Stream Reasoning For Linked Data
M. Balduini, J-P Calbimonte, O. Corcho, D. Dell'Aglio, and E. Della Valle
http://streamreasoning.org/events/sr4Id2014

Stream Reasoning introduction
Emanuele Della Valle
emanuele.dellavalle@polimi.it
http://emanueledellavalle.org
Share, Remix, Reuse — Legally

- This work is licensed under the Creative Commons Attribution 3.0 Unported License.

- **Your are free:**
  - ![to Share](image) — to copy, distribute and transmit the work
  - ![to Remix](image) — to adapt the work

- **Under the following conditions**
  - ![Attribution](image) — You must attribute the work by inserting
    - “[source http://streamreasoning.org/events/sr4ld2014]” at the end of each reused slide
    - a credits slide stating
      - These slides are partially based on “Streaming Reasoning for Linked Data 2014” by M. Balduini, J-P Calbimonte, O. Corcho, D. Dell’Aglio, and E. Della Valle, [http://streamreasoning.org/events/sr4ld2013](http://streamreasoning.org/events/sr4ld2013)

- To view a copy of this license, visit [http://creativecommons.org/licenses/by/3.0/](http://creativecommons.org/licenses/by/3.0/)
Agenda

- It's a streaming world
- Continuous semantics
- Data Stream Management Systems and Complex Event Processors
- Stream Reasoning
- Research Challenges
- Approaches
- Structure of the tutorial
- More on Stream Reasoning at ISWC 2014
It’s a streaming World!

What Happens in an Internet Minute?

- 639,800 GB of global IP data transferred
- 6 New Wikipedia articles published
- 135 Botnet infections
- 1,300 New mobile users
- 20 New victims of identity theft
- 204 million Emails sent
- 47,000 App downloads
- $83,000 In sales
- 100+ New LinkedIn accounts
- 61,141 Hours of music
- 20 million Photo views
- 320+ New Twitter accounts
- 3,000 Photo uploads
- 30 Hours of video uploaded
- 1.3 million Video views
- 277,000 Logins
- 6 million Facebook views
- 2+ million Search queries
- 100,000 New tweets

And Future Growth is Staggering

It’s a streaming World!

- Oil operations
- Traffic
- Financial markets
- Social networks
- Generate data streams!
What is the expected time to failure when that turbine's barring starts to vibrate as detected in the last 10 minutes?

Is public transportation where the people are?

Who are the best available agents to route all these unexpected contacts about the tariff plan launched yesterday?

Who is driving the discussion about the top 10 emerging topics?

A system able to answer those queries must be able to

- handle **massive datasets**
  - A typical oil production platform is equipped with about **400,000 sensors**
  
  - Telecom data is the most pervasive data source in urban areas, in Milano there are **1.8 million mobile users**

  - A global contact centre of a Telecom operator counts **500 millions of clients**

  - Facebook alone has **1.1 billion** of active **users**
A system able to answer those queries must be able to

- process **data streams** on the fly
  - The sensors on typical oil production platform generates **10,000** observations per minute with **peaks of 100,000 o/m**
  - The mobile users in Milano generates **20,000** call/sms/data connections per minute with **peaks of 80,000 c/m**
  - A global contact centre receives **10,000** contacts per minute with **peaks of 30,000 c/m**
  - Facebook, as of May 2013, observes **3 millions "I like"** per minute
A system able to answer those queries must be able to

- **cope with heterogeneous dataset**
  - The sensors on typical oil production have been deployed over 10 years by *10s of different producers*

- **Tens of data sources** are normally needed to make sense of an urban phenomena

- A global contact centre consists in *100s of offices* owned by different subsidiary companies *engaged yearly*

- Each social network has *its own data model, APIs, ...*
A system able to answer those queries must be able to

- **cope with incomplete data**
  - 10s of *sensors* and networking links broke down daily

- **Coverage is incomplete**

- Only standard cases are covered by fully machine processable data records

  100s of contacts per minute are manage ad-hoc

- **Conversations** happen outside the social networks, too!
A system able to answer those queries must be able to

- **cope with noisy data**
  - Sensor out-of-operating range
  - Faulty sensors
  - Agents misunderstand, get tired, ...
  - Irony, sarcasm, ...
A system able to answer those queries must be able to

- provide **reactive answers**
  - detection of dangerous situations must occur within **minutes**
  - recommendations to citizens must be performed in **few seconds**
  - routing a contact through each step of the decision tree must take less than a **second**
  - Search autocompleting may need to be updated every **few minutes**
A system able to answer those queries must be able to

- support **fine-grained information access**
  - Identify **a turbine** among thousands
  - Locate **a bus** among thousands
  - Contact **an agent** among thousands
  - Identify **an opinion maker** among thousands of influencers for a topic
A system able to answer those queries must be able to

- integrate **complex domain models** of
  - operational and control process

- **various city aspects**

- **contact** management, **contract** types, **agent skills**, **contactor** profiles, ...

