode = "csv",
  field_delimiter = ","
);

/** 输出套牌车信息 **/
INSERT INTO fake_licensed_car
SELECT * FROM camera_license_data MATCH_RECOGNIZE
(
  PARTITION BY car_license_number
  ORDER BY proctime
  MEASURES A.car_license_number as car_license_number, A.camera_zone_number as first_zone,
  ONE ROW PER MATCH
  AFTER MATCH SKIP TO LAST C
  PATTERN (A B+ C+)
  WITHIN interval '5' minute
  DEFINE
    B AS B.camera_zone_number <> A.camera_zone_number,
    C AS C.camera_zone_number = A.camera_zone_number
) MR;
```

错误: 0 行 78, 列 6

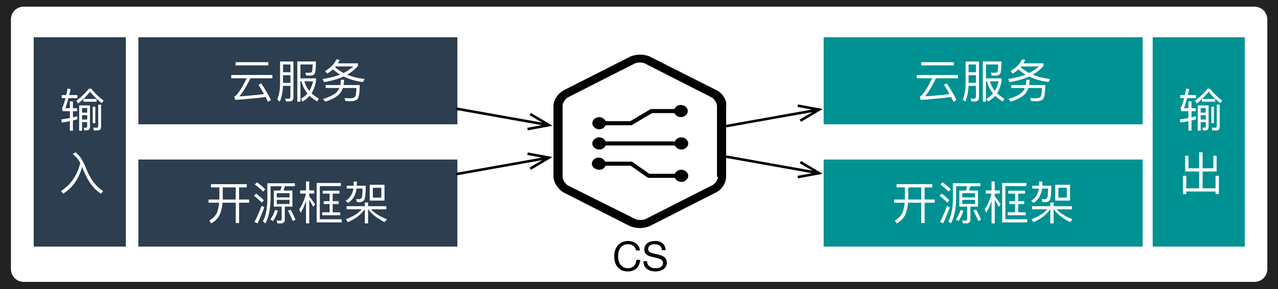
2. 可视化SQL编辑器，SINK数据可视化（下周上线）

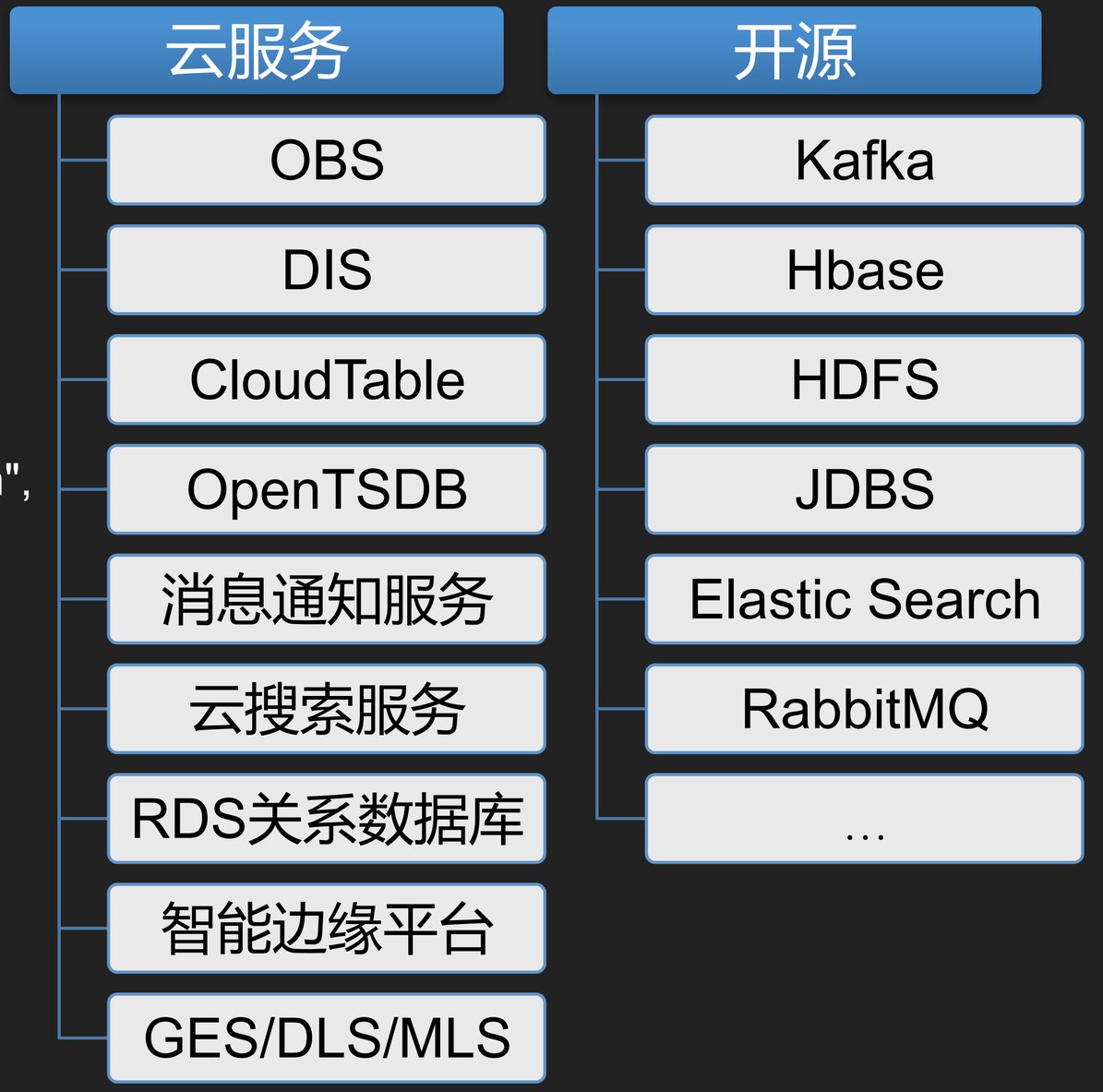


3. 完整全生态：开源生态+云原生生态

示例：

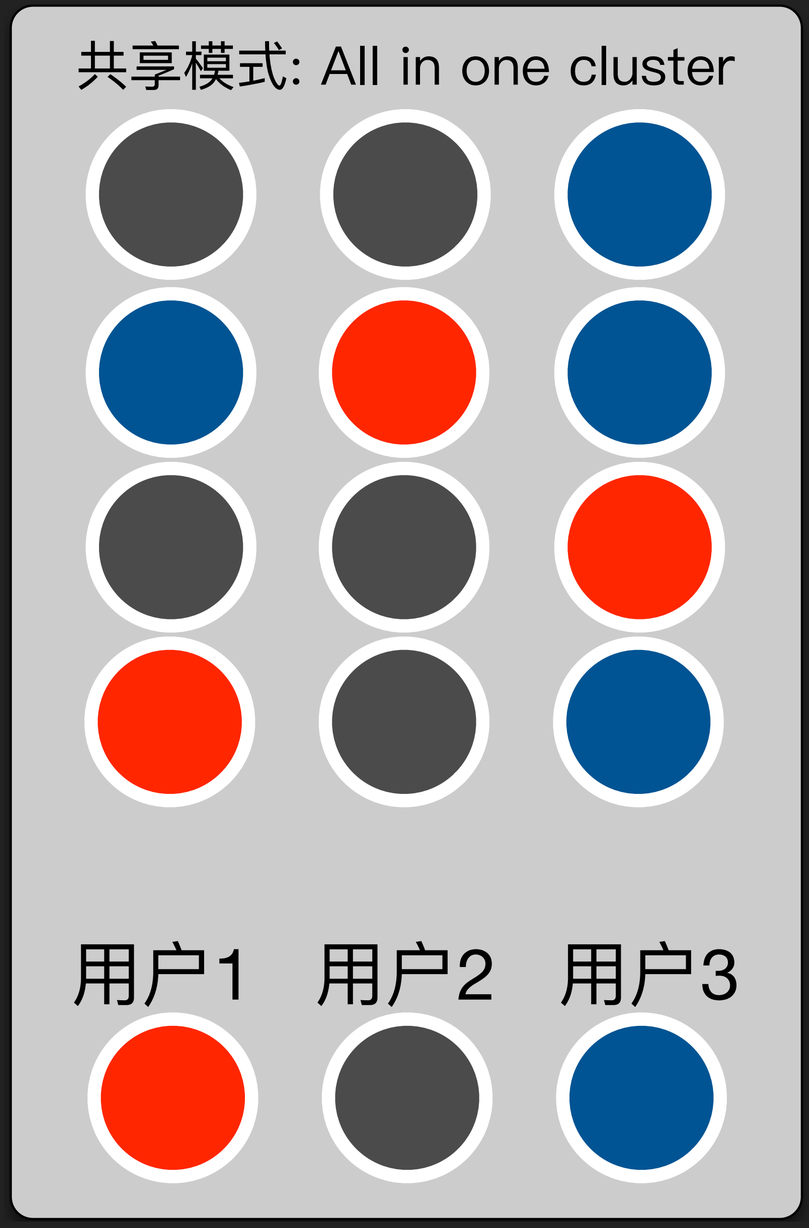
```
CREATE SINK STREAM yaw_warning (
  MessageContent STRING /* 偏航消息内容 */
)
WITH (
  type = "smn",
  region = "cn-north-1",
  topic_urn = "urn:smn:cn-
north-1:a77d6595e37d443fab32d1db9739ed23:Yaw_alarm",
  message_subject = "Yaw_alarm",
  message_column = "MessageContent"
);
```

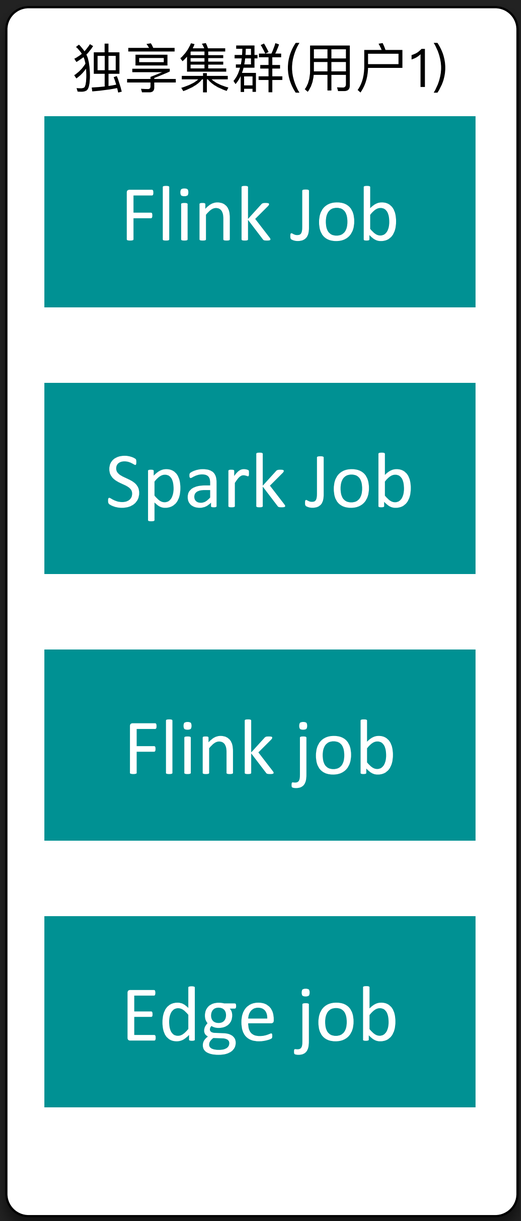




独享集群特点

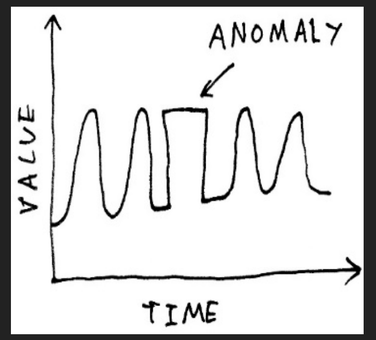
