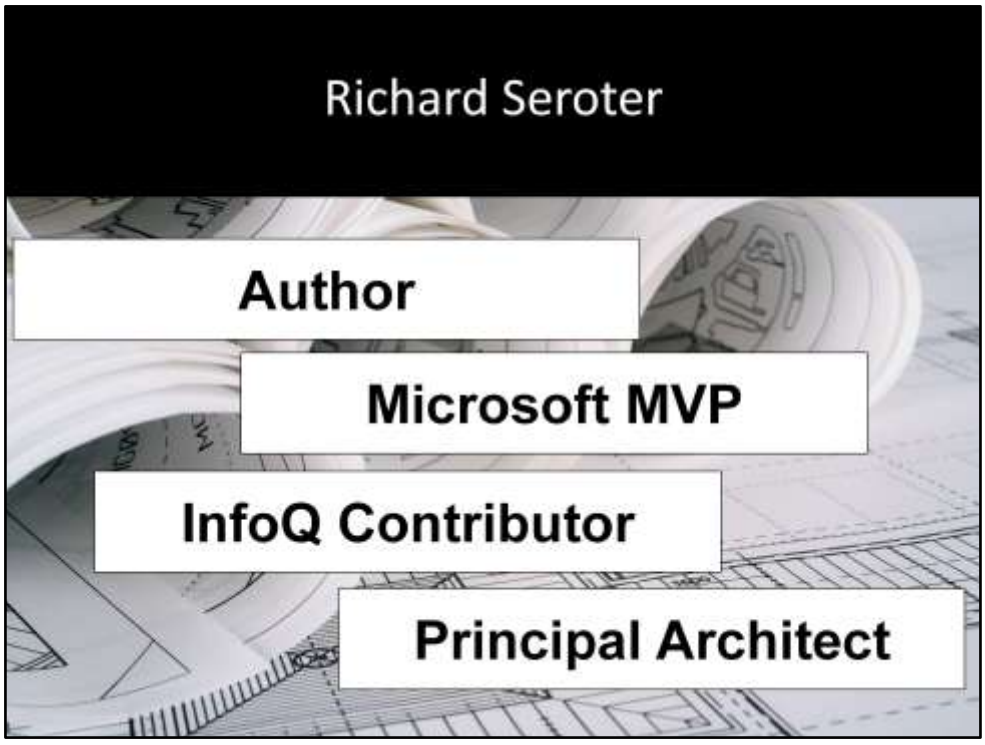
Enabling a Real Time Enterprise Through Event Driven Architecture

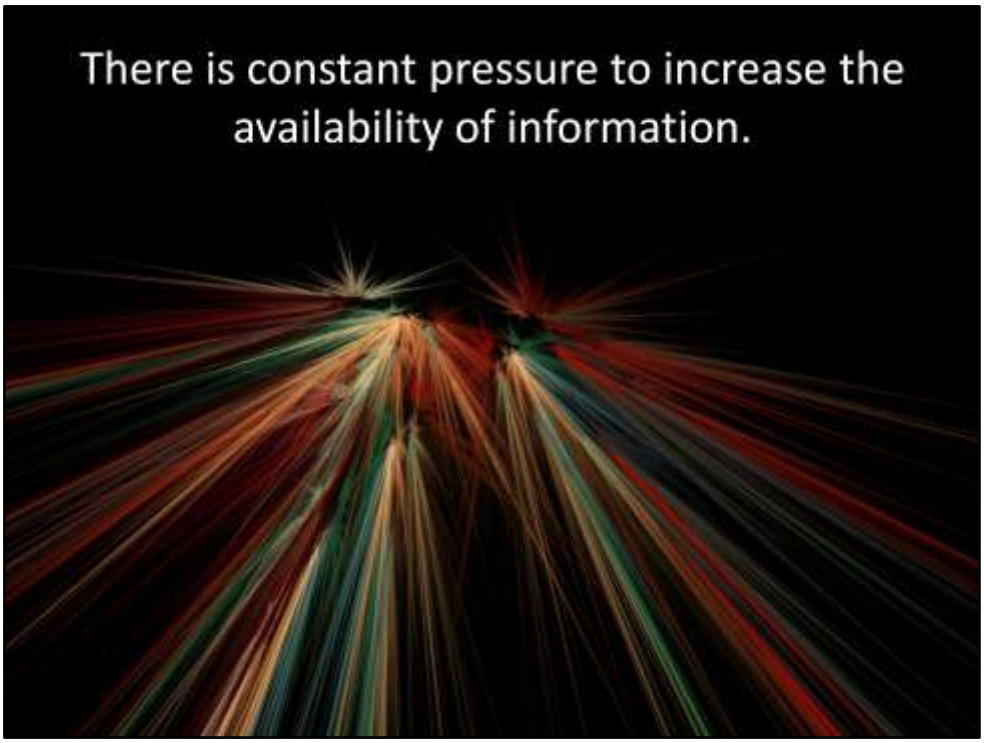
Richard Seroter

Principal Architect, Amgen

Twitter: @rseroter

Blog: seroter.wordpress.com


Richard Seroter
Author
Microsoft MVP
InfoQ Contributor
Principal Architect



[Problem]

