

实时流式数据处理及应用

徐正君

Zheng-jun.xu@sap.com

议程

- ▶ 为什么需要流计算？
- ▶ 什么是流计算？
- ▶ 流计算产品介绍
- ▶ 流计算实例



管理人员/业务人员的困扰



传统技术的代价

- ▶ 用数据库是不是更好呢？
 - ▶ 数据量
 - ▶ 时间
 - ▶ 其他



什么是流计算？

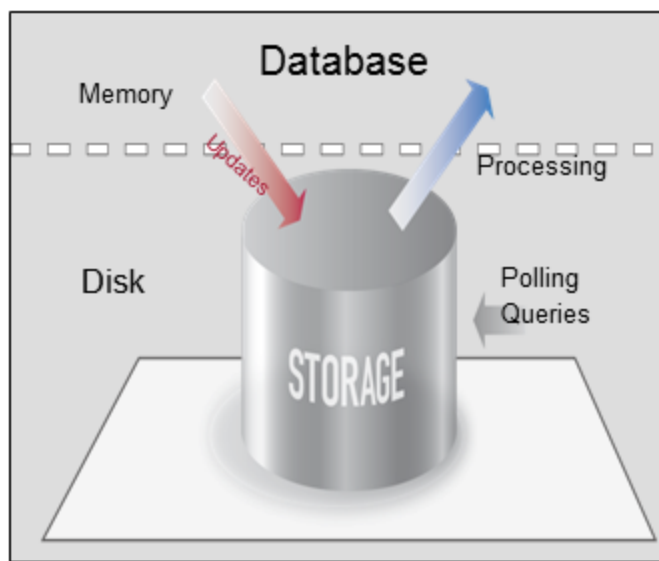


数据库技术

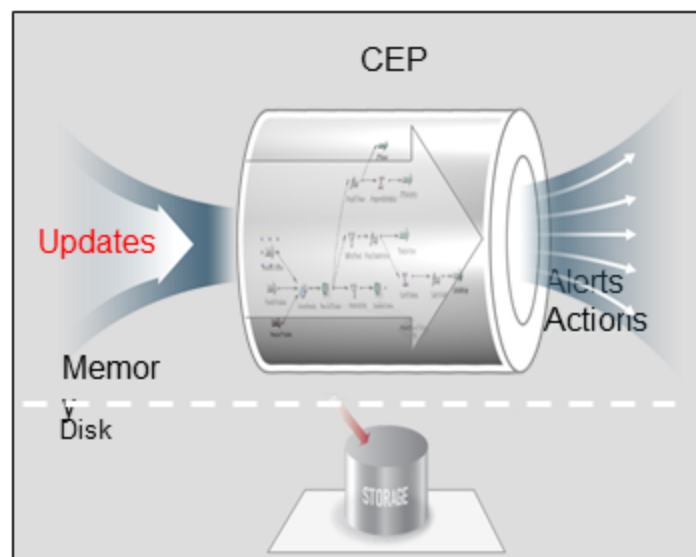
流计算技术



什么是流计算？



- Hard drive based
- Store data first, then query and process
- Query and then get data



- Memory-based ✓
- Define events -> filter, identify ✓
- Data -> Query ✓

数据流的特征

- ▶ 数据连续不断。
- ▶ 数据来源众多，格式复杂。
- ▶ 数据量大，但是不十分关注存储。
- ▶ 注重数据的整体价值，不要过分关注个别数据
- ▶ 数据流顺序颠倒，或者不完整。



对处理工具的要求

- ▶ **高性能**
 - ▶ 这是处理大数据的基本要求，例如：几十万条/秒
- ▶ **实时性**
 - ▶ 必须保证一个较低延迟时间，最好是毫秒级
- ▶ **分布式**
 - ▶ 支持大数据的基本架构，必须能够平滑扩展 (Scale Out)
- ▶ **易用性**
 - ▶ 能够快速开发和部署
- ▶ **海量数据**
 - ▶ TB级...PB级
- ▶ **提供各类计算**
 - ▶ 不仅仅是转发
- ▶ **支持复杂条件的搭配**
 - ▶ 业务模型
- ▶ **实时响应**
 - ▶ 秒级...毫秒级
- ▶ **结果精确**



你需要流计算吗？

- ▶ 是否对数据的实时性有迫切需求？
 - ▶ 是否更关注对当前数据的分析与响应？
 - ▶ 高吞吐量，低延迟事件流处理的场景！
 - ▶ 当机会和威胁在不可预测的时间出现时，事件驱动的系统必须在外部的确定的时间响应事件。这些系统对环境更改时采取行动，而不只是按照预先计划的时间表采取行动。通常，流计算适用于需要将许多组件实时聚集在一起以便完成任务的业务场景。如果您在处理动态流程（其特征更改得相当频繁）、非线性流程（其计时和顺序不可预测，因而无法预先定义）和需要实时响应外部事件的流程（连续地受到系统必须适应的外部事件的影响），流计算是最适合的。
-

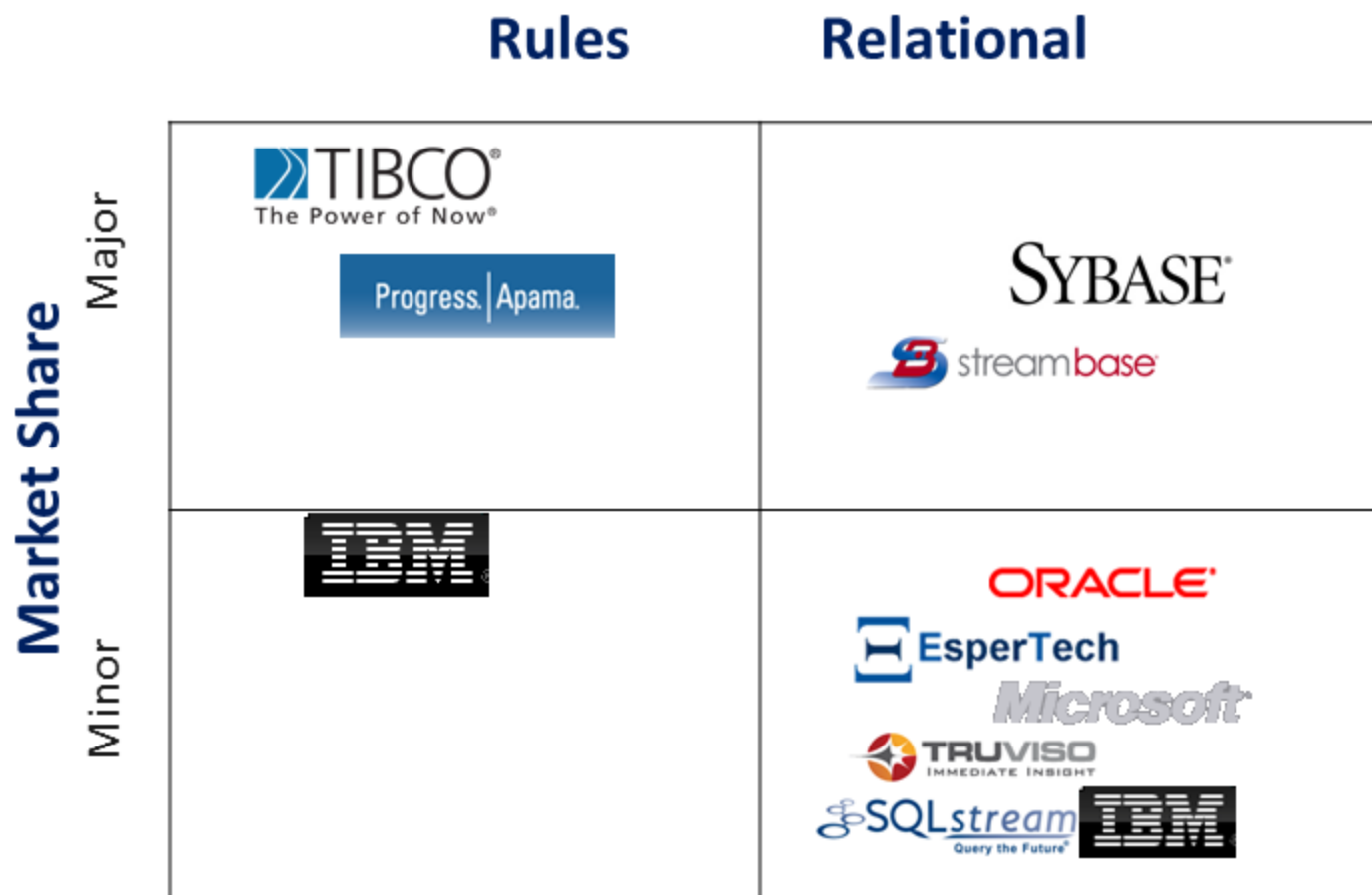
常用的开源流计算产品

- ▶ Puma——Facebook
- ▶ Storm——Twitter
- ▶ S4 (Simple Scalable Streaming System) ——Yahoo!

