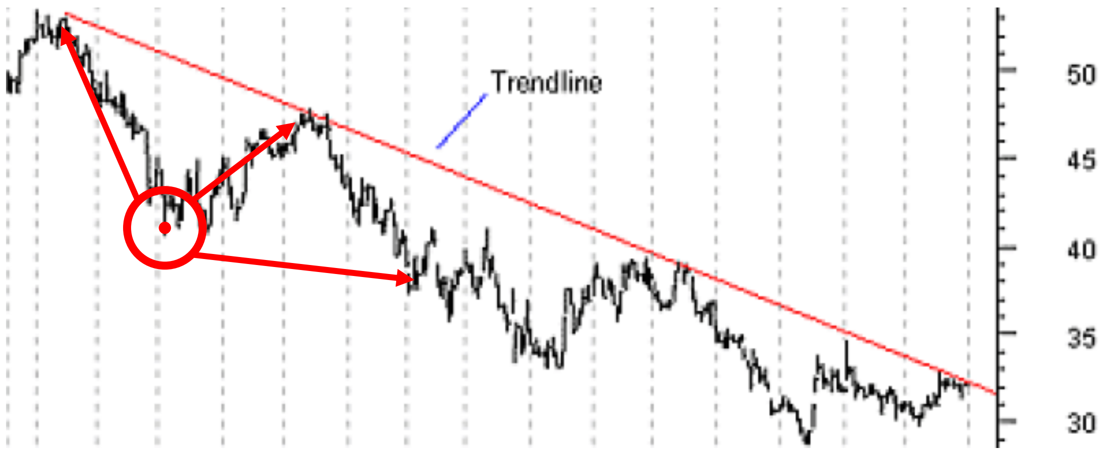
Event Trend Aggregation Under Rich Event Matching Semantics

Olga Poppe[1], **Chuan Lei**[2],
Elke A. Rundensteiner[3], and David Maier[4]
[1]Microsoft Gray Systems Lab, [2]IBM Research – Almaden,
[3]Worcester Polytechnic Institute, [4]Portland State University