- Lots more data than ever before
  - Twitter: 90 million tweets per day, 8 terabytes per day
  - On a cross country flight, the sensors in a Boeing 737 generate about 240 terabytes of data
  - RFID has the potential to generate 100 to 1000 times the data volume of conventional bar code technology.
  - **In 2010, more than 1200 exabytes of digital info was created. Single Exabyte is equal to about 1 trillion books.**
    - **data created by individuals, but lots from increasing # of sensors as well**
  - Gartner says enterprise data in all forms will grow 650% over the next five years with IDC claiming entire world's data doubles every 18 months
  - 30% of biz leaders say they can't get the data they need at the speed they need it, and 61% want faster access
    - 1 in 3 want MORE data
- Too much data overwhelming our computing capability and new model needed
- How do we know more, faster?; Continuous intelligence

[Where we want to be]

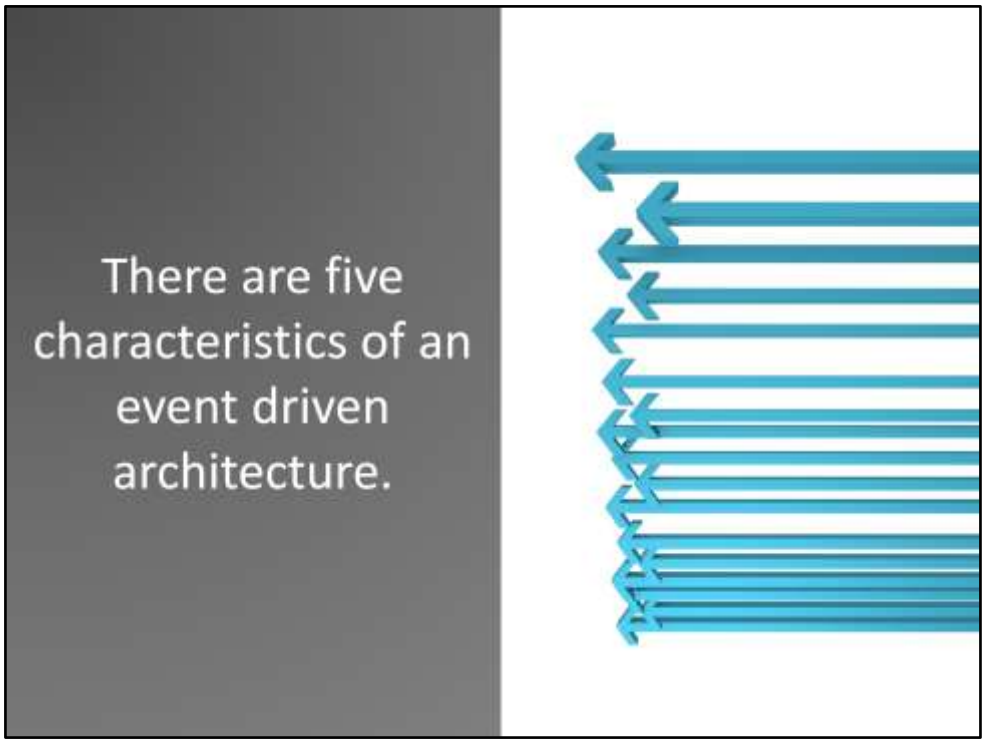
- EDA between systems, in analytics, etc
  - As opposed to ETL, polling-based solutions, point-to-point
  - Data in motion vs. data at rest
- Use even in IS-centric solutions (e.g. monitoring)
  - Monitor SLAs in real time and exceptions from the norm
- Well designed EDA application is an attention amplifier
- ** a critical and increasingly scare resource is interrupted time **
- Getting just a little bit of the right info just ahead of when it's needed is a lot more valuable than all the info in the world a month or day later
- New competitive advantage will be an ability to anticipate events based on info about what's happening right now
- Sometimes you cannot act quickly, so MUST be predictive (think making a product with long lead time)

[Key Point 1]

- Improves reaction time
- reduces biz process duration
- improves availability of info (data consistency, situational awareness)
- Value of event processing lies in recognizing the significance of an event from a business context, and identifying the right responses to associate with that event