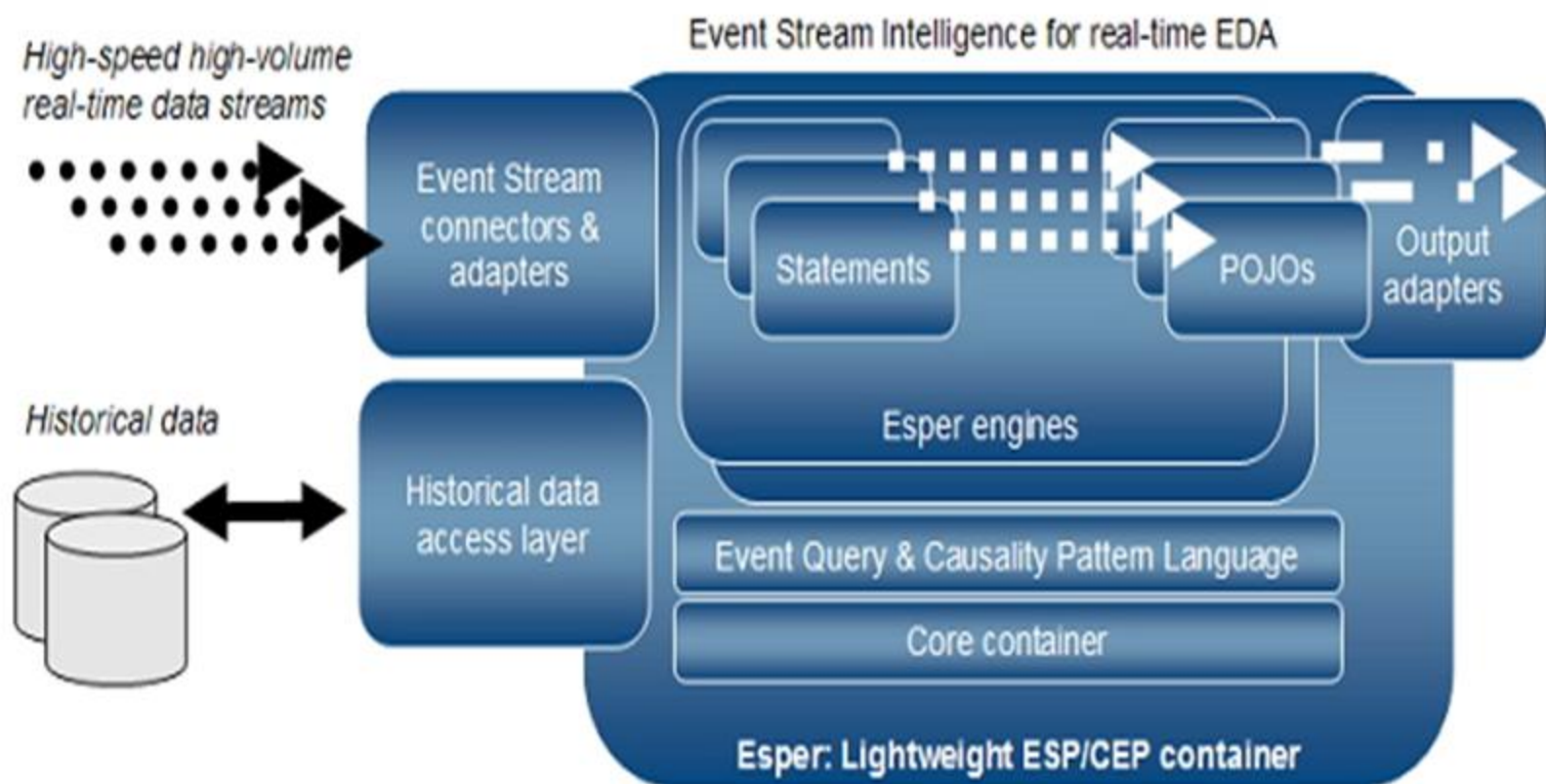
Feature	Puma	Storm	S4	Comments
Development	Java	Clojure	Java	Different systems towards different application requirement
HA	Primary Standby	Upstream Backup	Primary Standby	
Precise Recovery	No	No	No	
Architecture	Symmetric	Master-Slave	Symmetric	
Resource Utilization	Low	High	Low	
Recovery Time	Short	Long	Long	
State Persistence	Yes	No	Yes	
De-dup	Yes	No	No	