July 3rd, 2019

Supported by NSF grants IIS-1815866, CRI-1305258, IIS-1018443

Algorithmic Trading

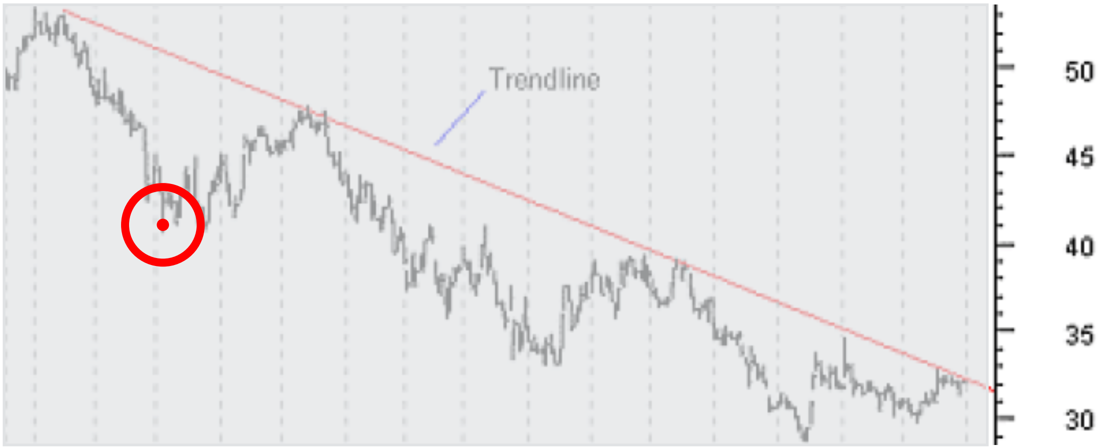


Goal: Reliable actionable insights about the stream

Solution: Each event is considered in the context of other events in the stream

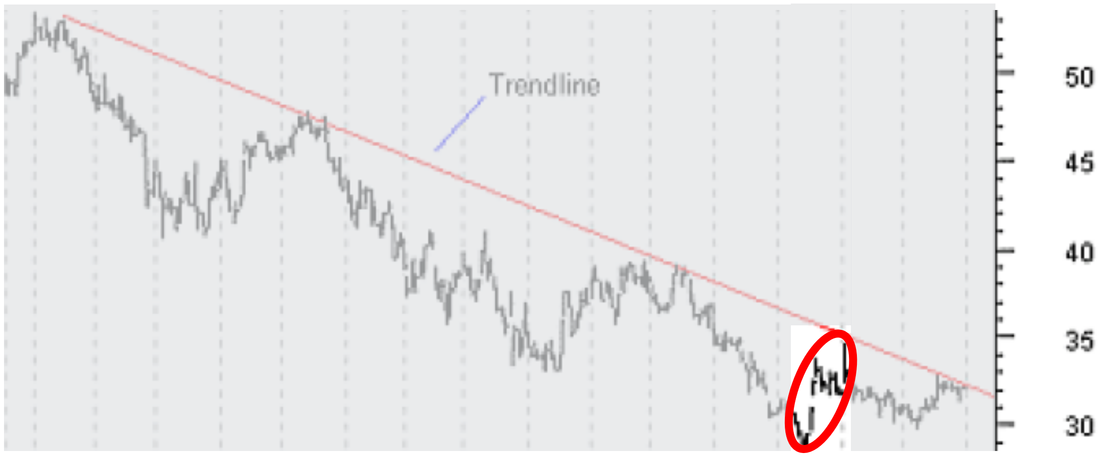
Picture source: http://www.businessxack.com/how-to-know-the-stock-market-trend/1303

Algorithmic Trading



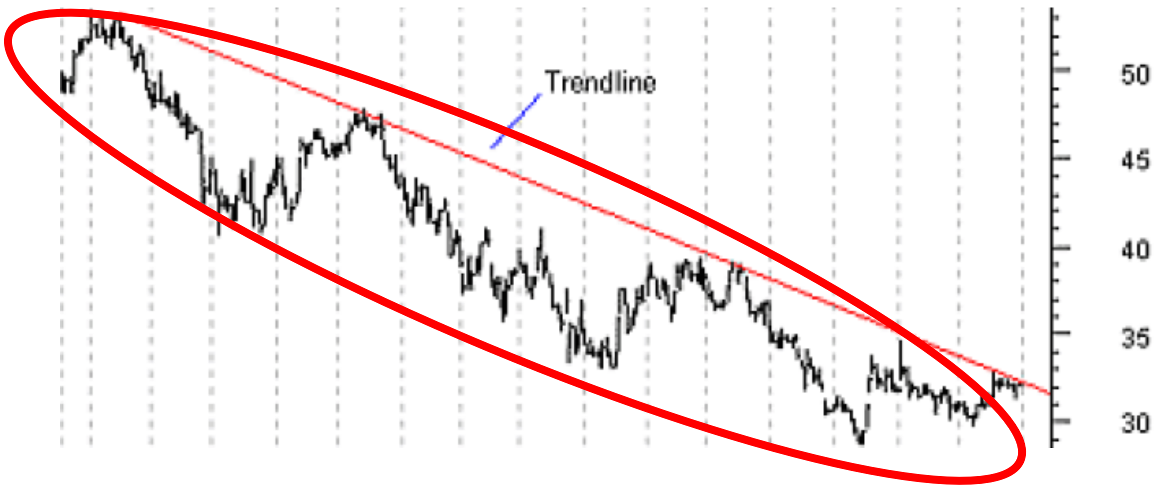
- Single event = *Single* stock value
- Event sequence = Stock down trend of *fixed* length
- Event trend = Stock down trend of *arbitrary* length