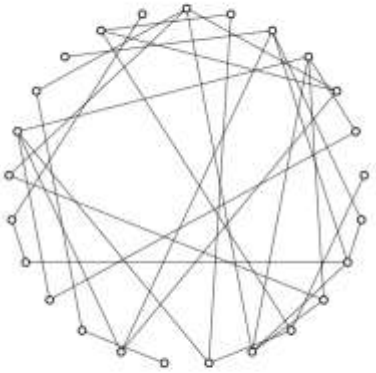
- **event driven good for unpredictable factors, situations and timing**;
- time driven behavior occurs when nature and timing of event can be planned in advance while request-driven is appropriate when nature of activity is understood and agreed to, but timing is not predictable
- Use EDA for apps that must respond quickly to situations that change rapidly and async and where interactions do NOT need to be transactional

- Event can be receiving customer order, making bank payment, changing customer address, hiring an employee, detecting attempted fraud, change in competitor price, detecting non-compliance, duration of business process
- What is an event? an event is a change in the state of a component of an enterprise or its environment
  - State change: RFID tag
  - Happening: war, buy a house (typically a complex event)
  - Detectable condition: GPS message of truck at standstill
- ** absence of messages conveys information in event driven interaction**
- Now, every EVENT (not just transactions) can become a bit of digital information
  - Log into site, abandon cart
  - Databases can't process non-events. If something DIDN'T happen, the data doesn't get created



- Continuous processing systems have benefit of distributing work more evenly to eliminate bottlenecks, peaks and troughs
- EDA is an arch style in which one or more of the components in a software system are event driven and minimally coupled (one way transfer of event objects)
- EDA if (first three timeliness, last two distribution):
  - report current events as happen (could be bundled for efficiency)
  - pushes notifications (event producer decides)
  - responds immediately
  - one way (fire and forget)
  - free of commands (a report, not a command)

Complex Event Processing (CEP) is the continuous and incremental processing of event streams from multiple sources based on declarative query and pattern specifications with near-zero latency.



- Cases in which a single event will not trigger an action performed by a consumer, but instead the action is triggered by a complex composition of events happening at different times and in different contexts
- CEP is a way of distilling the information value from a number of simple business events into a few more useful, summary level "complex" events **
- Term "complex" may be off-putting, but in reality, complex events actually simplify data by summarizing and abstracting what is happening
    - Sift through countless individual events (e.g. calls) or provides sums/averages
- Manual CEP is often time driven (when do you acquire data) and relies on low data volumes, available people to do analysis and slow forming decisions
- Diff from traditional database driven applications
    - Query is more or less permanent, data is ephemeral,

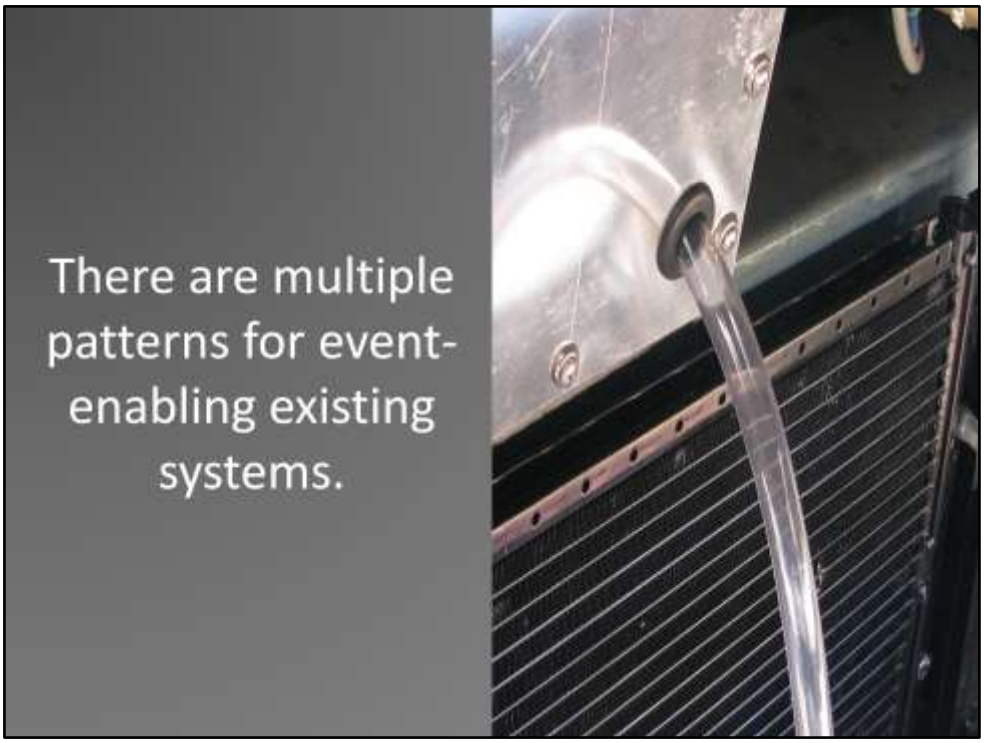
continuously arriving and then disappearing as it becomes out of date

- Rather than an application repeatedly compiling a query, submitting it to a database and waiting for a result, applications using CEP submit a query once.
- Databases have major handicaps. Inherently focused on the past. Analyze what has already happened, not predict what's about to happen.