***Universal Data Stream Processing System**

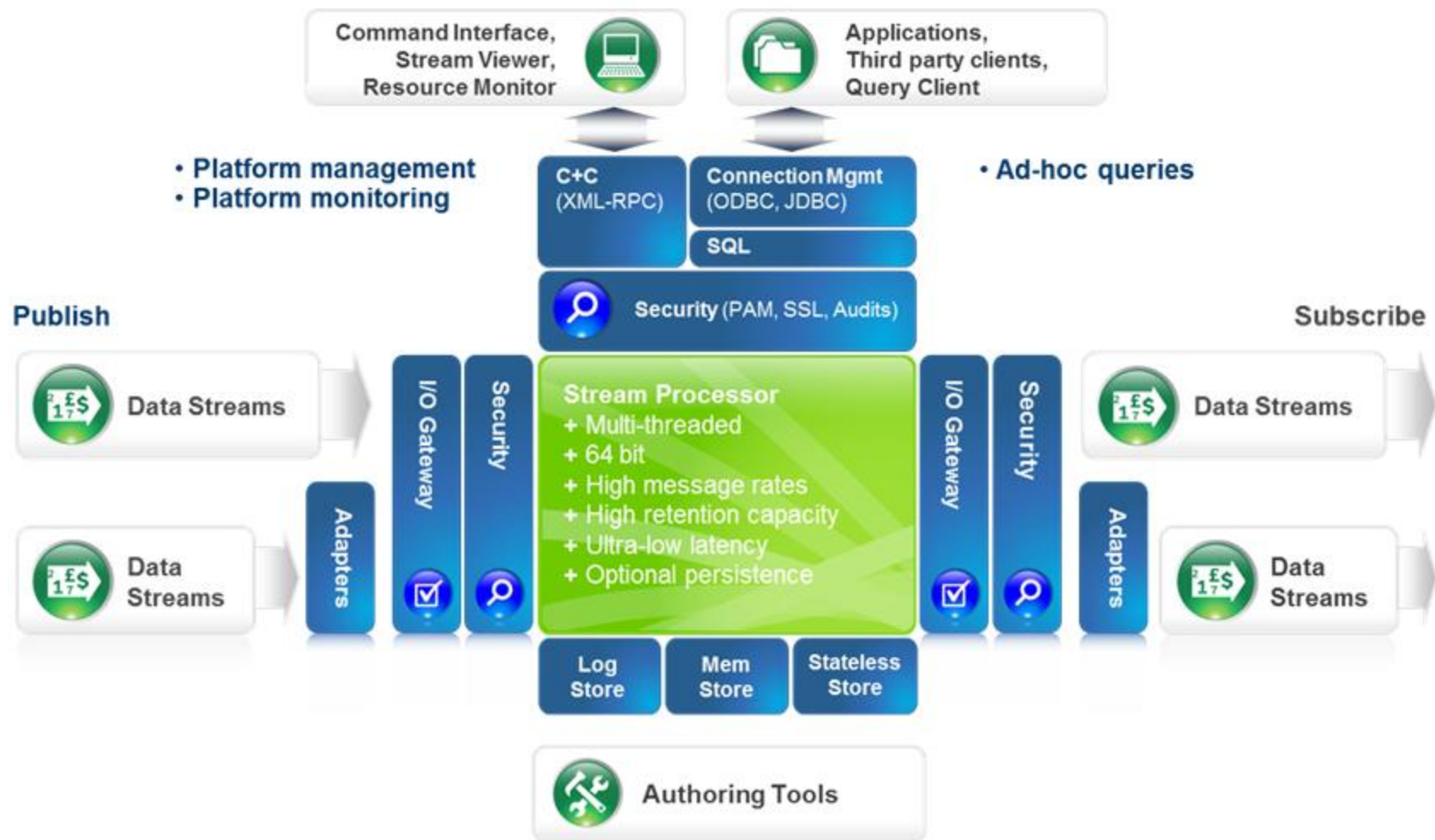
常用的商业流计算产品



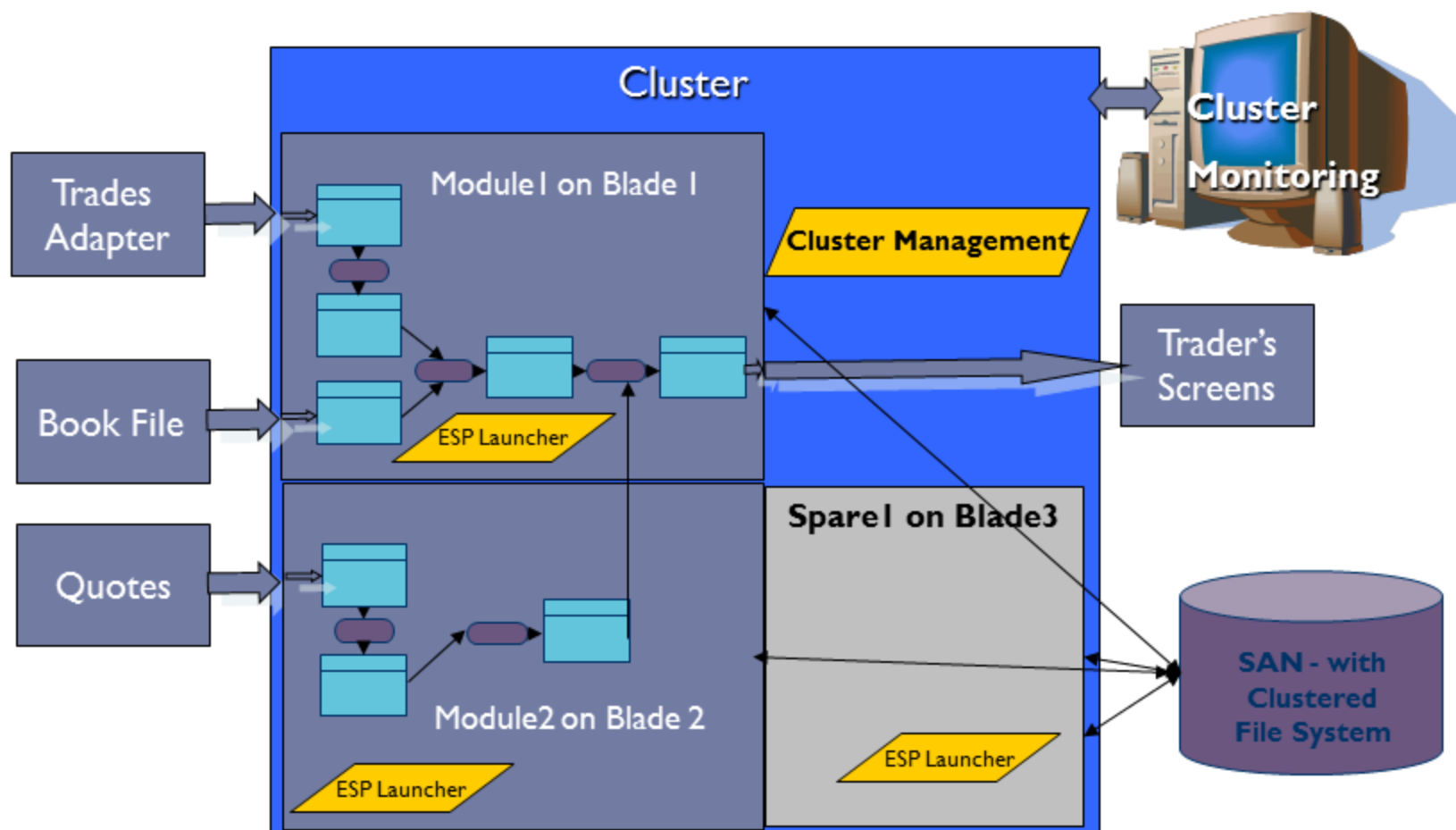
流计算产品架构: Esper



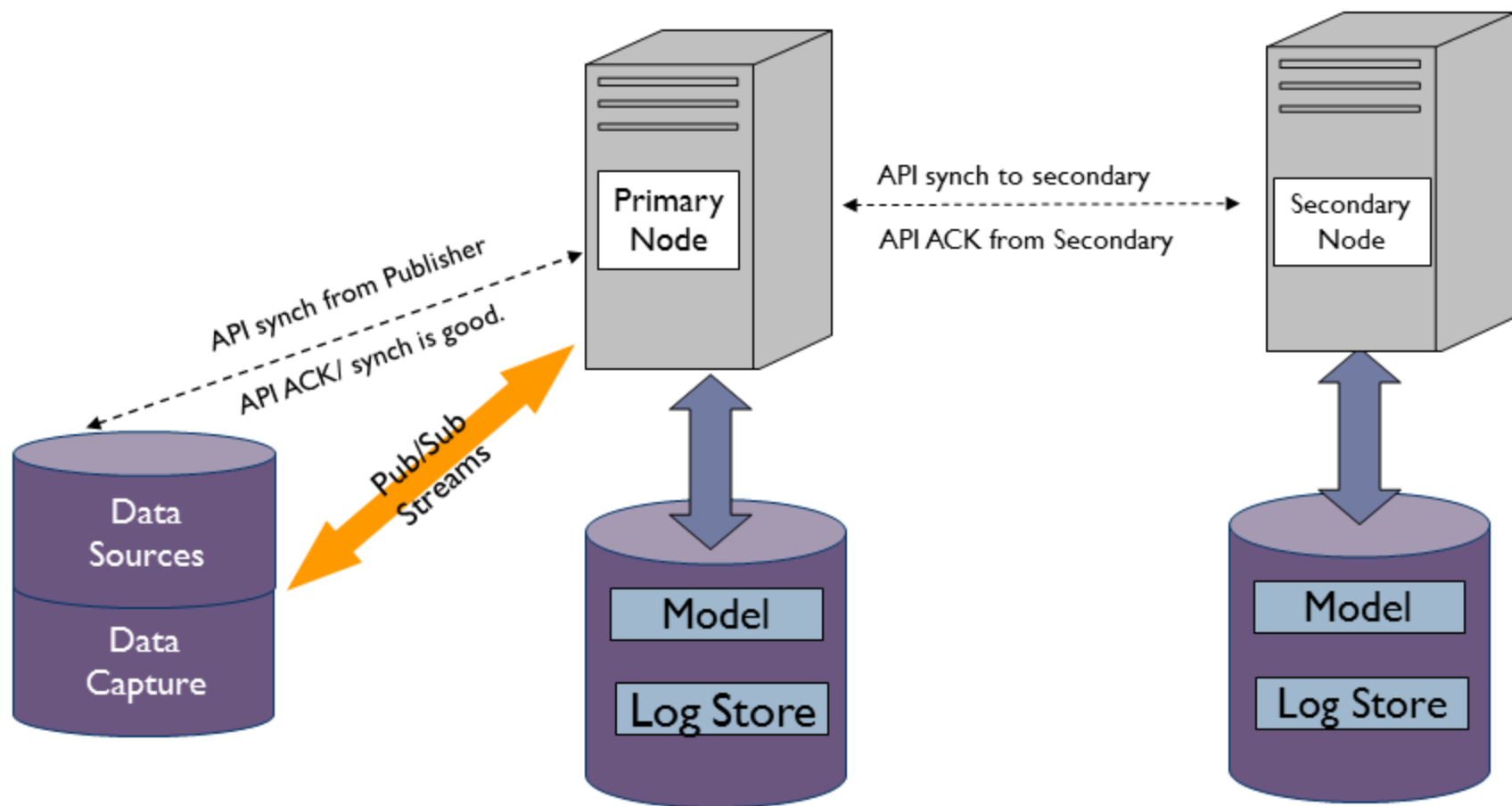
流计算产品架构：ESP