Algorithmic Trading



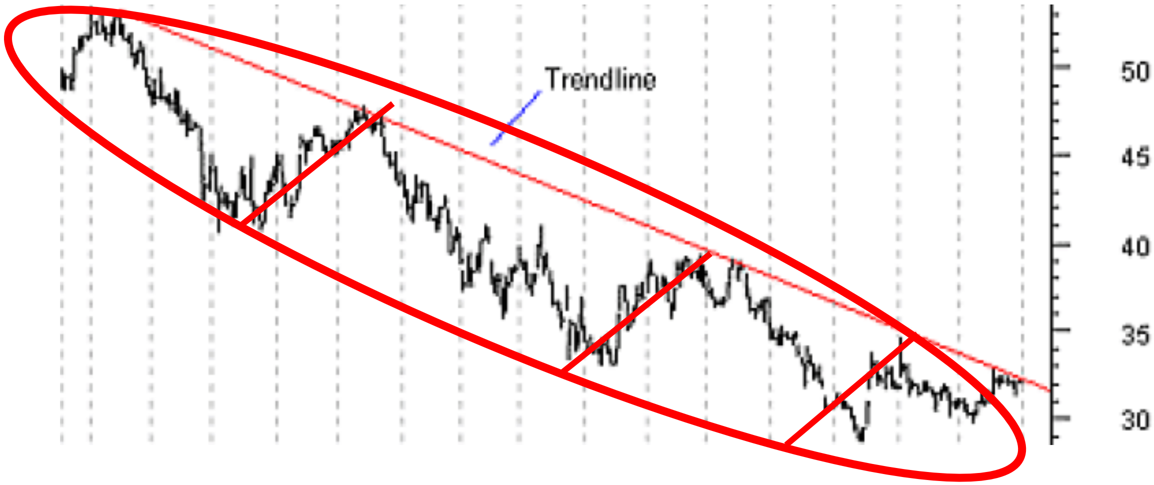
- Single event = *Single* stock value
- Event sequence = Stock down trend of *fixed* length
- Event trend = Stock down trend of *arbitrary* length

Algorithmic Trading



- Single event = *Single* stock value
- Event sequence = Stock down trend of *fixed* length
- Event trend = Stock down trend of *arbitrary* length

Algorithmic Trading



- Single event = *Single* stock value
- Event sequence = Stock down trend of *fixed* length
- Event trend = Stock down trend of *arbitrary* length under the *skip-till-next-match* semantics

Event Trend Aggregation Under Rich Event Matching Semantics

Algorithmic Trading

Number of down-trends per sector **ignoring local price fluctuations**

Skip-till-any-match semantics

Ridesharing Service

Average speed of Uber trips per district **ignoring irrelevant events**

Skip-till-next-match semantics

Cluster Monitoring

Total CPU load per mapper experiencing **contiguously** increasing load

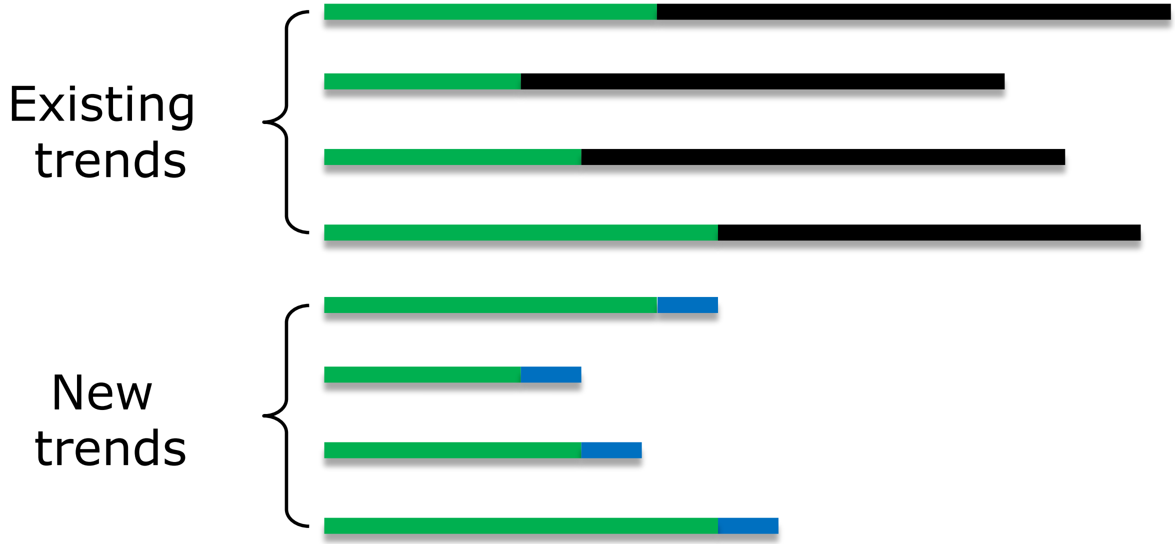
Contiguous semantics

E.Wu, Y.Diao, and S.Rizvi. High-performance Complex Event Processing over streams. SIGMOD, pages 407-418, 2006

