Event Processing

Raffaella Grieco

NewTech4Steel WorkShop, Buttrio 13-14 November 2018

Event Processing - Definitions

"Event processing is a method of tracking and analyzing (processing) streams of information (data) about things that happen (events), and deriving a conclusion from them"

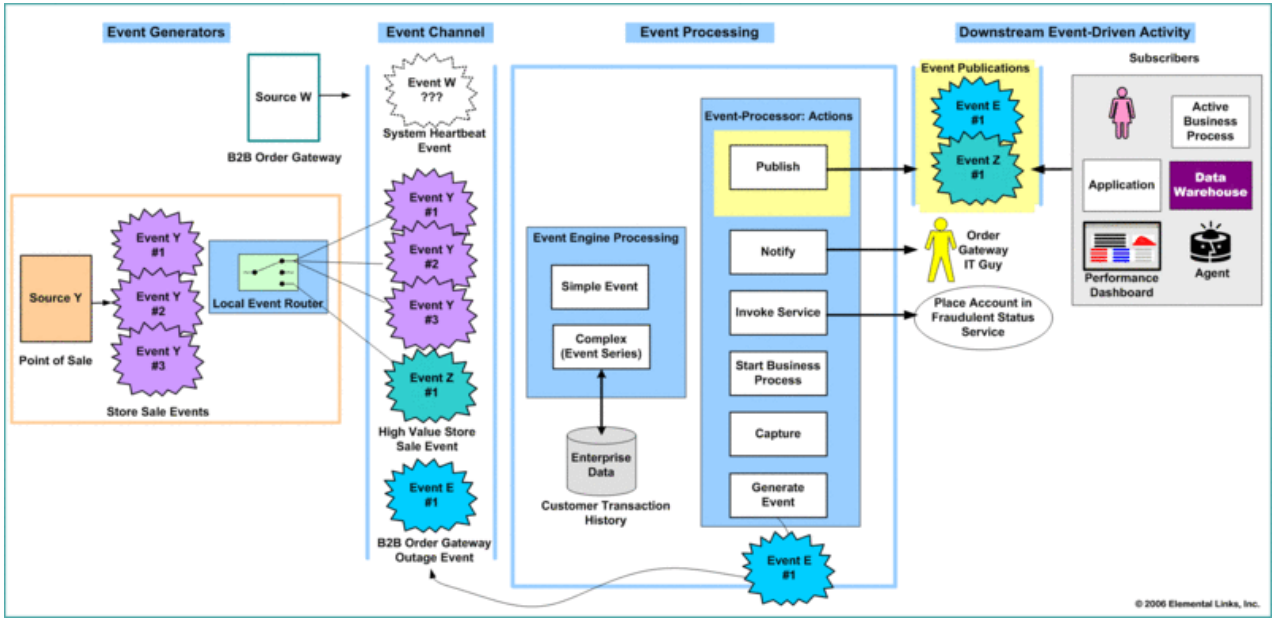
- *"**Simple event processing** concerns events that are directly related to specific, measurable changes of condition. In simple event processing, a notable event happens which initiates downstream action(s). Simple event processing is commonly used to drive the real-time flow of work, thereby reducing lag time and cost"*

- *"**Complex event processing**, or CEP, is event processing that combines data from multiple sources to infer events or patterns that suggest more complicated circumstances. The goal of complex event processing is to identify meaningful events (such as opportunities or threats) and respond to them as quickly as possible"*

- *"**Event stream processing**, or ESP, is a set of technologies designed to assist the construction of event-driven information systems. ESP technologies include event visualization, event databases, event-driven middleware, and event processing languages, or complex event processing (CEP). In practice, the terms ESP and CEP are often used interchangeably"*

CEP – Part of Event Driven Architecture

Event-driven architecture (EDA) is a software architecture pattern promoting the production, detection, consumption of, and reaction to events