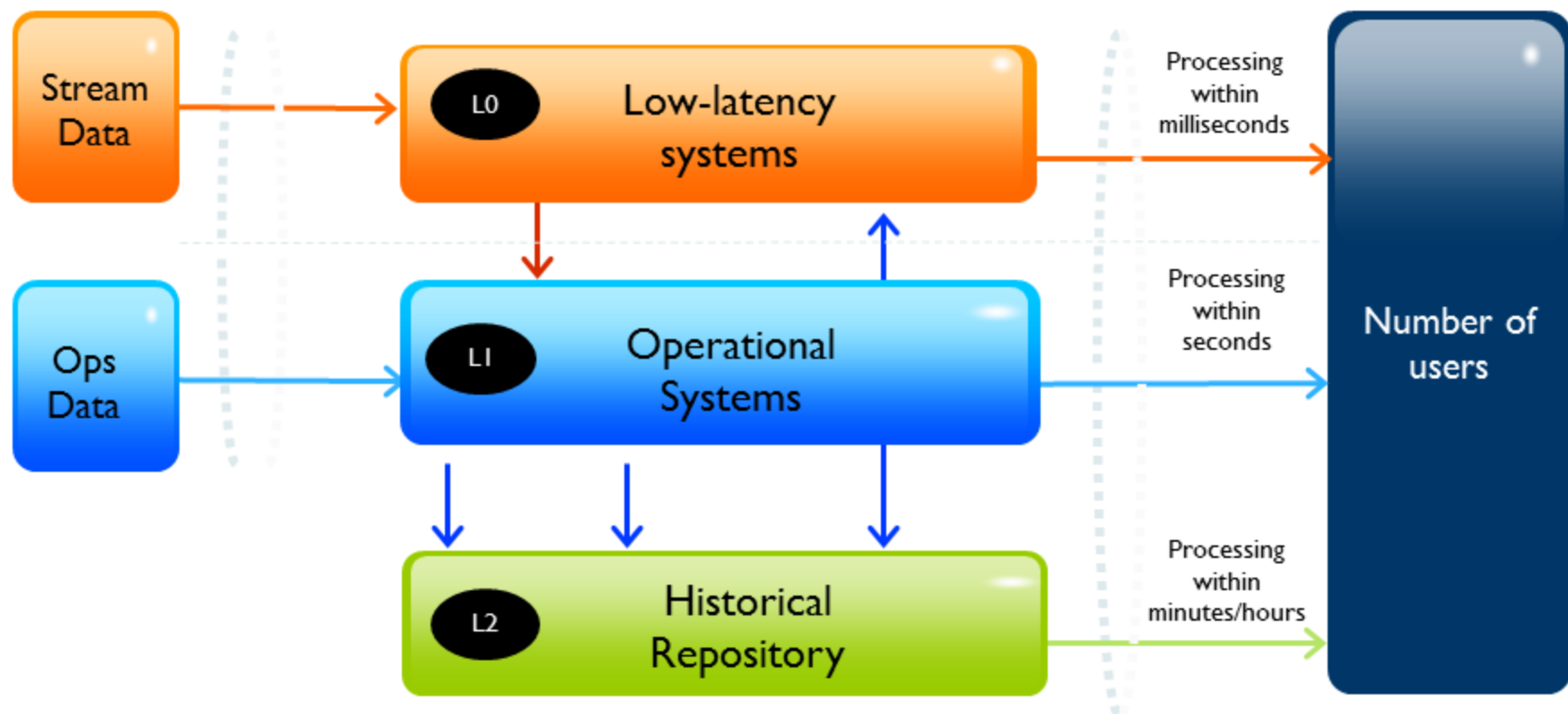
流计算产品架构：集群



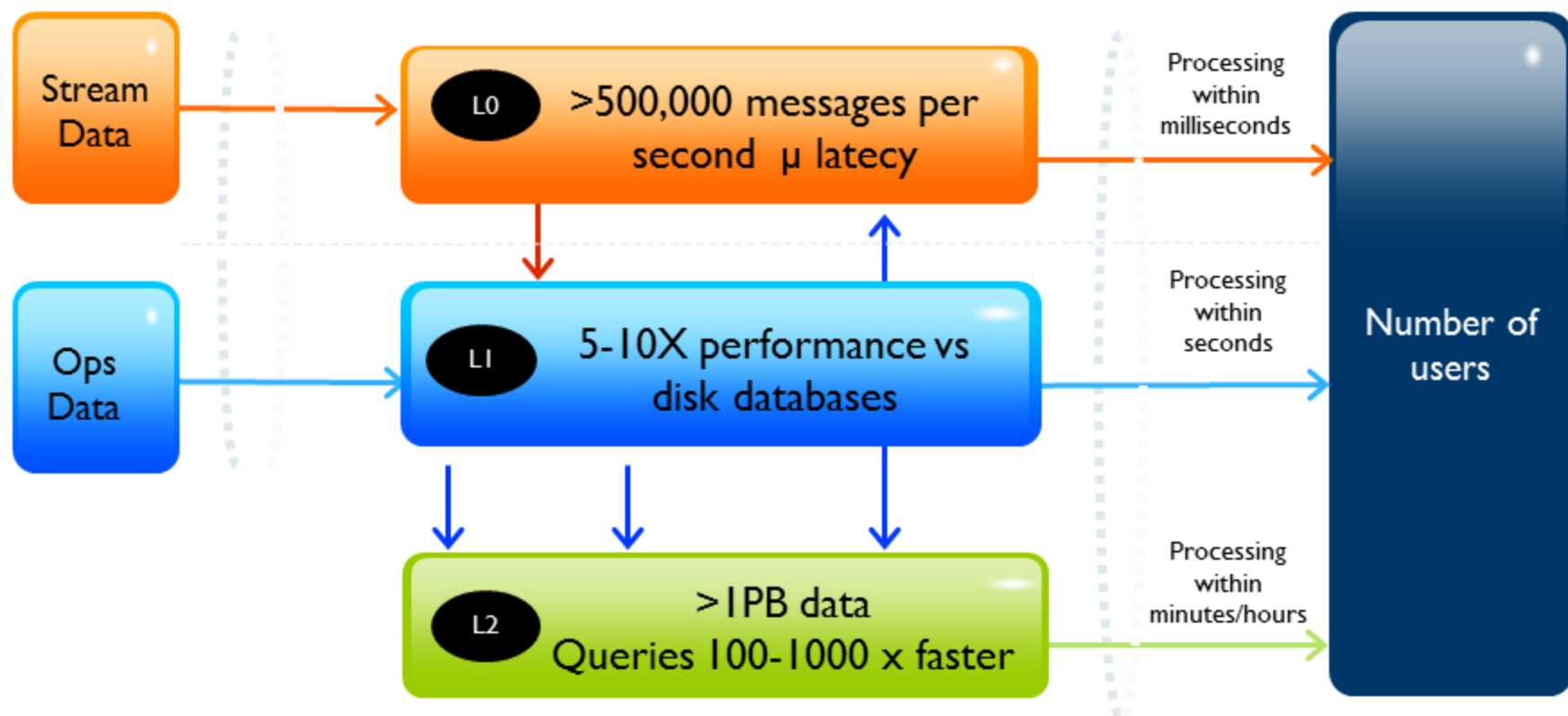
流计算产品架构：热备



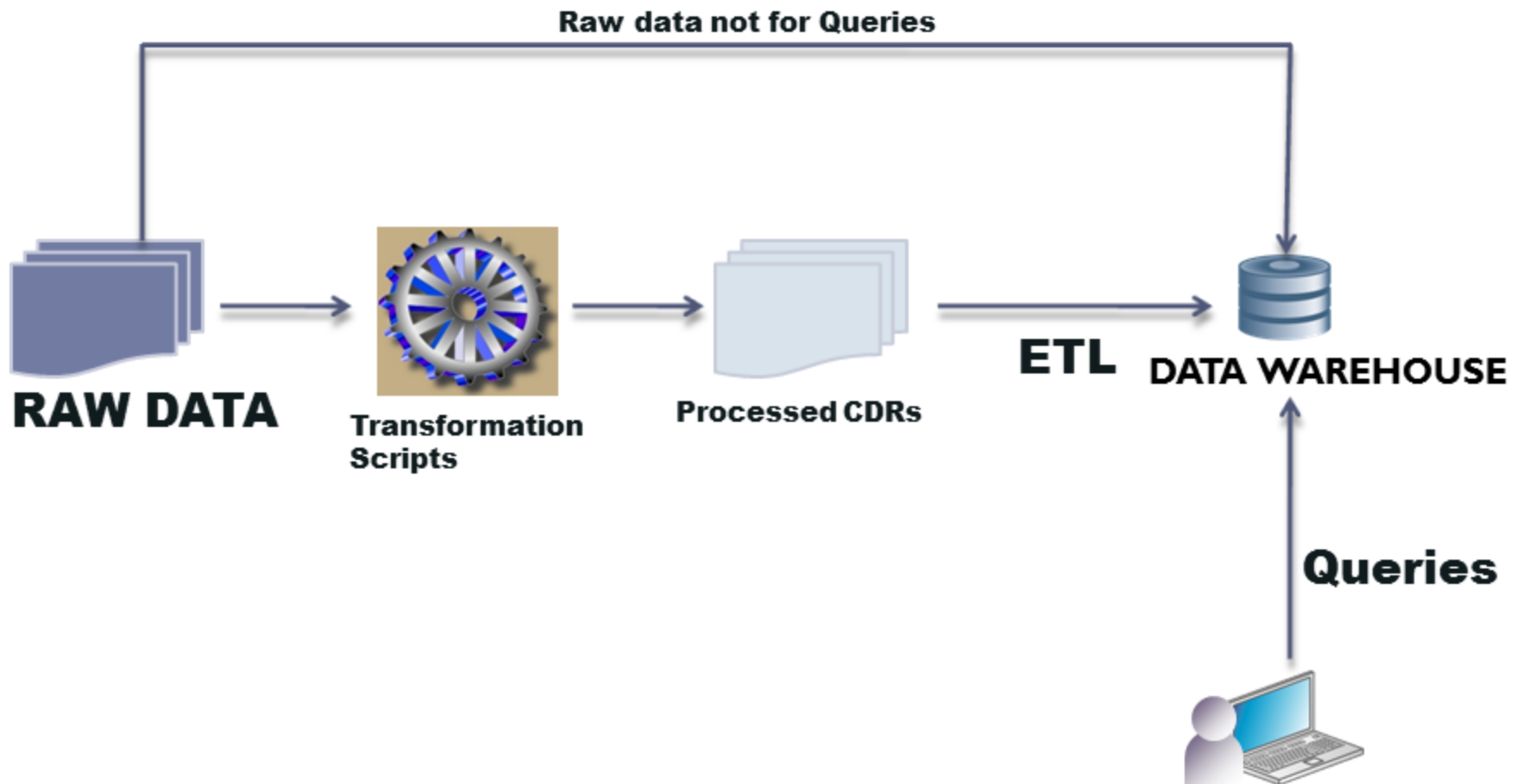
ESP 与 Database 的结合



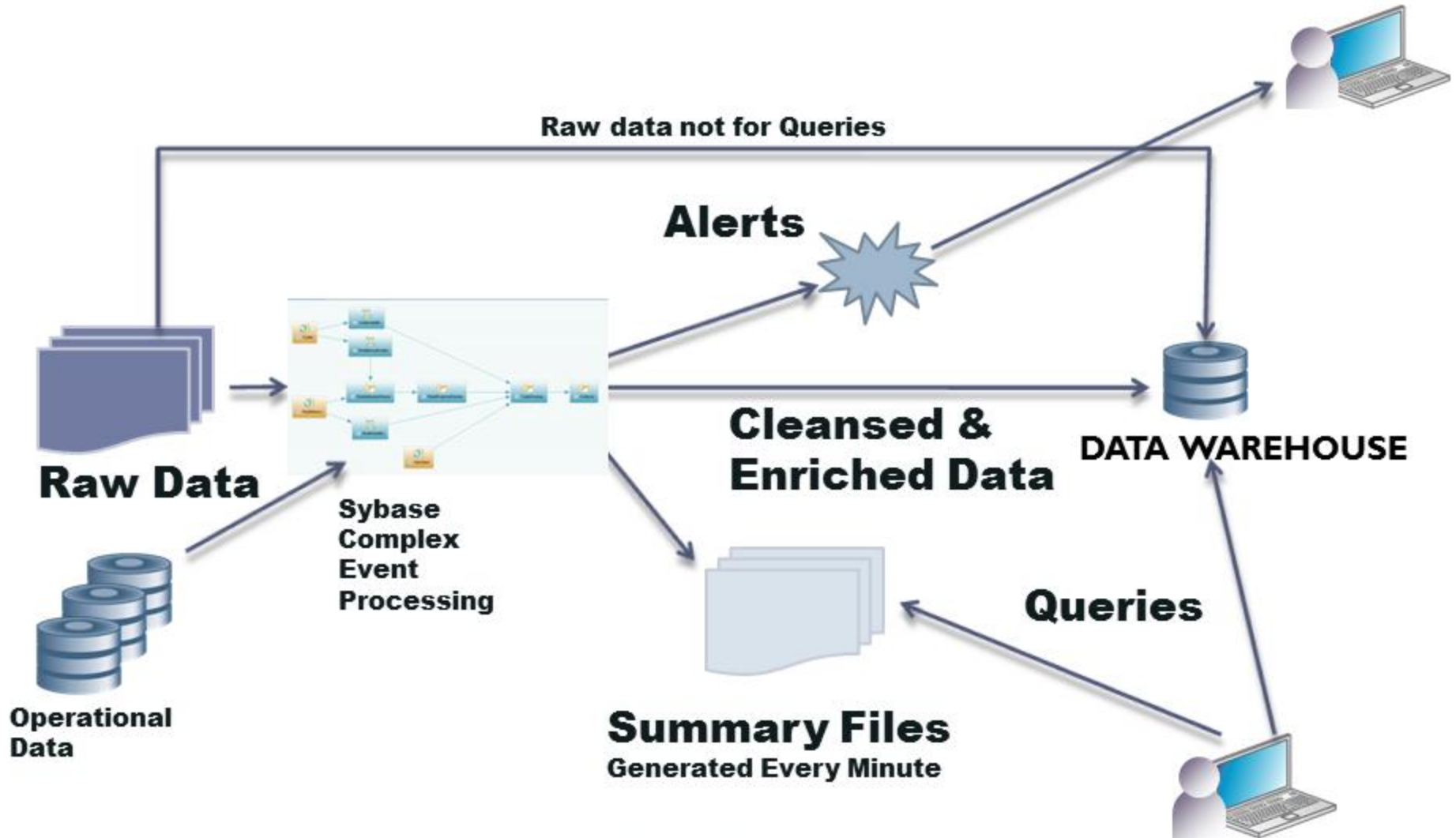