Complexity of Event Trend Analytics

Complexity of Event Trend Analytics

Complexity of Event Trend Analytics



Real-time event trend aggregation despite

- Rich event matching semantics
- Exponential number and arbitrary length of trends
- Complex event inter-dependencies in a trend

Existing Two-Step Approaches

Step 2: **Event Trend Aggregation**

Step 1: **Event Trend Construction**

Exponential time & space complexity

Event Trend Aggregation Query

RETURN sector, COUNT(*)
PATTERN Stock S+
WHERE [company, sector] **AND** S.price > **NEXT**(S).price
SEMANTICS skip-till-any-match
GROUP-BY sector **WITHIN** 30 min **SLIDE** 1 min

Event Stream

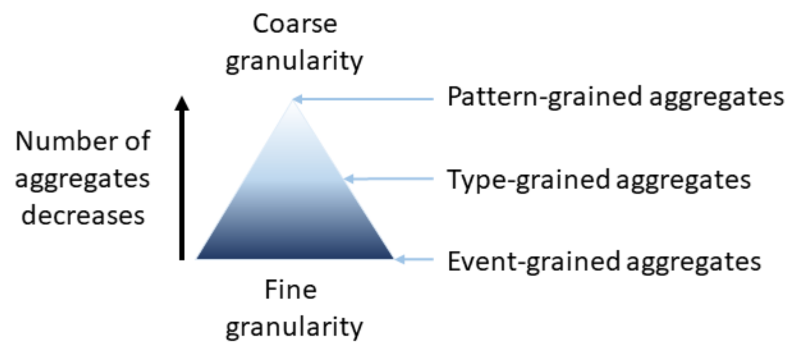
Transaction event

- Sector id
- Company id
- Price
- Time

Picture source: http://www.zerohedge.com/news/2015-12-05/dozens-global-stock-markets-are-already-crashing-not-seen-numbers-these-2008

Coarse-Grained Online Trend Aggregation

Cogra: Coarse-Grained Online Trend Aggregation



Quadratic time & **linear** space complexity

Event Trend Aggregation Query

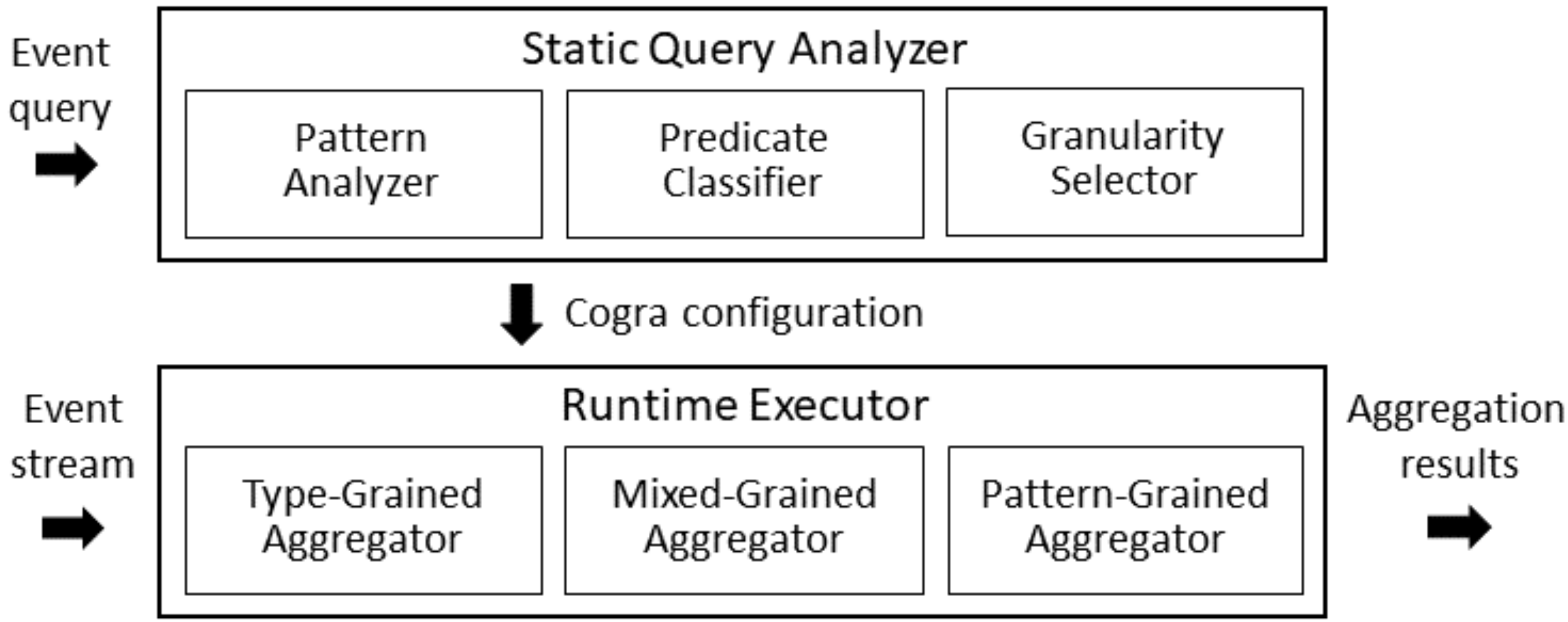
RETURN sector, COUNT(*)
PATTERN Stock S+
WHERE [company, sector] **AND** S.price > **NEXT**(S).price
SEMANTICS skip-till-any-match
GROUP-BY sector **WITHIN** 30 min **SLIDE** 1 min