- **topics**, **user** profiles, ...
Requirements (wrap up)

A system able to answer those queries must be able to

- handle **massive datasets**
- process **data streams** on the fly
- cope with **heterogeneous dataset**
- cope with **incomplete data**
- cope with **noisy data**
- provide **reactive answers**
- support **fine-grained information access**
- integrate **complex domain models**
What are data streams anyway?

- Formally:
  - Data streams are unbounded sequences of time-varying data elements

- Less formally:
  - an (almost) “continuous” flow of information

- Assumption
  - recent information is more relevant as it describes the current state of a dynamic system
The continuous nature of streams

- The nature of streams requires a paradigmatic change*
  - **from persistent data**
    - to be stored and queried on demand
    - a.k.a. one time semantics
  - **to transient data**
    - to be consumed on the fly by continuous queries
    - a.k.a. continuous semantics

* This paradigmatic change first arose in DB community [Henzinger98]
Continuous queries registered over streams that, in most of the cases, are observed through windows.
Example

- **Input**
  - Smoke and Temperature sensors in many areas

- **Query**
  - Alert me when there is a fire, i.e. smoke and temp > 50

- **DSMS formulation**
  - Stream the areas where smoke is detected over two windows open on smoke and temperature streams
    
    Select IStream(Smoke.area)
    From Smoke[Rows 30 Slide 10], Temp[Rows 50 Slide 5]
    Where Smoke.area = Temp.area AND Temp.value > 50

- **CEP formulation**
  - Rise a fire event in an area when smoke and high temperature events are received within 1 minute
    
    define Fire(area: string, measuredTemp: double)
    from Smoke(area=$a) and
    each Temp(area=$a and val>50) within 1Min.
    where area=Smoke.area and measuredTemp=Temp.value

Content
- Type of models compared
  - Functional and processing
  - Deployment and interactions
  - Data, Time, and Rule
  - Language
- # of systems surveyed:
  - Academic: 24
  - Industrial: 9
  - Total: 33
- To learn more:
DSMS/CEP Market Players

CEP Market Players to Dec 2012

[source](https://ctrlaltcep.files.wordpress.com/2013/01/cepmarket1212.png)

http://streamreasoning.org/events/sr4ld2014
## Existing solutions

<table>
<thead>
<tr>
<th>Requirement</th>
<th>DSMS</th>
<th>CEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>massive datasets</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>data streams</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>heterogeneous dataset</td>
<td>✗</td>
<td></td>
</tr>
<tr>
<td>incomplete data</td>
<td>✗</td>
<td></td>
</tr>
<tr>
<td>noisy data</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>reactive answers</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>fine-grained information access</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>complex domain models</td>
<td>✗</td>
<td></td>
</tr>
</tbody>
</table>
Given ontology $O$ and query $Q$, use $O$ to rewrite $Q$ as $Q'$ so that, for any set of ground facts $A$ contained in multiple databases:

- $\text{answer}(Q, O, A) = \text{answer}(Q', \varnothing, A)$
  - The answer of the query $Q$ using the ontology $O$ for any set of ground facts $A$ is equal to answer of a query $Q'$ without considering the ontology $O$

Use mapping $M$ to map $Q'$ to multiple SQL queries to the various databases
## Existing solutions

<table>
<thead>
<tr>
<th>Requirement</th>
<th>DSMS</th>
<th>CEP</th>
<th>OBDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>massive datasets</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>data streams</td>
<td>✓</td>
<td></td>
<td>✗</td>
</tr>
<tr>
<td>heterogeneous dataset</td>
<td>✗</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>incomplete data</td>
<td>✗</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>noisy data</td>
<td>✓</td>
<td></td>
<td>✗</td>
</tr>
<tr>
<td>reactive answers</td>
<td>✓</td>
<td></td>
<td>✗</td>
</tr>
<tr>
<td>fine-grained information access</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>complex domain models</td>
<td>✗</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
Stream Reasoning Definition

- Making sense
  - in real time
  - of multiple, heterogeneous, gigantic and inevitably noisy data streams
  - in order to support the decision process of extremely large numbers of concurrent user

- Note: making sense of streams necessarily requires processing them against rich background knowledge, an unsolved problem in database

Research Challenges

- Relation with DSMSs and CEPs
  - Just as RDF relates to data-base systems?

- Data types and query languages for semantic streams
  - Just RDF and SPARQL but with continuous semantics?

- Reasoning on Streams
  - Theory: formal semantics
  - Efficiency
  - Scalability and approximation

- Dealing with incomplete & noisy data
  - Even more than on the current Web of Data

- Distributed and parallel processing
  - Streams are parallel in nature, data stream sources are distributed, ...

- Engineering Stream Reasoning Applications
  - Development Environment
  - Integration with other technologies
  - Benchmarks as rigorous means for comparison
Stream Reasoning feasibility (intuition)

- Many relevant reasoning methods are not able to deal with high frequency data streams
- However, trade-off exists between the complexity of the reasoning method and the frequency of the data stream the reasoner

Approaches (a selection) 1/4

- **RDF Stream Processors (ordered by year)**
  - **Streaming SPARQL**
    - Andre Bolles, Marco Grawunder, Jonas Jacobi: Streaming SPARQL - Extending SPARQL to Process Data Streams. ESWC 2008: 448-462
  - **C-SPARQL**
  - **SPARQL\textsubscript{stream}**
  - **CQELS**
  - It continues in next slide ...
Approaches (a selection) 2/4

• ... it continues from previous slide

**INSTANS**

**Streaming Linked Data**
  – Marco Balduini, Emanuele Della Valle, Daniele Dell’Aglio, Mikalai Tsytsarau, Themis Palpanas, Cristian Confalonieri: Social listening of City Scale Events using the Streaming Linked Data Framework. ISWC 2013

**TEF-SPARQL** (under development)
  – Shen Gao, Thomas Scharrenbach, Abraham Bernstein: The CLOCK Data-Aware Eviction Approach: Towards Processing Linked Data Streams with Limited Resources. ESWC 2014: 6-20
Stream Reasoners (ordered by year)

**Streaming Knowledge Bases**

**IMaRS**

**TrOWL**
- Yuan Ren, Jeff Z. Pan: Optimising ontology stream reasoning with truth maintenance system. CIKM 2011: 831-836

**ETALIS (EP-SPARQL)**

- It continues in next slide ...
Approaches (a selection) 4/4

- ... continues from previous slide
- **Sparkwave**
  - Srdjan Komazec, Davide Cerri, Dieter Fensel: Sparkwave: continuous schema-enhanced pattern matching over RDF data streams. DEBS 2012: 58-68
- **SR-Based on Answer Set Programming**
- **Parallelising Stream Reasoning**
  - Chang Liu, Jacopo Urbani, Guilin Qi: Efficient RDF stream reasoning with graphics processing units (GPUs). WWW (Companion Volume) 2014: 343-344
- **STARQL (under development)**
  - ÖL Özçep, R Möller, C Neuenstadt, “A Stream-Temporal Query Language for Ontology Based Data Access”. Description Logics, 2014
Running Example

BlueRoom

Bob

Carl

BlueSensor

RedRoom

RedSensor

Alice

David

Elena

RFID

Foursquare

Facebook

is with
### Four ways to learn who is where

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Room</th>
<th>Person</th>
<th>Time-stamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>RedSensor</td>
<td>RedRoom</td>
<td>Alice</td>
<td>$T_1$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Person</th>
<th>ChecksIn</th>
<th>Time-stamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob</td>
<td>BlueRoom</td>
<td>$T_2$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Person</th>
<th>IsIn</th>
<th>With</th>
<th>Time-stamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Carl</td>
<td>null</td>
<td>Bob</td>
<td>$T_2$</td>
</tr>
<tr>
<td>David</td>
<td>RedRoom</td>
<td>Elena</td>
<td>$T_3$</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Running Example – Data Model

Streaming information

Observation

subClassOf

Post

discusses

who

subPropOf

discusses

posts

where

Background information

Room

isConnectedTo

Sensor

subClassOf

Person

isWith
Running Example – Data Model (formally)

- Details about hands-on ontology
  - isConnectedTo is a symmetric property
  - discusses is a transitive property
  - isWith is a composition of posts and who
  - isIn is either a composition of posts and where
    or a composition of isWith and isIn

- Available online
Structure of the tutorial

- **9.00 - 10.30**
  - Stream Reasoning introduction (30 min)
  - RDF stream processing models (60 min)

- **11.00 – 12.30**
  - An overview of Stream Reasoning (30 min)
  - C-SPARQL Engine: A RDF Stream Processing system for the Continuous Extension of SPARQL (C-SPARQL) with Naive Stream Reasoning support (30m)
  - MorphStream: Ontology-based streaming data access (30m)

- **13:45 - 15.30**
  - Hands on session (120min)
    - From Web tools to JavaCode in Eclipse

- **16:00 - 17.30**
  - IMaRS: Incremental Materialization for RDF Streams (30m)
  - Other Stream Reasoning approaches (30 min)
  - Wrap-up and conclusions (30 min)
Water, water, every where,
Nor any drop to drink.
-- The Rime of the Ancient Mariner
Samuel Taylor Coleridge, 1798

Streams, streams everywhere
nor any actionable fact to use
-- Emanuele and Daniele, 2013 :-P
Stream Reasoning
For Linked Data
M. Balduini, J-P Calbimonte, O. Corcho, D. Dell'Aglio, and E. Della Valle
http://streamreasoning.org/events/sr4ld2014

Stream Reasoning introduction
Emanuele Della Valle
eMANUELE.DELLAVALLE@POLIMI.IT
http://emanueledellavalle.org