ESP 与 Database 的结合



Legacy Architecture



New Architecture



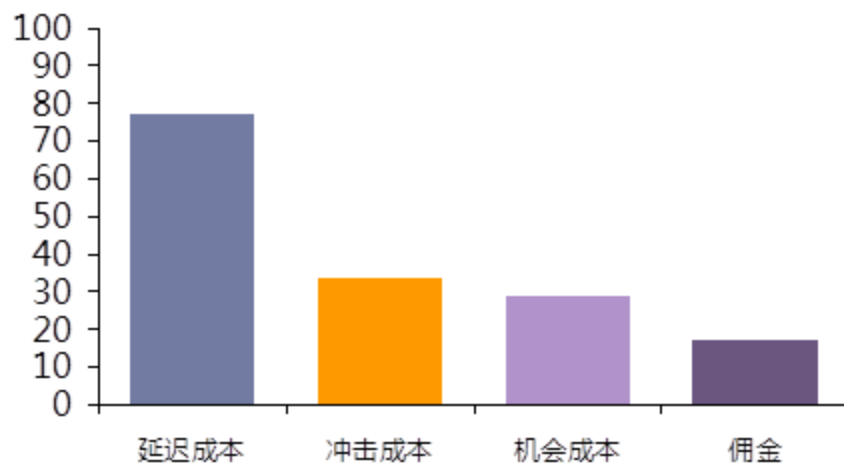
算法交易案例

▶ 算法交易系统定义：

- ▶ 用计算机来确定定订单最佳的执行路径、执行时间、执行价格以及执行的数量一种交易方法。
- ▶ 可用于多项交易品种：股票、期货、股指期货、权证、债券、ETF、外汇交易