- CEP is useful when characterization of anomalies are complex and when detection requires analysis gathered from multiple agents across time
- Want tech that can constantly watch events, run through predictive model built on memories of past patterns, and continuously look for the next move

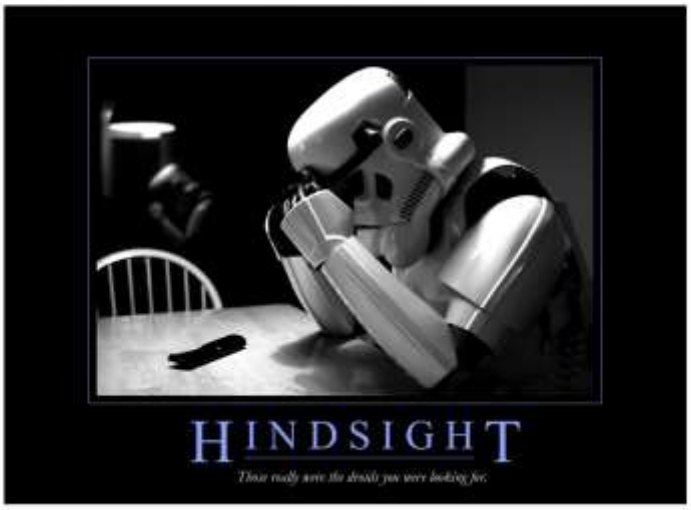
The middleware often embodies the event cloud
Event sources: middleware, databases, devices, agents, business processes

- Events are just data, not operations too
- Nothing special about data in event; unique nature of event processing derives from WAY in which data is exchanged and processed (5 principles)
- What defines an event; what goes in the body
  - Tag indicating type of event, unique ID, creation timestamp, start/end timestamps, producer, priority
  - Before/after (idempotence)
  - May support a “call back” for the receiver to go get the changed data
- Events may have a duration component (life)
- ** Event is not just a data record with a timestamp **
  - Static view of events used primarily for archiving and post event processing like BI
  - Data DERIVED from events
  - Data really the information collected from prior events (customer data acquired by customer regs, etc)



- Most systems don't natively emit events since there was no req for it
- May use …
  - Service bus
  - DBMS adapter to detect changes
  - request driven (scrape and compute) – Competitor or gov't website example

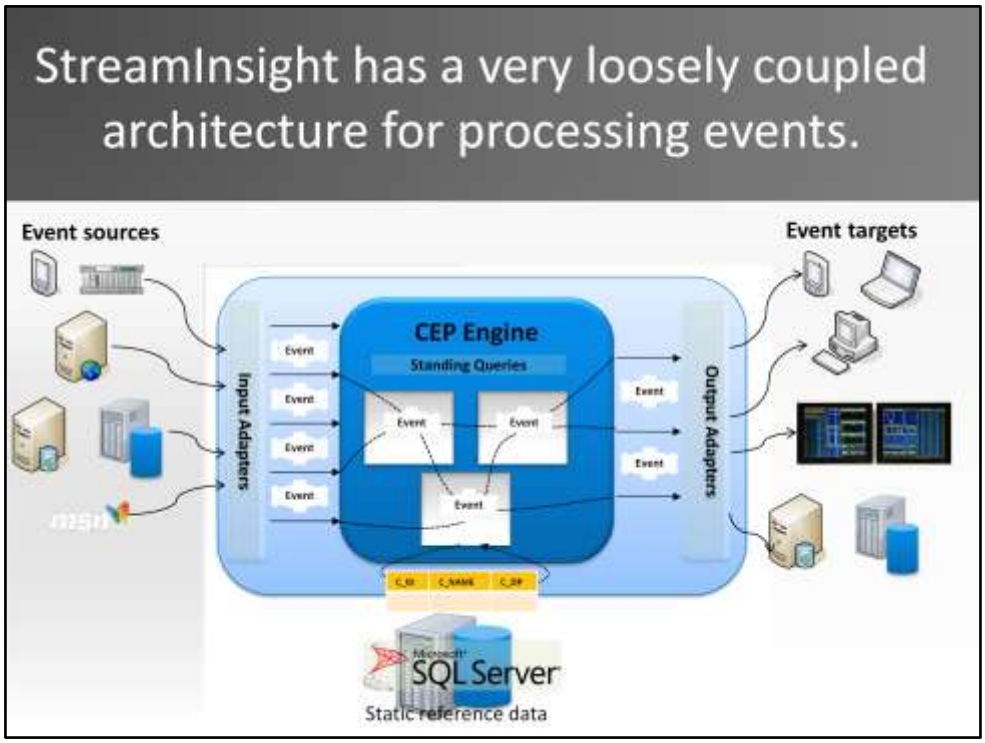
Doing effective CEP requires deep business knowledge and thorough analysis.