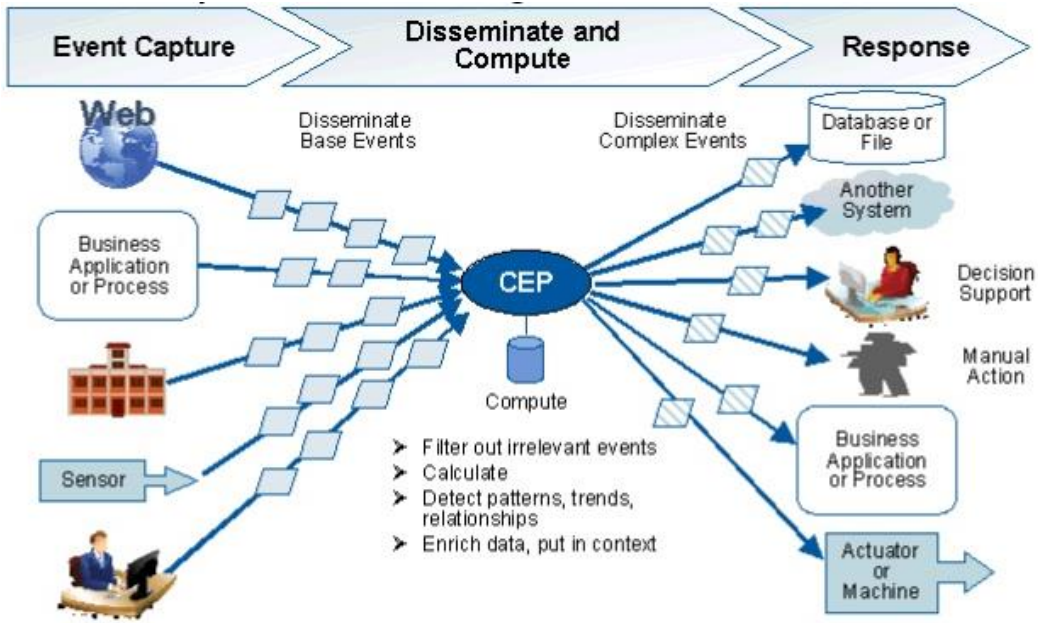


- Event might signify a problem, opportunity, threshold, variance etc.
- Event pushed to all interested parties
- Loose coupling – creator of event no knowledge of consumption

Traditional Application vs EDA

	Database Applications	Event-driven Applications
Query Paradigm	Ad-hoc queries or requests	Continuous standing queries
Latency	Seconds, hours, days	Milliseconds or less
Data Rate	Hundreds of events/sec	Tens of thousands of events/sec or more

CEP Architecture



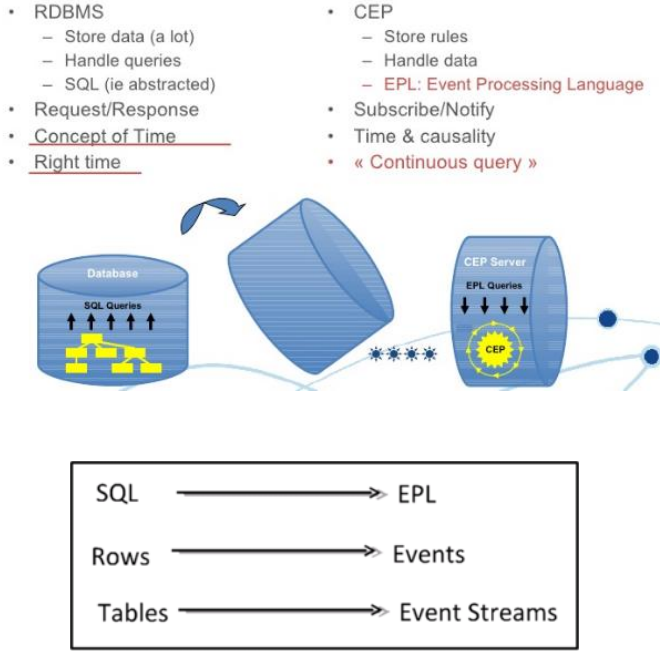
CEP is about complex event detection (meaningful, pattern, relationship and data abstraction) and fire reaction

- Volume&Latency. Efficient (near real-time) processing of large numbers of events
- Scalability
- Availability

CEP System

A CEP system is like your typical database model turned upside down.