▶ 算法交易系统价值：

- ▶ 增强投资收益
- ▶ 提高执行效率
- ▶ 订单执行的隐蔽性
- ▶ 减少人力成本
- ▶ 最佳的路由（海外市场）



算法交易案例

▶ CCI的计算方法：

$$CCI(N日) = (TP-MA) \div MD \div 0.015$$

$$TP = (\text{最高价} + \text{最低价} + \text{收盘价}) \div 3$$

$$MA = \text{最近}N\text{日收盘价的累计之和} \div N$$

$$MD = \text{最近}N\text{日}(MA - \text{收盘价})\text{的累计之和} \div N$$

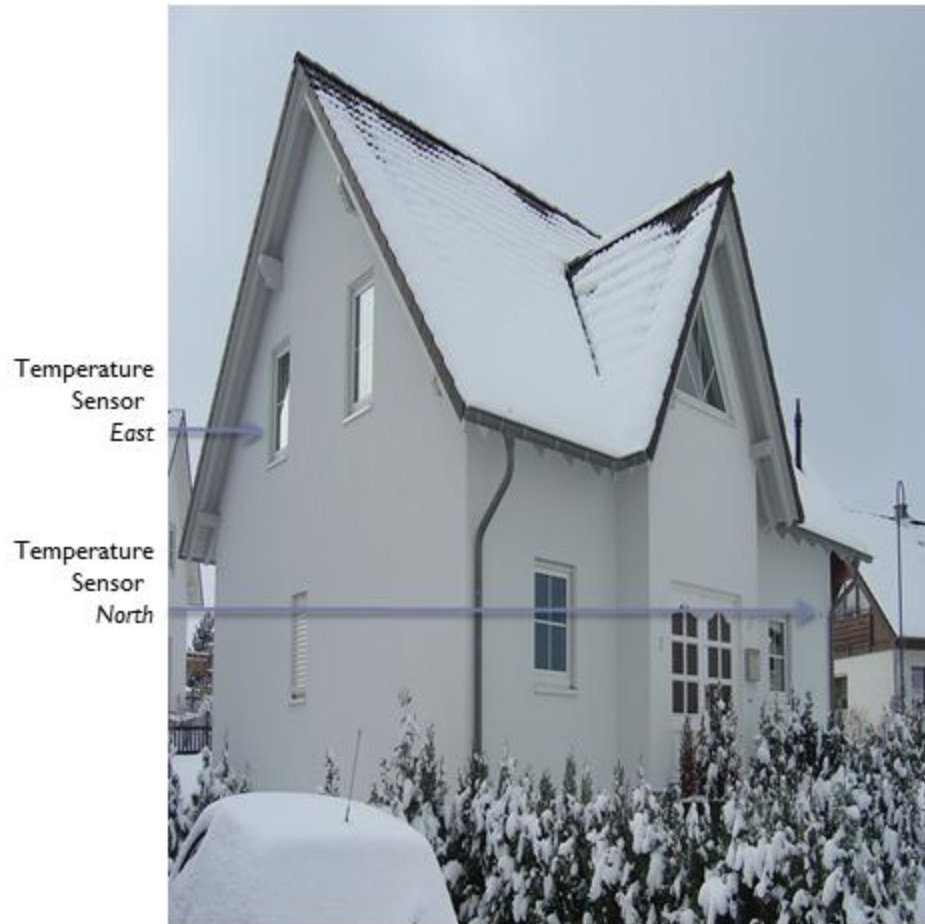
0.015为计算系数，N为计算周期

▶ CCI的判断标准：

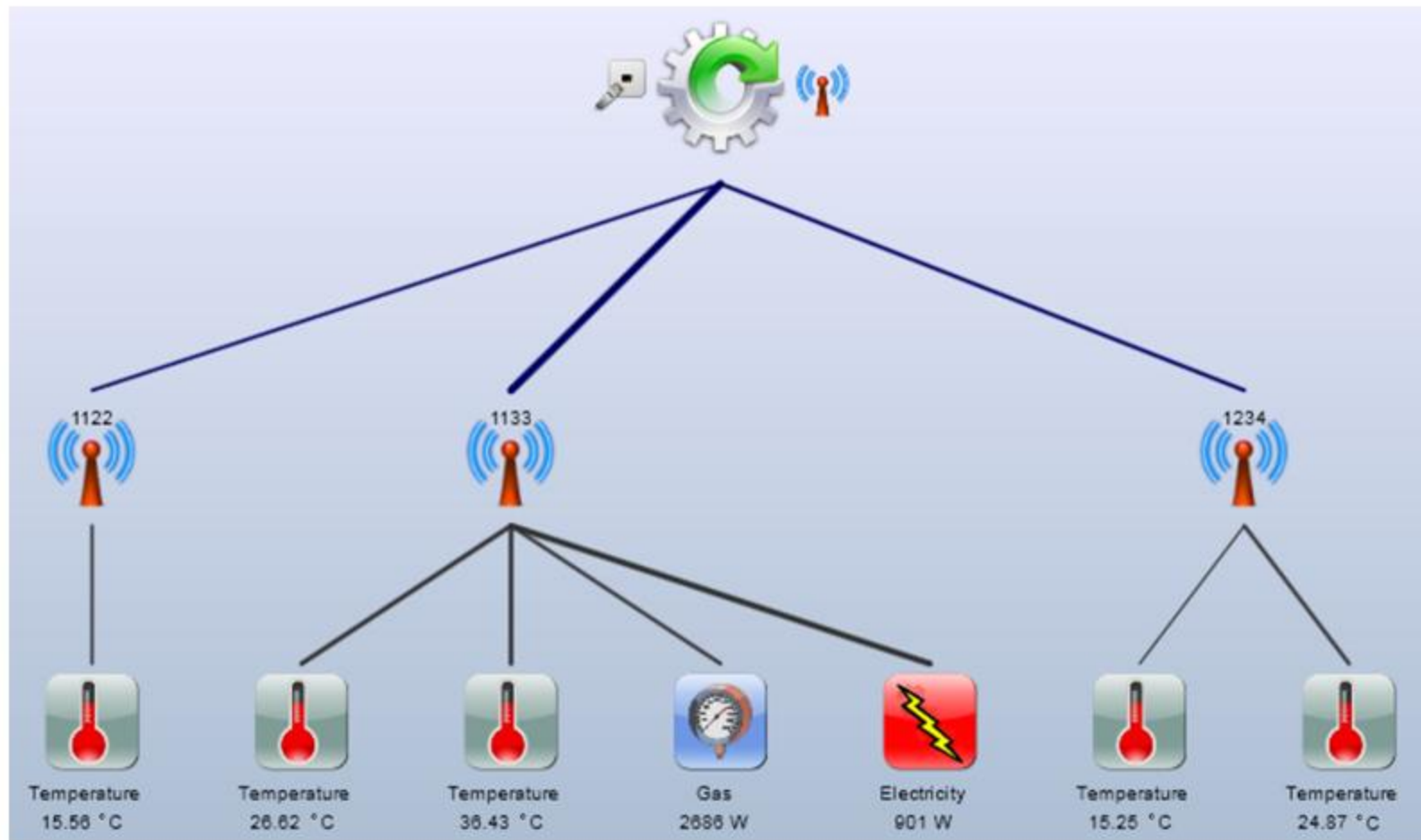
- ▶ 当CCI > +100时，表明股价已经进入超买区间，股价的异动现象应多加关注。
- ▶ 当CCI < -100时，表明股价已经进入超卖区间，投资者可以逢低吸纳股票。
- ▶ 当CCI介于+100—-100之间时，表明股价处于窄幅振荡整理的常态区间，投资者应以观望为主。