Event Stream

Transaction event
- Sector id
- Company id
- Price
- Time

Approach Overview

COGRA Framework

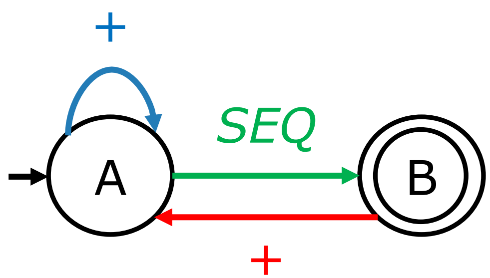


Cogra Template

Nested Kleene Pattern

$$P = (SEQ(A+, B))+$$

a’s are preceded by a’s and b’s

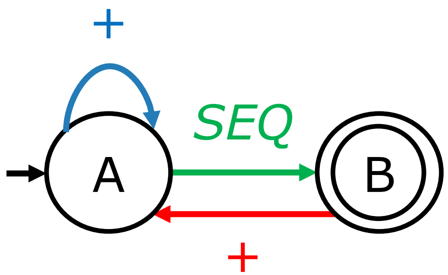


b’s are preceded by a’s

Start type **End type**

Online Type-Grained Aggregator

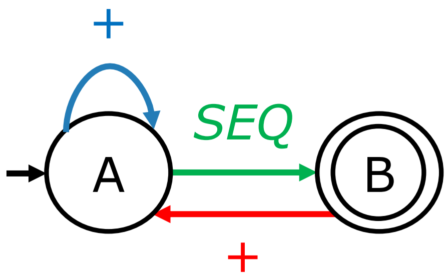
for skip-till-any-match semantics



Event	a.count	b.count	A.count	B.count
a1	1			

Online Type-Grained Aggregator

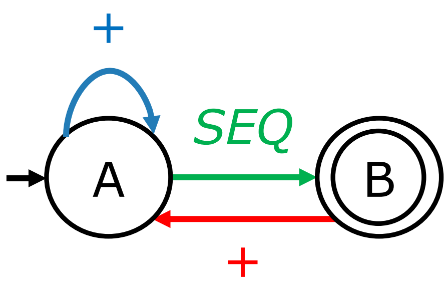
for skip-till-any-match semantics



Event	a.count	b.count	A.count	B.count
a1	1 →		1	

Online Type-Grained Aggregator

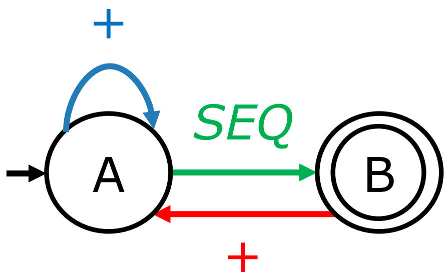
for skip-till-any-match semantics



Event	a.count	b.count	A.count	B.count
a1	1		1	
b2		1		

Online Type-Grained Aggregator

for skip-till-any-match semantics

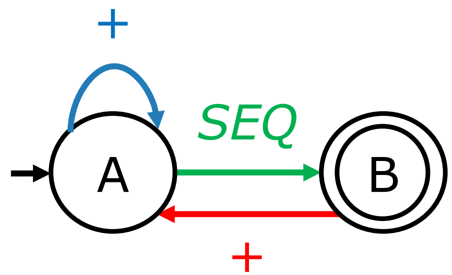


Event	a.count	b.count	A.count	B.count
a1	1		1	
b2		1		1

Event trends:
(a1,b2)

Online Type-Grained Aggregator

for skip-till-any-match semantics

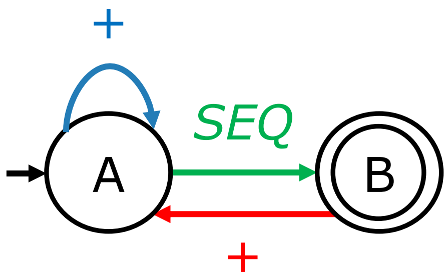


Event	a.count	b.count	A.count	B.count
a1	1		1	
b2		1		1
a3	3			

Event trends:
(a1,b2)

Online Type-Grained Aggregator

for skip-till-any-match semantics

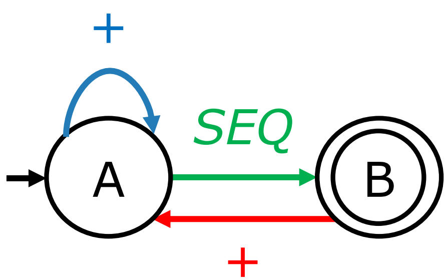


Event	a.count	b.count	A.count	B.count
a1	1		1	
b2		1		1
a3	3		4	

Event trends:
(a1,b2)

Online Type-Grained Aggregator

for skip-till-any-match semantics



Event	a.count	b.count	A.count	B.count
a1	1		1	
b2		1		1
a3	3		4	
a4	6		10	
b6		10		11
a7	22		32	
b8		32		43

Event trends:
(a1,b2)
(a1,a3,b6)
(a1,a3,a4,b6)
(a1,b2,a3,a4,b6)
(a1,b2,a2,b6,a7,b8)
(a1,b2,a2,a3,b6,a7,b8)
…

Online Type-Grained Aggregator

for skip-till-any-match semantics

	Existing Two-Step Approaches	Cogra
Idea	1. Construct all trends 2. Aggregate them	One aggregate is kept per event type
Time complexity	Exponential in #events per window	Linear in #events per window, i.e., **optimal**
Space complexity	Exponential if all trends are stored	Linear in #event types in the pattern

Online Pattern-Grained Aggregator

for skip-next-any-match & contiguous semantics

	Existing Two-Step Approaches	Cogra
Idea	1. Construct all trends 2. Aggregate them	One aggregate is kept per pattern
Time complexity	Polynomial in #events per window	Linear in #events per window, i.e., **optimal**
Space complexity	Polynomial if all trends are stored	Constant

Cogra enables **real-time in-memory** event trend aggregation

Experimental Setup

Execution infrastructure:

Java 8, 1 Linux machine with 16-core

3.4 GHz CPU and 128 GB of RAM

Data sets:

- New York city taxi and Uber data set (330 GB)
  - Event trend = Taxi or Uber trip
- Physical activity real data set (1.6 GB)
  - Event trend = Sequence of physical activities
- Stock real data set (1.3 GB)
  - Event trend = Stock market trend

- Unified New York City Taxi and Uber data. https://github.com/toddwschneider/nyc-taxi-data
- Historical Stock Data. http://www.eoddata.com
- A.Reiss and D.Stricker. Creating and Benchmarking a New Dataset for Physical Activity Monitoring. In PETRA, 2012, 40:1–40:8

Event Aggregation Approaches

Approaches	Kleene closure	Event matching semantics			Online sequence\trend aggregation
		Skip-till-any-match	Skip-till-next-match	Contiguous	
Flink	+	+	+	+	--
Sase	+	+	+	+	--
Greta	+	+	--	--	+
A-Seq	--	+	--	--	+
Cogra	+	+	+	+	+