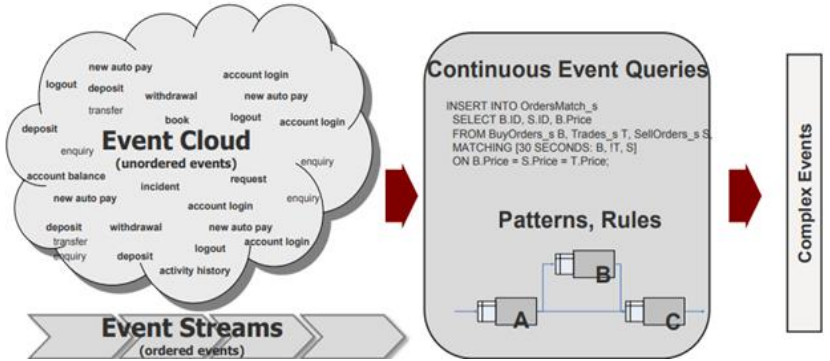
- Typical database stores data, and runs queries against the data
- CEP data stores queries, and runs data through the queries.
- Basically needs:
  - Data – in the form of ‘Events’
  - Queries – using EPL (‘Event Processing Language’)
  - Listeners – code that ‘does something’ if the queries return results



CEP module

The Complex Event Processor module can be broken down into the following functional components:

- event representation
- processing model
- language specification (EPL)



CEP module

Event representation

In CEP an “event” is an object that is a record of an activity in a system.

It has three features:

- **Form**: is an object with attributes and data components. Can be a simple strings or a series of data items
- **Significance**: it is an activity
- **Relativity**: an activity is related to other activities by time, causality and aggregation

CEP module

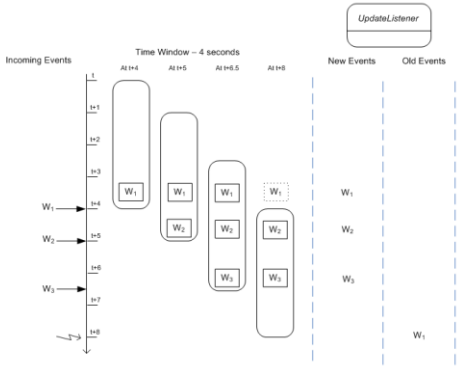
The three most common and relevant relationship between events:

- **Time**: a relationship that orders events (event A happened before B)
- **Cause**: a dependence relationship between activities in a system (if the activity represented by event A had to happen in order for the activity represented by event B, then A caused B)
- **Aggregation**: an abstraction relationship (if event A signifies an activity that consists of the activities of a set of events B1, B2, B3 then A is an aggregation of all the events in B

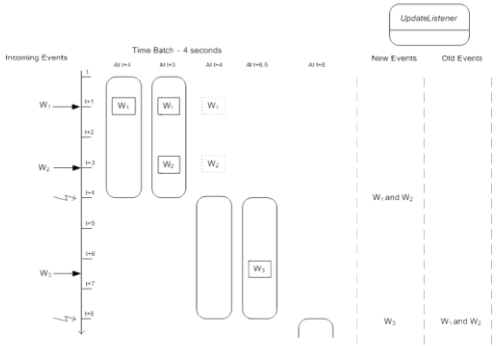
CEP module

Processing model

- Continuous, results are output as soon as incoming events are received
- Incoming events may be processed through either sliding or batched windows (Row based/Time Based)



Sliding Window (Time Based)

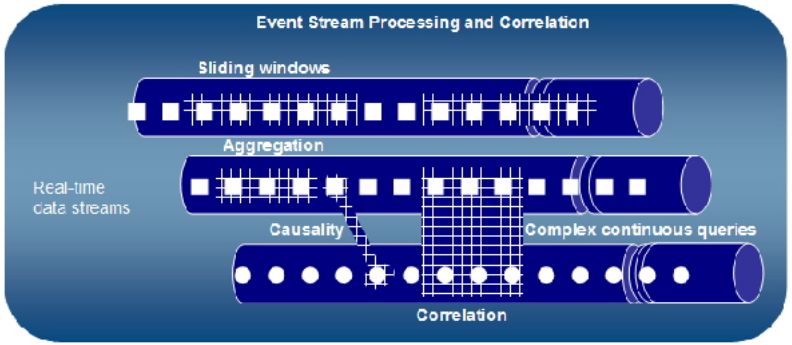


Batched Window (Time Based)

CEP module

Language specification (Event Processing Language - EPL)

- SQL-like language
- To express filtering, aggregation and join also over sliding windows of multiple event streams