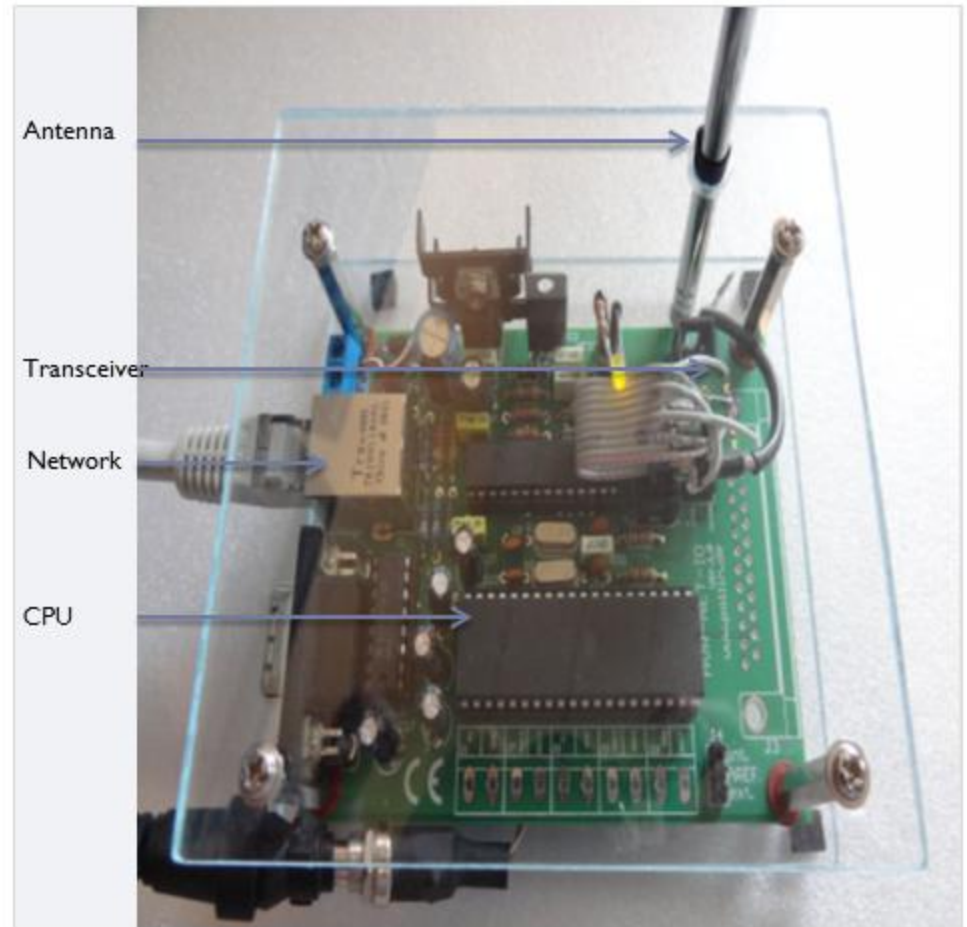
Location of the Sensors



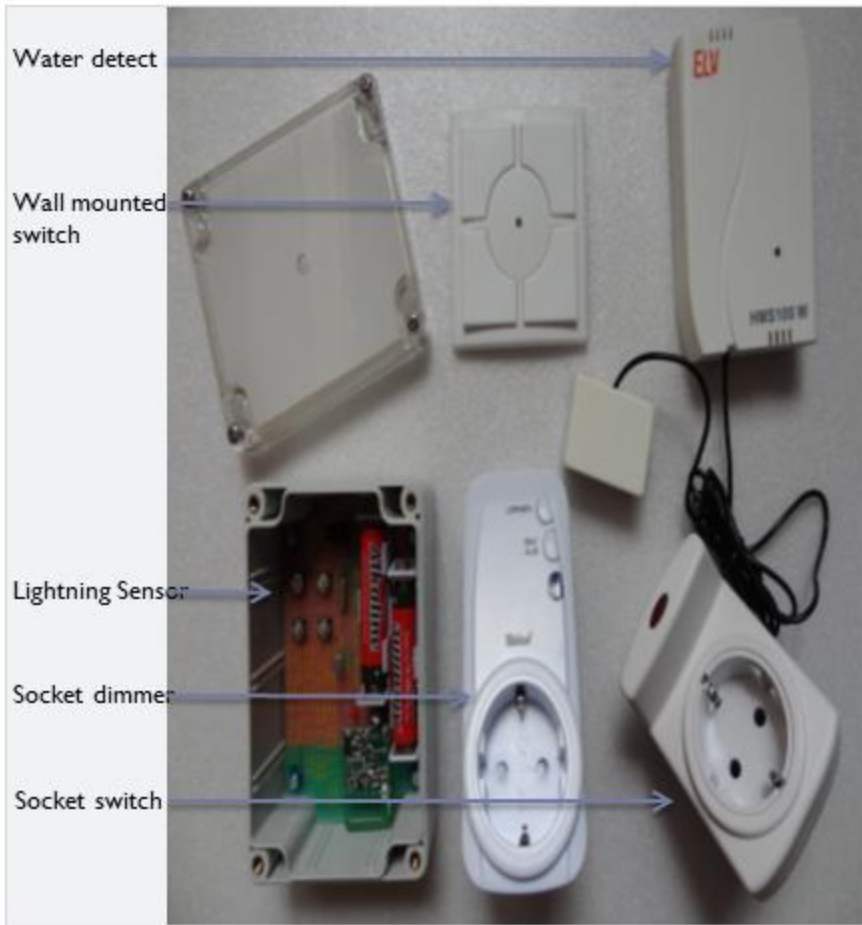
Sensor Network Topology



Base Station

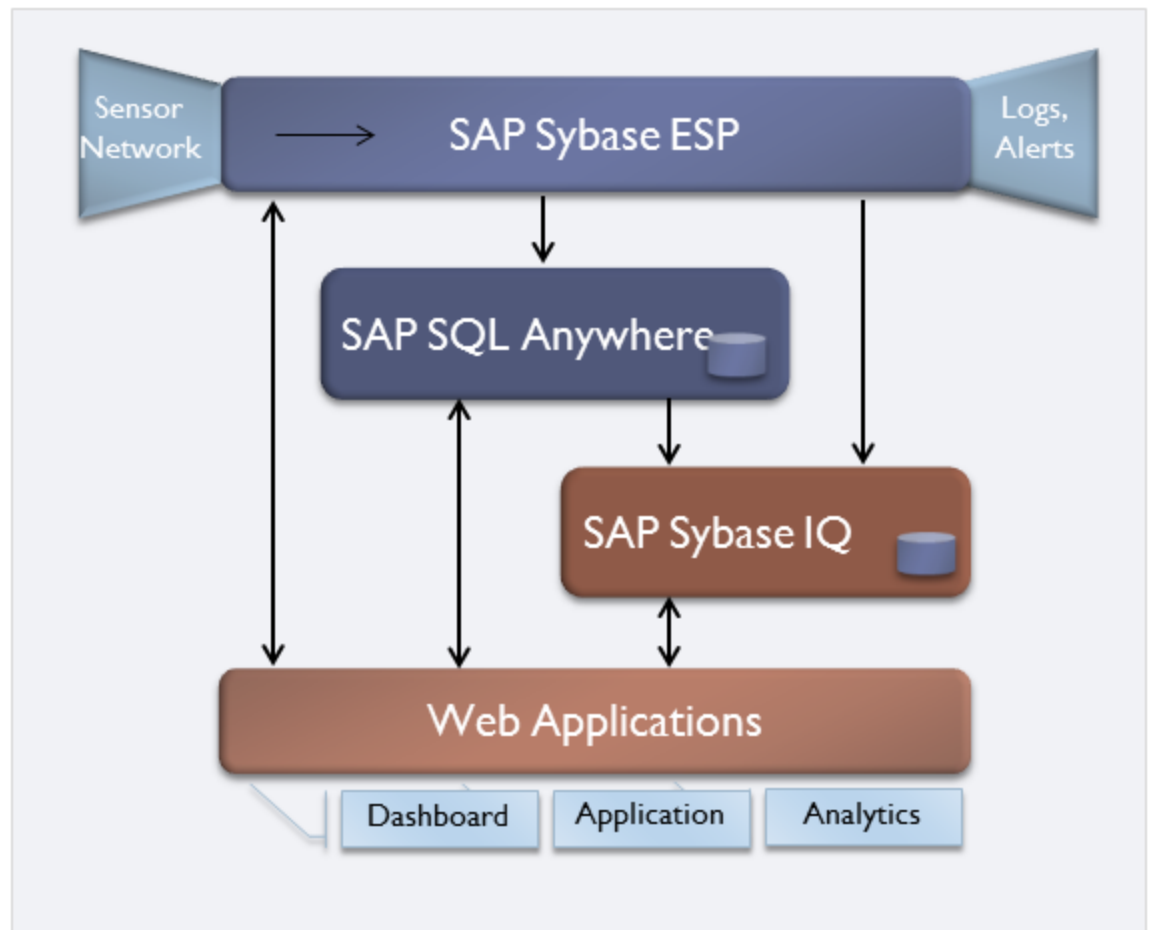


Sensor Board for Gas & Power Consumption



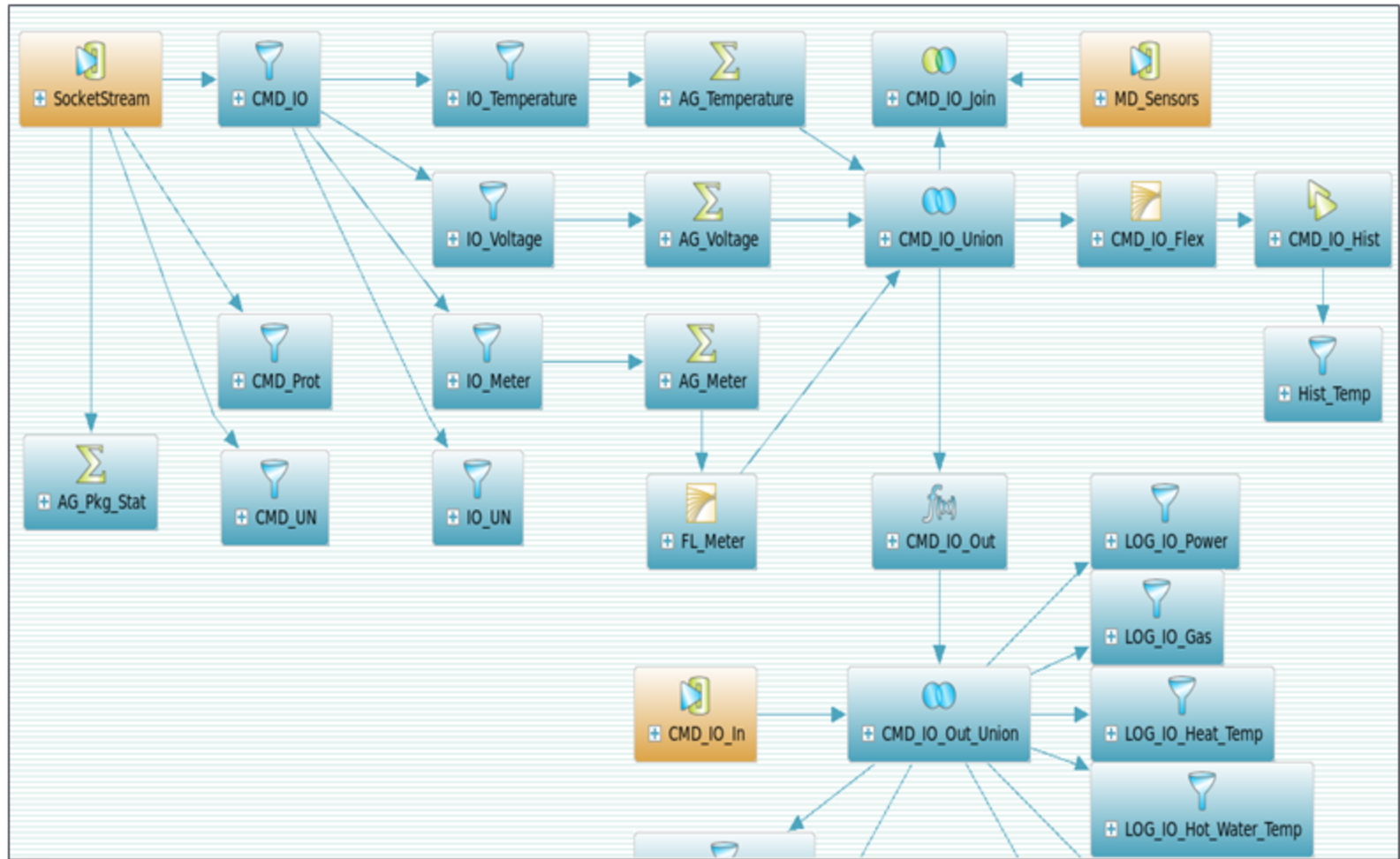
ESP Architecture

- ▶ **ESP Infrastructure**
- ▶ **SAP Sybase ESP**
 - ▶ Does Event Preprocessing, Situational Detection and Event Tracking today
- ▶ **SAP SQL Anywhere**
 - ▶ Holds data for at least two days (48h)
 - ▶ Used as ESP's master data store
- ▶ **SAP Sybase IQ**
 - ▶ Historical store with >2 years of data
- ▶ **Applications**
 - ▶ Various applications: Dashboards, Histograms, Reporting & Analytics, Home Automation



ESP Architecture

Model Detail



Thank You