Situational awareness

- **Analysts must study the business situation to identify patterns that reflect situations of interest. **
  - May be done by replaying past situations and applying rule permutations to see which ones reveal the most meaningful connections.
    - E.g. a service is considered unavailable if offline for more than X seconds and NOT in a maintenance state
  - Information about events happening in the moment could be correlated with historical data to predict future patterns. Anticipate what was about to take place before the moment arrives.
  - Big mental model allows quick decision making
- Processing isn't just matching a pattern; may be decisions you make based on event

- EPAs, also sometimes know as event mediators, are needed to detect patterns in raw events, to process these events through enrichment, transformation, and validation, and finally to derive new events and publish them.
- Event processing agents (EPAs) may: filter events, copy data from base event and change format, reorder; enrich events by adding relevant data from other sources; compute aggregate values such as totals, avgs, min/max, topK from multiple events; split the stream, do other CEP by applying rules to detect patterns in a set of base events; threshold queries; correlation across sources
- Real-time customer evaluation, automated anomaly using high volume sensors or biz apps
- **Value of in-memory, **incremental** processing**
- Can process 100,000 events per second (high volume, low latency)

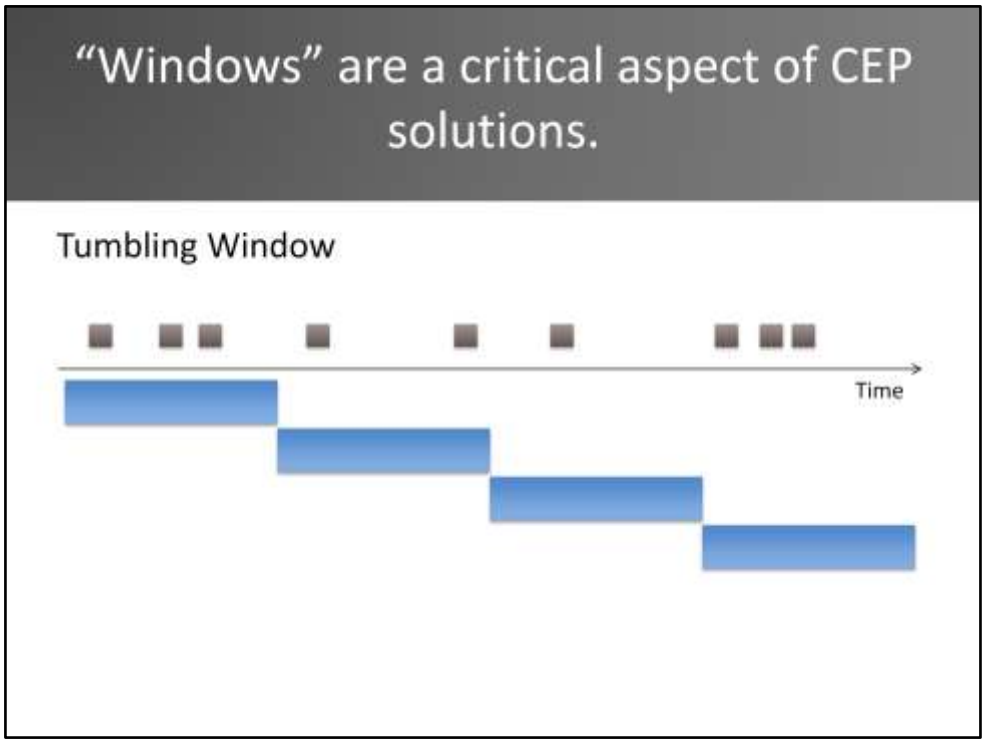


- Events can be point in time, interval with fixed duration, or interval with initially unknown duration (EXAMPLES)
  - Point event – an order
  - Interval event – auction bid
  - Edge event – phone call in service center
- Stream is a possibly infinite sequence of events; may get out of order events
- Check for absence of data
- Use other streams of events
- Leverage reference data
- Deploy as embedded engine or central service