Source: EsperTech

Notation:
C = Set of all event
V = value
Xi, Yi = Event with order number i
Xi(a, b, …) = a, b, … are attributes of event Xi
Xi(where a = Yi.a) = attribute a is matched with attribute a from event Yi
T = time interval expressed in seconds, minutes, hours, days, weeks, months or years
Z = expression that is built with elements from the general CEP language

Operators:
Operators are divided into three classes:
- Logical Operator: “and”, “or” and “not”
- Time operator: “within T(Z)”
- Sequence operator: “->”

Example expressions:
“X and Y” within T(40 seconds)
“A -> B” (event B has to arrive after A)

CEP module

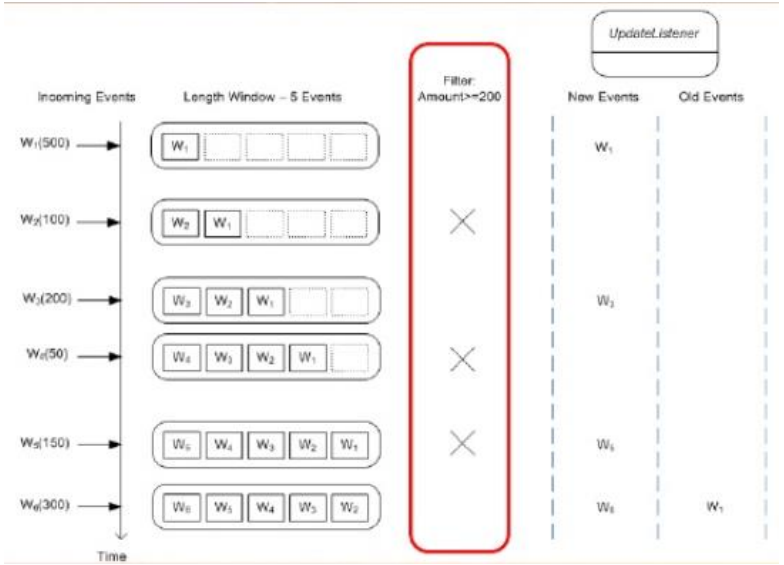
Query examples on Sliding Window Row Based

Select * from Withdrawal(amount >= 200).win: length(5)

- Event are filtered into the sliding window

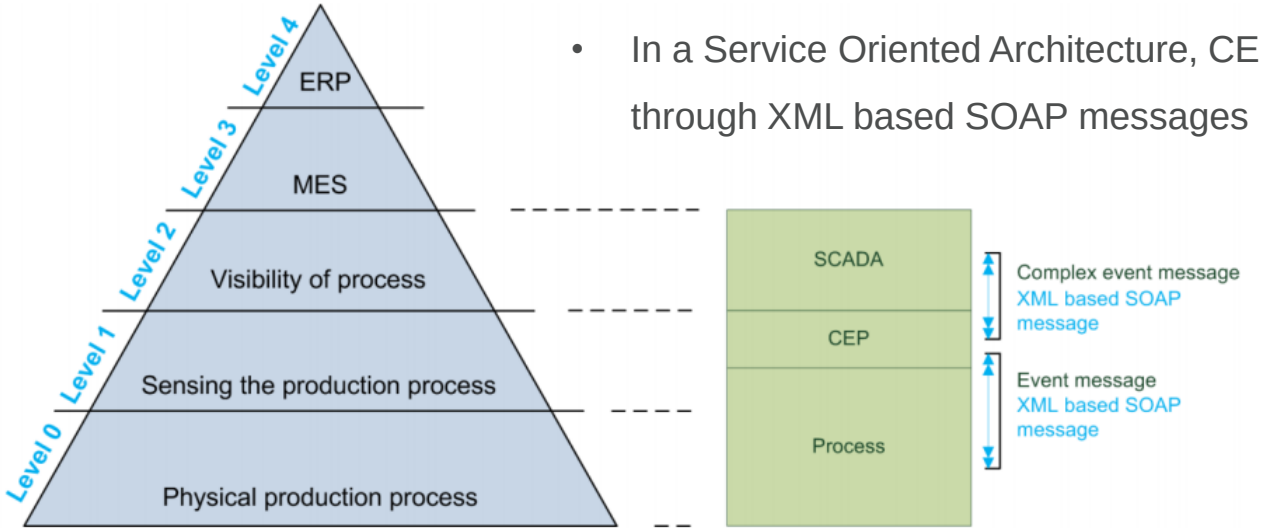
Select * from Withdrawal.win: length(5) where amount >= 200

- Event passed onto the listener are filtered



CEP for Manufacturing

- Event Producers is the physical production process (Level 0)
- Event Consumers are typically Level 2 System
- In a Service Oriented Architecture, CEP module can communicate through XML based SOAP messages



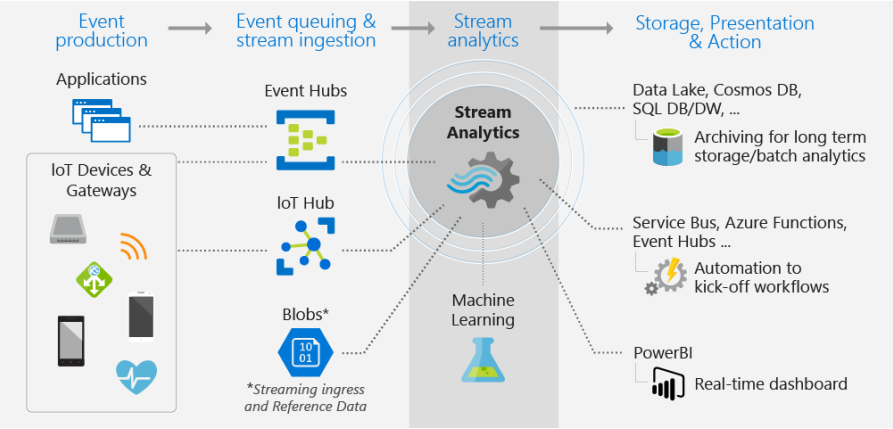
CEP Products

- Apama
  - CEP/ESP/IoT, Java/C++, Cross-platform, AG Software
- Drools
  - Rule Engine, Java, Cross-platform, Red Hat
- Esper
  - CEP/ESP, Java/C#, Cross-platform, EsperTech Inc.
- Sybase ESP
- Oracle Event Processing
- IBM WebSphere Business Events
- SqlServer StreamInsight
- TIBCO BusinessEvents & Streambase

Solution for CEP

The major public cloud vendors provide CEP as service

- CEP platforms require large memory and computational power which are abundant in the cloud.
- Lower cost than on premise implementations



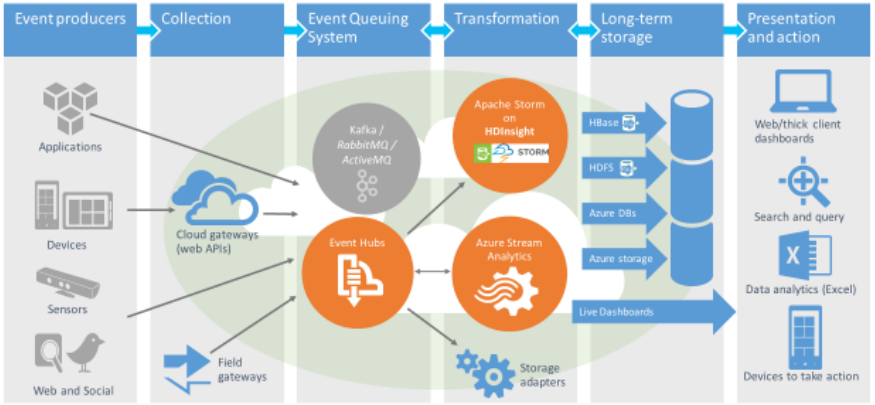
Azure **Stream Analytics** is a CEP Platform as a Service (**PaaS**) in Azure

- Azure Event Hubs takes a few clicks to connect multiple sources and sinks to create an end-to-end pipeline.
- Simple SQL-based query language with its powerful temporal constraints to analyze data in motion
- Built-in support for commonly used scenarios such as anomaly detection

IoT & CEP

Complex event processing is a key enabler in Internet of Things (IoT) settings and Smart Cyber-physical systems (CPS) as well.

- Processing dense and heterogeneous streams from various sensors and matching patterns against those streams is a typical task in such cases.



Via Corsica, 12
16128 Genova - Italy
P. +39 010 53851 | info@rina.org
rina.org