- 物理资源隔离
  - 专用集群，无多用户交叉
- 业务隔离
  - 如消防IoT设备在同一个集群中管理连接、数据和计算，无干扰
- 沙箱问题
  - 沙箱无法解决所有安全问题，矛盾。通过硬件和VM隔离。粗暴有效
- 弹性伸缩
- 按需收费，作业用多少算多少
- 包周期计费，更多折扣

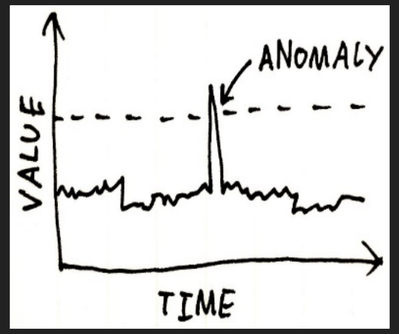


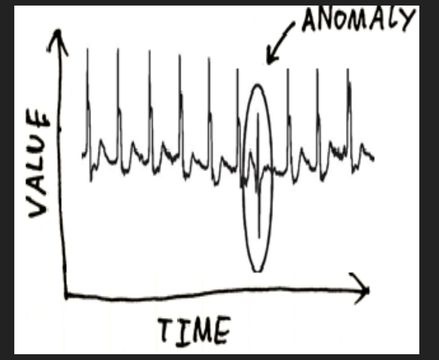


StreamingML:

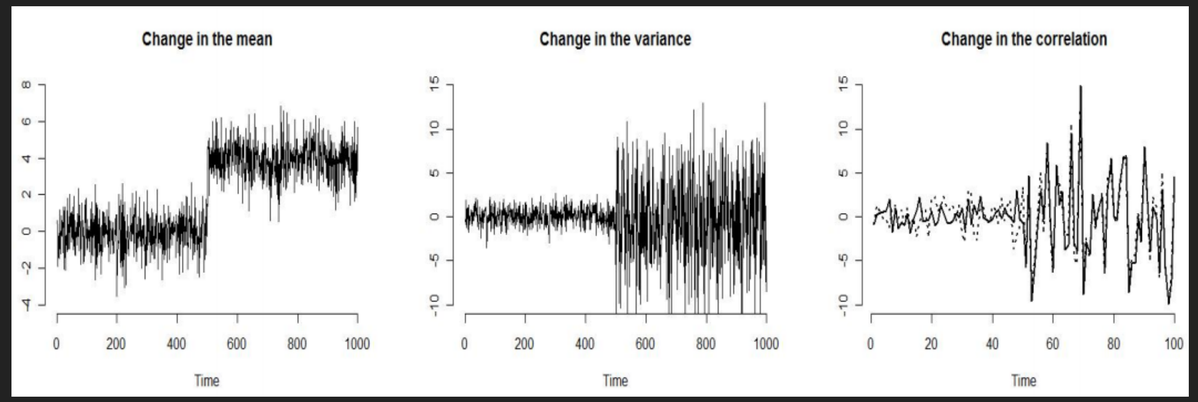
- 流式随机森林：实时故障检测
- 特征工程：特征抽取函数
- 为何发展在线机器学习?







- 更多分类和回归算法
- 使用场景：
  - 实时推荐
  - 实时故障检测



6. 时空分析：基于时间地理位置的实时大数据分析（详见官方文档）

DDL for Time Geospatial – 基本元素

1. ST_Point(latitude, longitude) 纬度和经度构成点
2. ST_Line(array[point1...pointN]) 多点构成线
3. ST_POLYGON(array[point1...point1]) 多点构成多边形
4. ST_CIRCLE(point, radius) 点和半径构成圆

SQL Geospatial Scalar Functions - 基本操作

1. ST_DISTANCE 计算两点间**距离**

示例： select *ST_DISTANCE(ST_POINT(x1, y1), ST_POINT(x2, y2))* FROM input

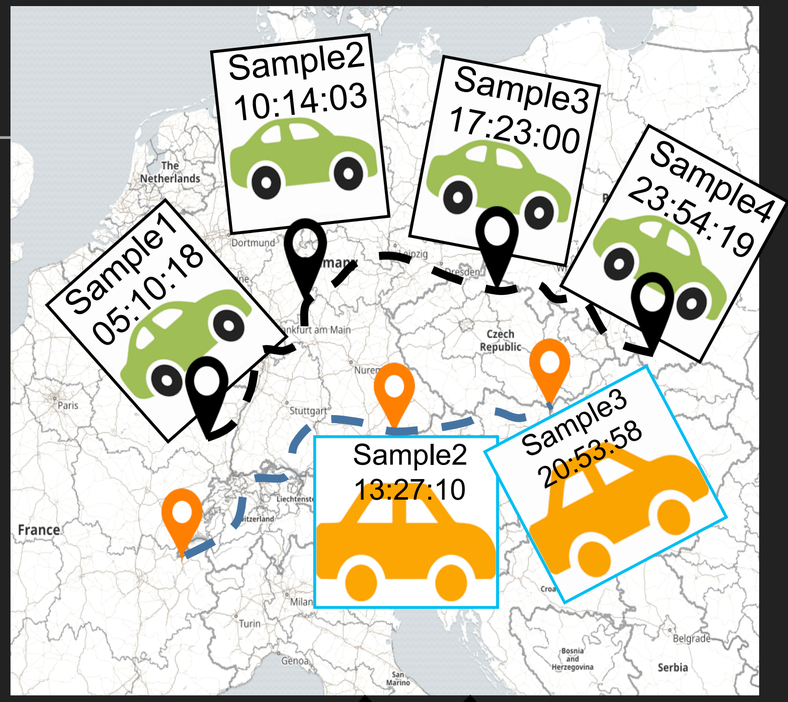
2. ST_PERIMETER 计算多边形**周长**

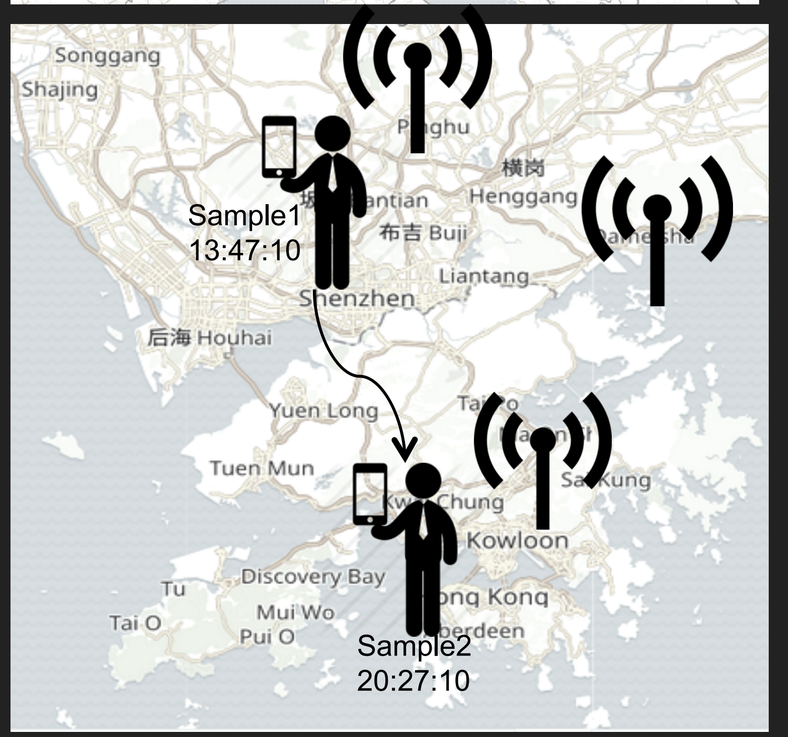
示例： Select *ST_PERIMETER(ST_POLYGON(ARRAY[ST_POINT(x11, y11), ST_POINT(x12, y12), ST_POINT(x11, y11)])* FROM input

3. ST_AREA (polygon) 计算**多边形面积**

示例： Select ST_AREA(ST_POLYGON(ARRAY[ST_POINT(x11, y11), ST_POINT(x12, y12), ST_POINT(x11, y11)]) FROM input

4. ST_OVERLAPS (polygon1, polygon2) **多边形是否相交**
5. ST_INTERSECTS 检查**两条线是否相交**
6. ST_WITHIN 检查一个点是否**被包含**在一个几何形状中
7. ST_CONTAINS 检查一个多边形是否**包含**另一多边形
8. ST_COVERS 检查一个多边形是否被另一多边形**覆盖**
9. ST_DISJOINT 检查两个多边形是否**不相交**





成本对比

对比项目	线下搭环境	实时流计算服务	节省
硬件成本	8 * 3 = 24 万元 *单机8万元（仅供参考）*	0.5 * 20 * 24 * 30 * 12 * 3 = 25.9万元 *单SPU 0.5元/小时, 单机对应20个SPU*	
1运维人力	20 万/人年	0	
水电、制冷、机房费用	7.63万/年	0	
合计	51.63万元	25.9万元,	42.9%

相同的计算能力：节省42.9%

包年包月，更优惠

持续推出更劲爆的创新特性

https://www.huaweicloud.com/product/cs.html