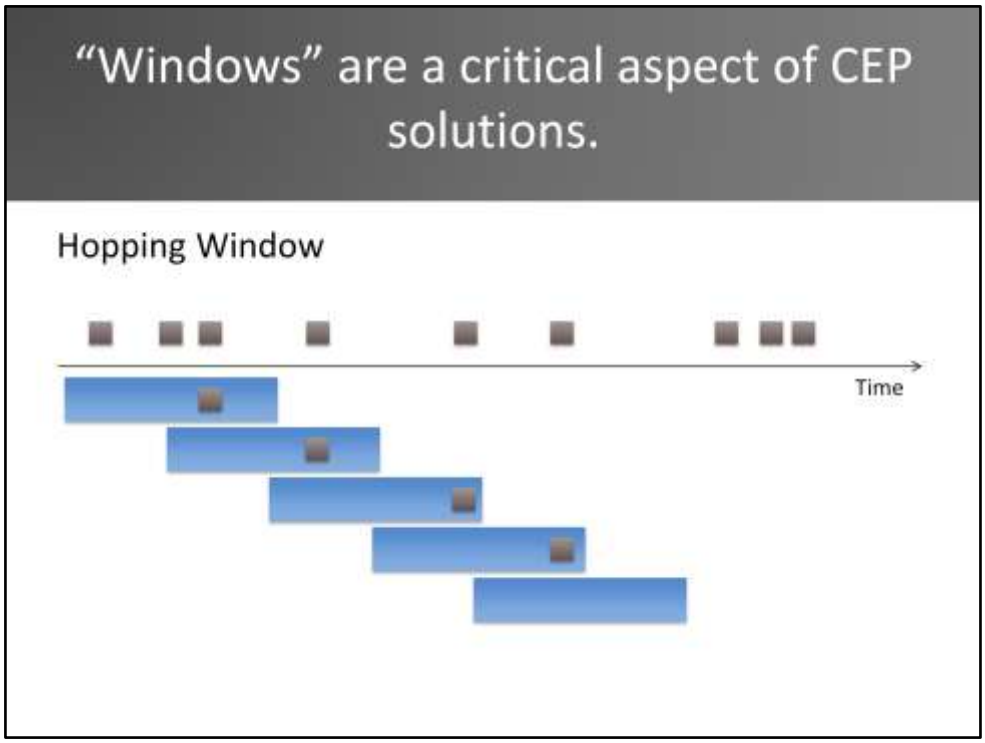
- Event windows

“Windows” are a critical aspect of CEP solutions.

- Windows rely on APPLICATION time, not system time
- Write queries against a window
    - If I create a window of all the parts manufactured in a plant within the last hour, I can see query the average size, …



- Creates windows of a specific size, and flushes the results before creating a new window



- Much more liveliness; may get earlier notification of a trend
- If something starts happening at the end of the first window, we're likely to see it roll into the metrics of the second window

Standing queries are written using the LINQ syntax for .NET.

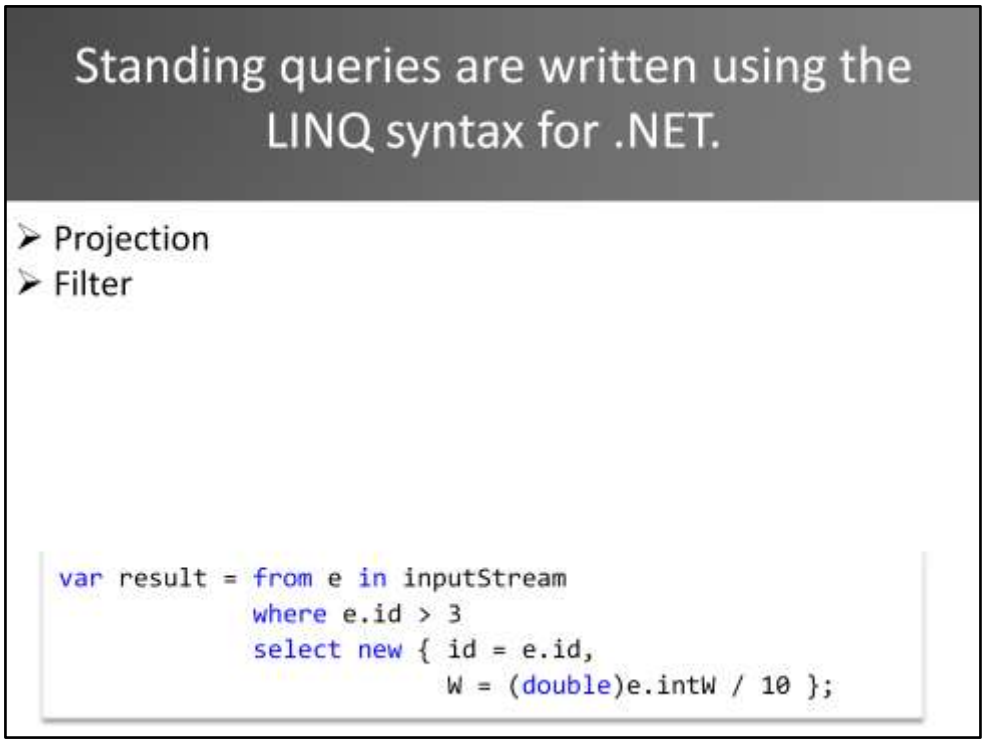
- Same queries for in-flight or replay scenarios
- Operators over streams: project, join, exists, filter, group/apply, aggregation, ranking
- User defined functions and operators when domain-specific scenarios occur
- Company's IP is in the queries

Standing queries are written using the LINQ syntax for .NET.