Flink: https://fink.apache.org/

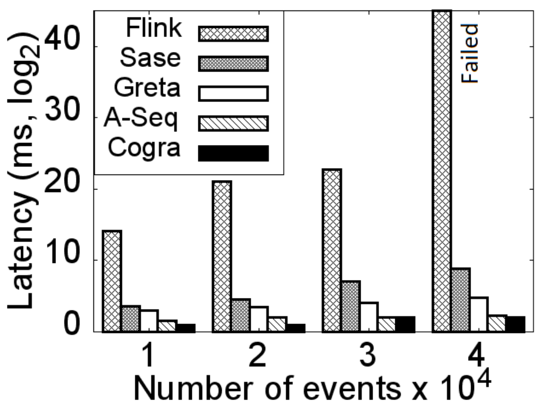
Sase: H.Zhang, Y.Diao, and N.Immerman. On complexity and optimization of expensive queries in Complex Event Processing. In SIGMOD, pages 217-228, 2014

Greta: O.Poppe, C.Lei, E.A.Rundensteiner and D.Maier. Greta: Graph-based Real-time Event Trend Aggregation. In VLDB, pages 80-92, 2017

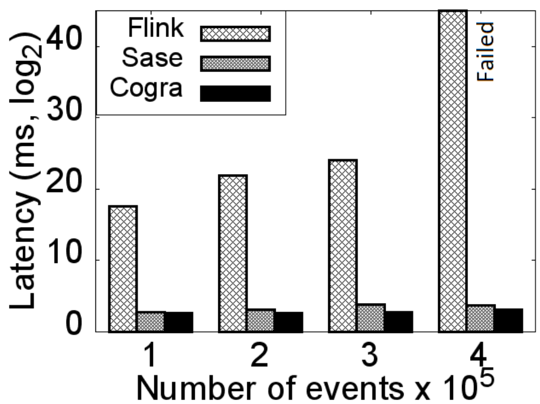
A-Seq: Y.Qi, L.Cao, M.Ray, and E.A.Rundensteiner. Complex Event Analytics: Online Aggregation of Stream Sequence Patterns. In SIGMOD, pages 229–240, 2014

Experimental Results

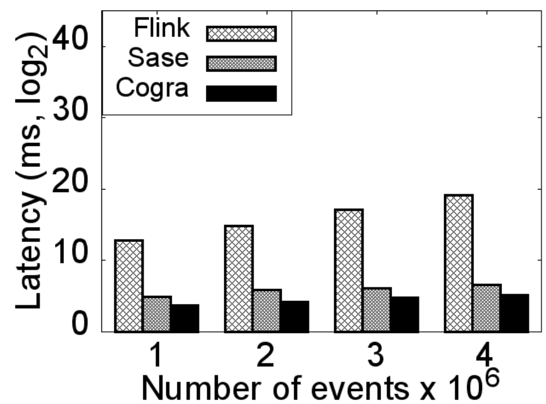
Skip-till-any-match semantics



Skip-till-next-match semantics



Contiguous semantics



Cogra is a win-win solution that achieves up to **10^6 speed-up** and up to **10^7 memory reduction** compared to state-of-the-art

Contributions

We are the first to compute **aggregation of Kleene pattern matches under rich event matching semantics** with **optimal time complexity**

- Cogra **incrementally** maintains event trend aggregates at the **coarsest granularity**
- Cogra guarantees **quadratic** time complexity and **linear** space complexity in the number of events in the worst case
- Cogra enables **real-time in-memory** event trend aggregation as required by time-critical streaming applications

Thank
You
Merci
Gracias
TerimaKasih
Hvala
Arigatô
KamSahHamnida
Spacibo
Danke
Mèsi
Xièxie
Chokrane
Dziakuju
Grazie
Takk
aDank
BarakAllahuFiik
ThintKo
Buznyg
Blagodaram
Rahmat
Matondo
Mercé
Taiku
Dakujem
Gràcies
Faleminderit
Efharisto
NaijsTuke
Grandmerce
Tanemirt
Tenki
Bayarlalaa
Nirringrazzjak
Obrigado
Salamat
Saha
Multumesc
CámOn
Gracie
GratiasAgo
KopKhunKha
Motashakkeram
Sagolun
Tak
Dankie
TänanVäga
Ačiū
GratiasAgimus
Mauruuru
Kiitos
Barkal
Chnorakaloutioun
Aabhar
Dhanyavaad
Murakoze
Köszönöm
TapadhLeat
WaadMahadsantahay
Grazzi
Gratzias
KhobChaiDeu