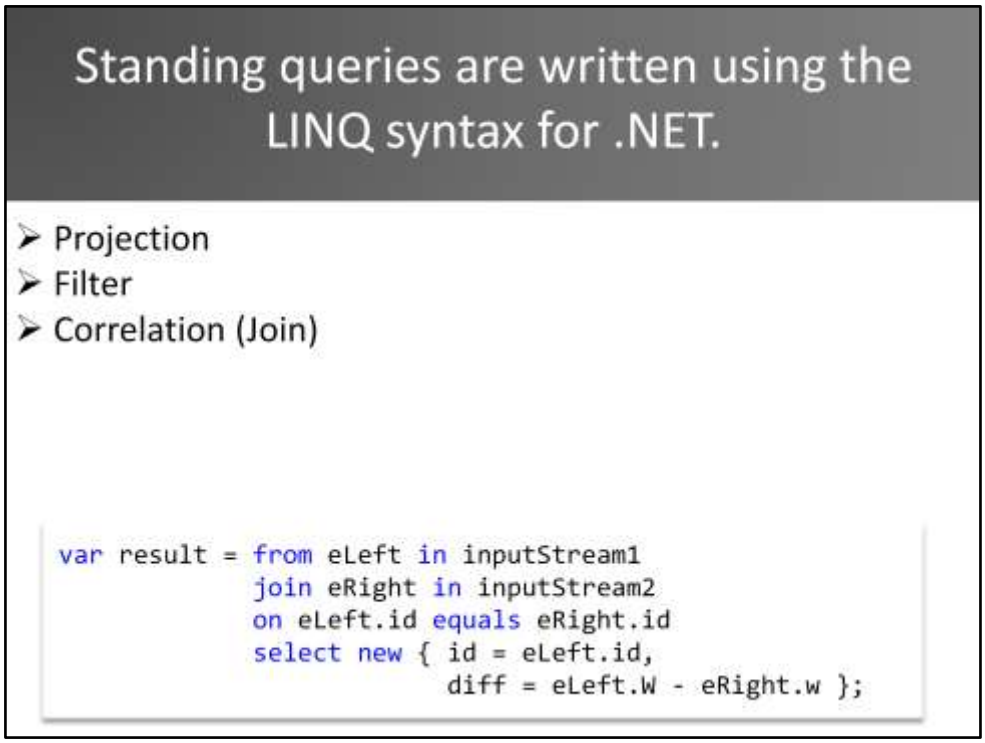
- Projection

```
var result = from e in inputStream
             select new { id = e.id,
                          W = (double)e.intW / 10 };
```

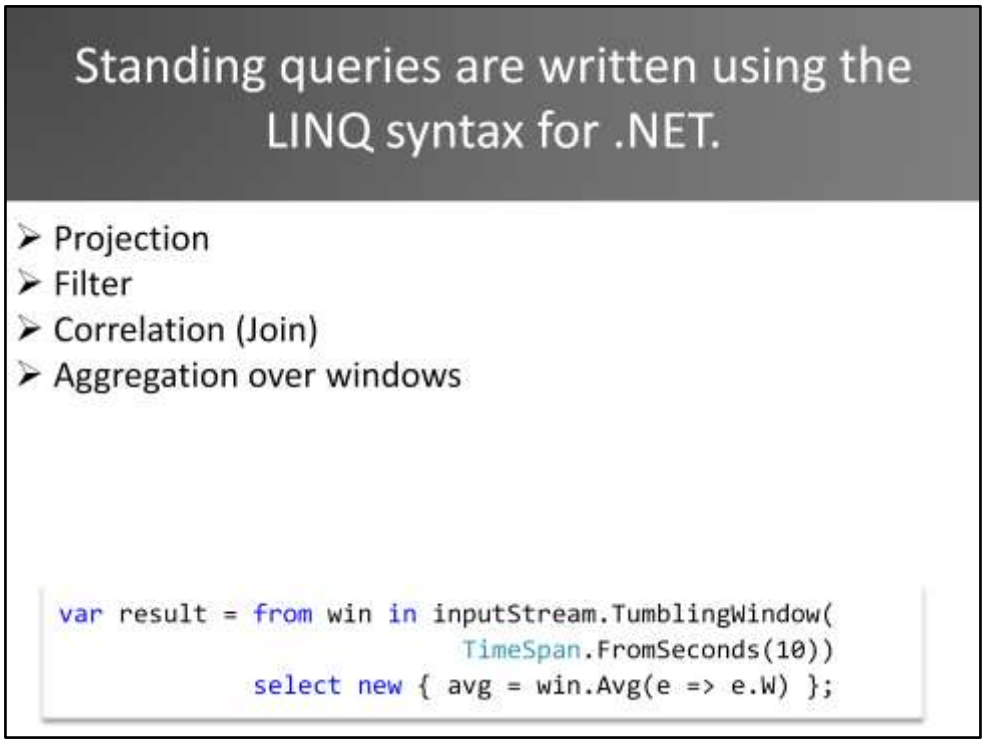
- Maybe do unit conversations, or other manipulation of data



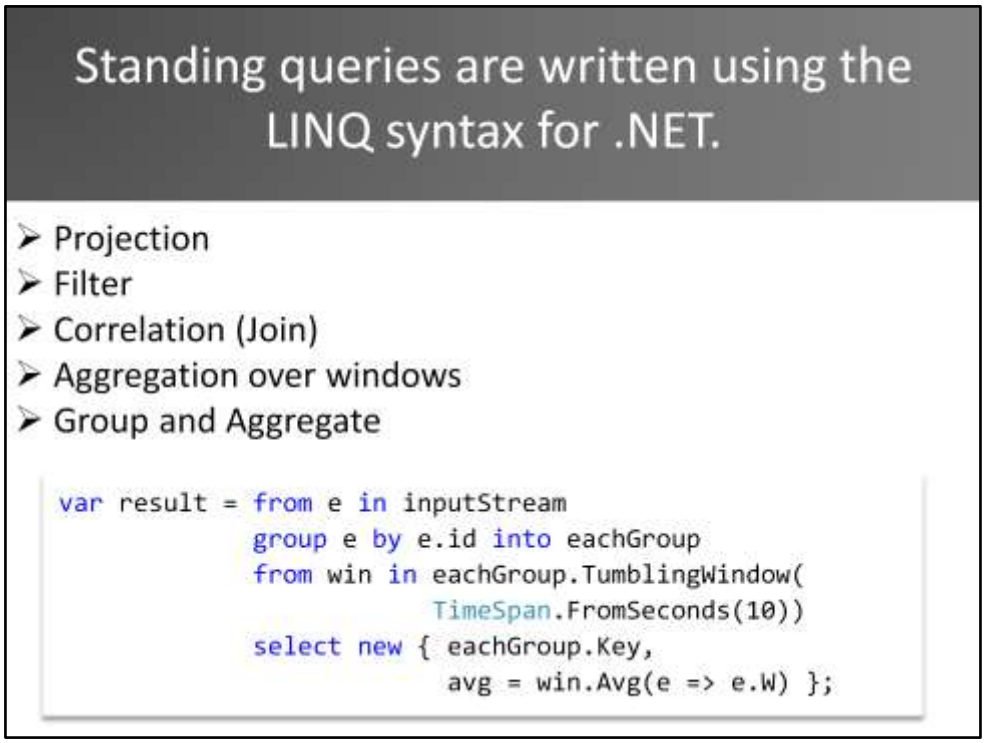
- Less data, not more (through summation)
- “where” clause
- Could use “user defined function” and filter out all events that didn’t exceed a previously-defined threshold (yesterday’s max)



- Train the model with analytics, and prime the model with reference data
- If events overlap, they can referenced in the join
- Could join to …
    - Other streams (e.g. combine marketing campaign events with call center events)
    - Reference data (list of average order amounts from previous day)



- Assess events within a given window and calculate averages, sum, min, max, count and other user-defined options
    - Compiled incrementally, so very high performing



- Group based on key(s)
- Also, have extension functions for more complex calculations (standard deviation, etc)

Generates ticker events, and the query puts all ticker quotes into a 5 second window and recalculates the window every 100 miliseconds

produces averages of all the ticker events in the window

- in event driven CEP, don't have to remember to look at reports and system notifies you only when items need attention
- Event driven CEP is useful in **scenarios that NEED continuous intelligence** (lots of event data, short decision cycles)
- Has a place in an entire BI stack where long term analytics and what-ifs are required
- Consider for fraud, safety, process monitoring, competition awareness
  - High volume, need low latency response, complicated temporal patterns
- SENSE AND RESPOND
- Usual way, plowing through historical data to find trends and insights, isn't good enough
- Still great deal of value in deeply mining databases to discover large trends


QCon
北京站·2012年4月18~20日
www.qconbeijing.com（11月启动）
QCon杭州站官网和资料
www.qconhangzhou.com
全球企业开发大会
INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE