Towards
Ontology Based Event Processing

RISE SICS, Electrum Kista Stockholm, Sweden

R. Tommasini - Politecnico di Milano

ME

PhD Student @ Politecnico di Milano

Research Interests:

-Semantic Web & Reasoning

-**Stream** Processing

-Programming **Languages**

-**Distributed** Systems

@rictomm

rictomm.me

rictomm@gmail.com

My Advisor

- Assistant Professor at DEIB Politecnico di Milano
- Expert in semantic technologies and stream computing
- Brander of **stream reasoning**
- 17 years of experience in research and innovation projects
- Startupper: http://www.fluxedo.com

emanuele.dellavalle@polimi.it
@manudellavalle
http://emanueledellavalle.org
http://streamreasoning.org
http://fluxedo.com

What is Stream Reasoning?

Can we detect fire?

*Expected Answer: **YES**

Can we (actually) detect fire?

Expected Reaction: **Perplexed Audience**


SMOKE ALARM


SMOKE ALARM


100
°
70°
20°


100
°
70°
20°


70%


30%

Summary

Workarounds

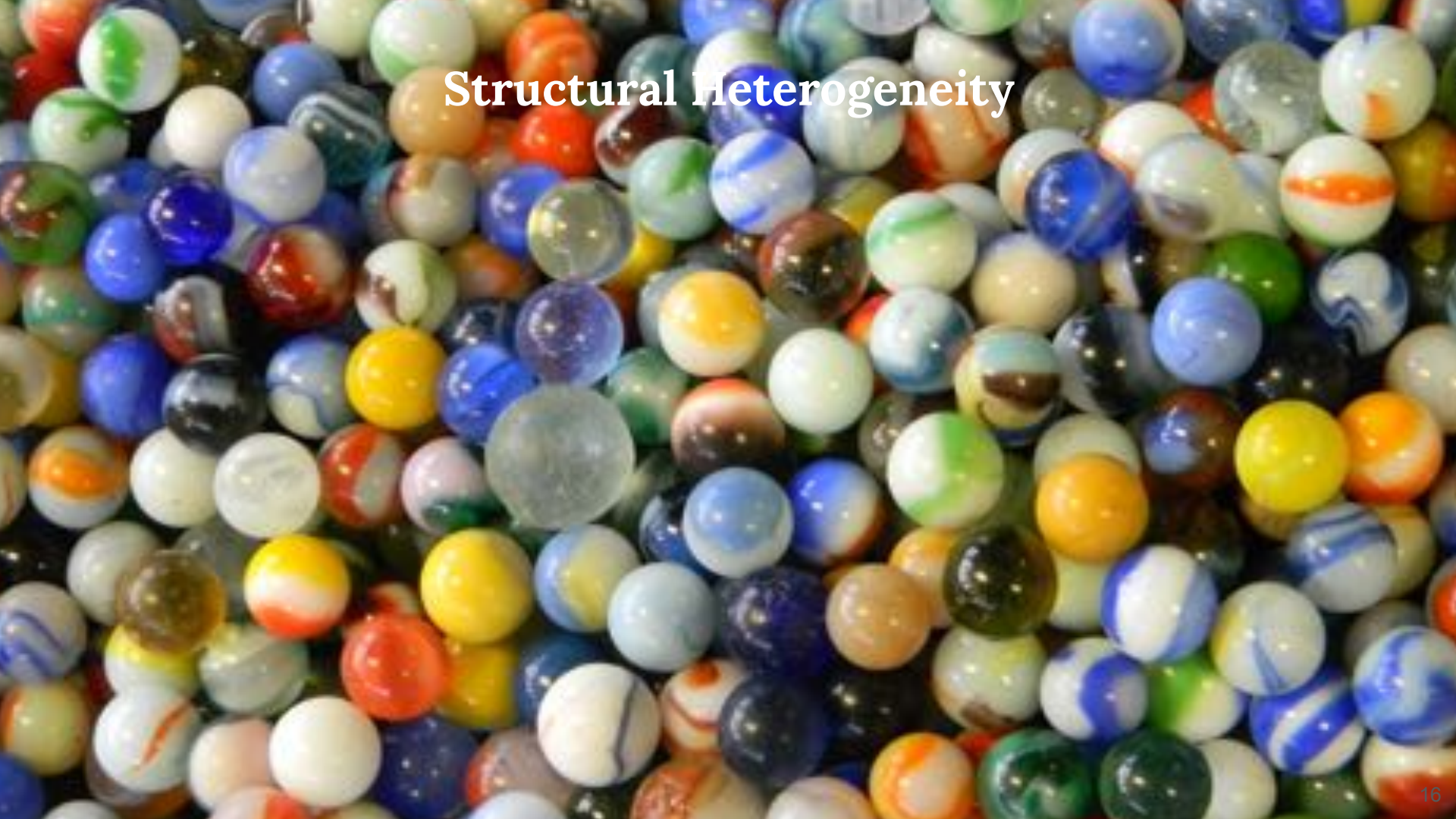
Smoke Detection

Humidity Variations (decreases)

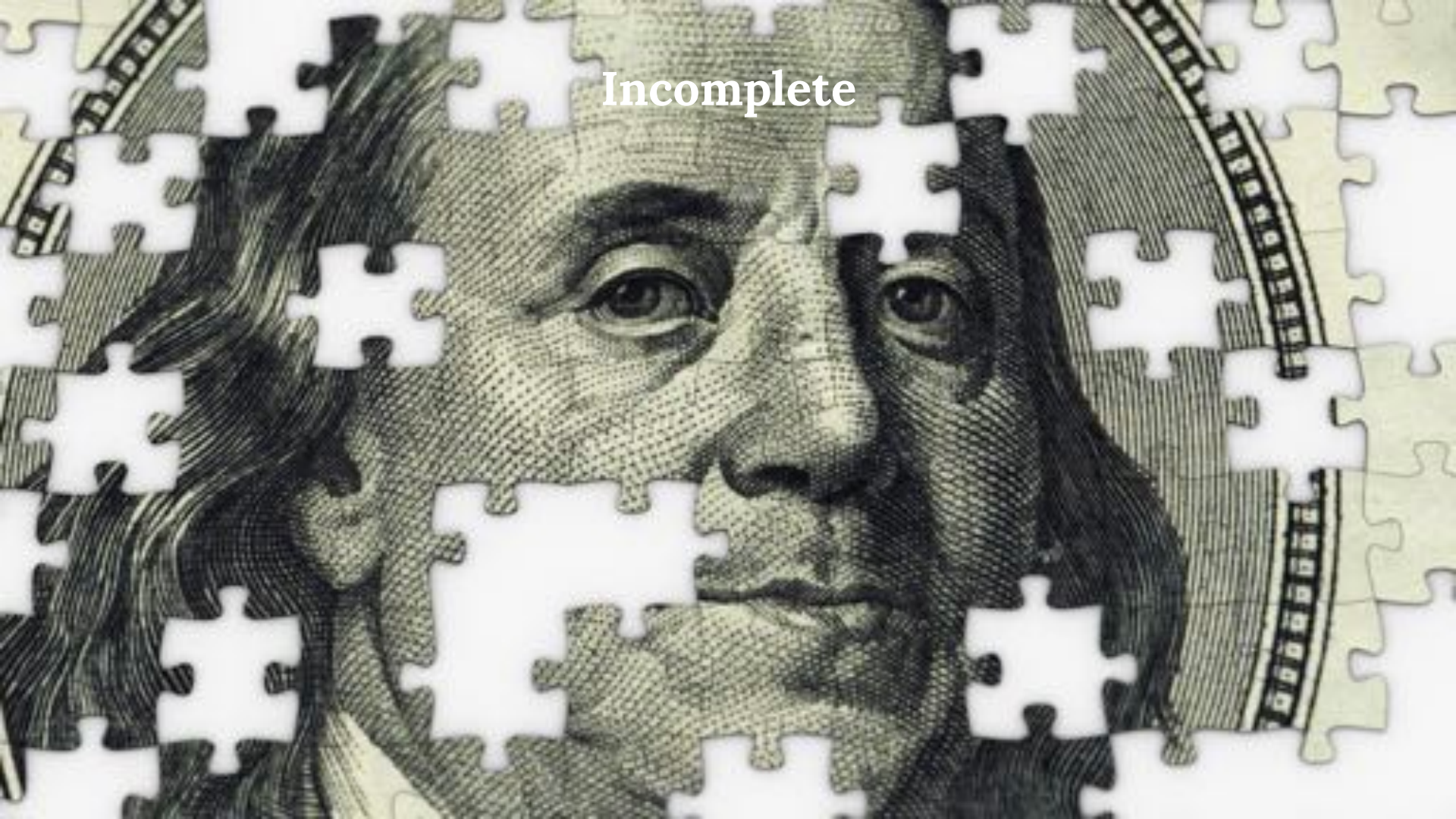
Temperature Variations (increases)

This is Stream Reasoning!

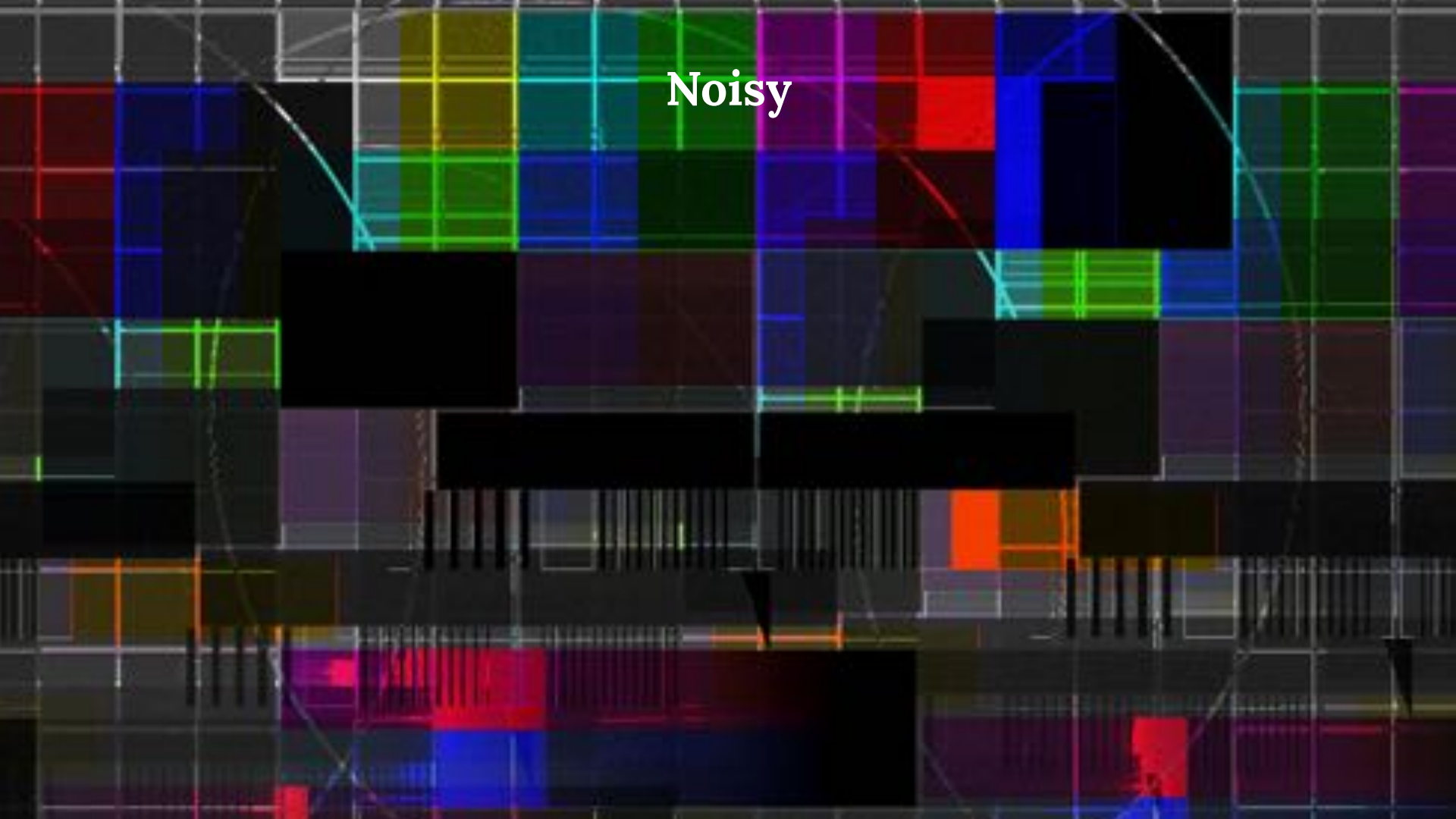
Structural Heterogeneity



Semantic Heterogeneity


Incomplete


Vast


Noisy

Complex Domain

Can we **make sense** in **real-time** of **heterogeneous**, **vast**, **incomplete**, and inevitably **noisy** and **data streams** in order to support the decision processes of extremely large numbers of **concurrent users**?

Requirement Analysis

- handle **massive** datasets
- process **data streams**
- cope with **heterogeneous** data
- cope with **incomplete** data
- cope with **noisy** data
- provide **reactive** answers
- access **fine-grained** information
- model **complex** domains

Volume	Velocity	Variety	Veracity
x			
	x		
		x	
		x	x
			x
		x	
		x	x
		x	

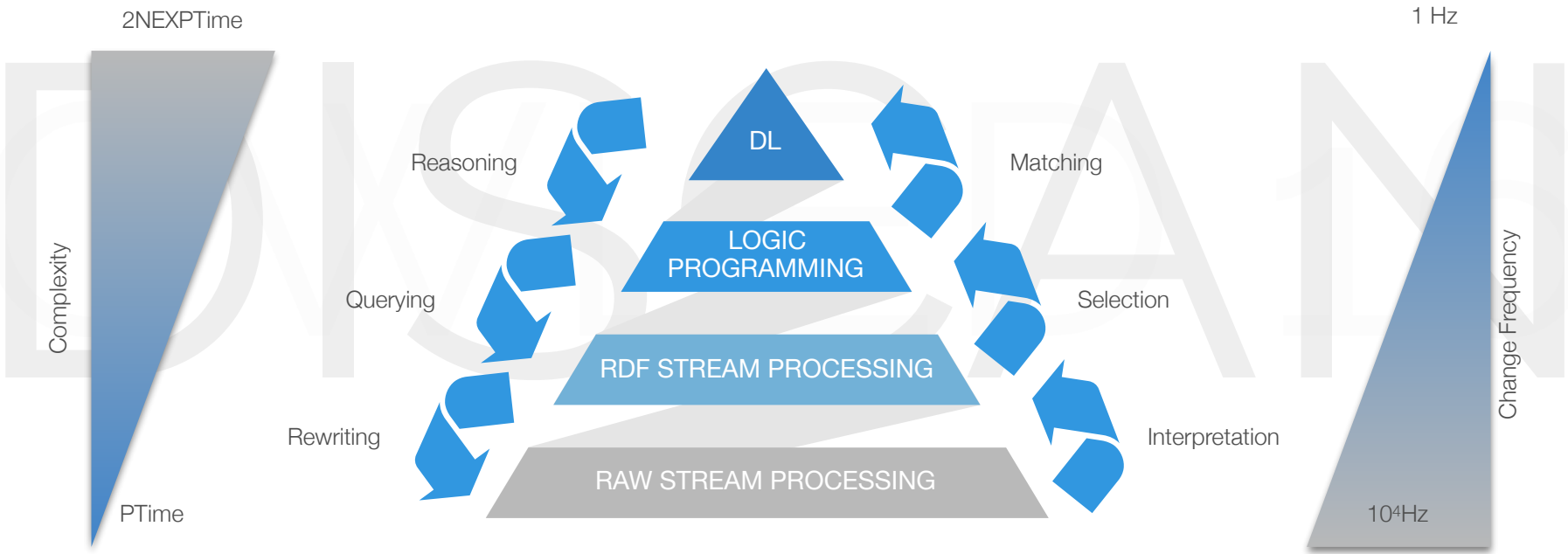
Stream Processing vs Semantic Technologies

Requirement	SP	ST
massive datasets	✓	
data **streams**	✓	
heterogeneous dataset		✓
incomplete data		✓
noisy data		✓
reactive answers	✓	
fine-grained information **access**	✓	✓
complex domain **models**		✓

Stream Reasoning

Stuckenschmidt, H., Ceri, S., Della Valle, E., & Van Harmelen, F. (2010).
Towards expressive stream reasoning

Cascading Reasoning



Stuckenschmidt, H., Ceri, S., Della Valle, E., & Van Harmelen, F. (2010). Towards expressive stream reasoning

RDF Stream Processing (RSP)

Continuous Data Integration

RDF Streams

-An **RDF Stream** is an **partially** ordered sequence of pairs (G_i, t_i) where

-**Gi**, is a [named] **RDF graph** and

-**ti** is a timestamp.

An Example

```
( { :s1 :observes :o1 ; :o1 :value 20C }, 1)
( { :s1 :observes :o2 ; :o2 :value 20C }, 2)
( { :s1 :observes :o3 ; :o3 :value 30C }, 3)
( { :s1 :observes :o4 ; :o4 :value 50C }, 4)
```

RSEP-QL

- A Reference **Model** fo Continuous **SPARQL**
- Extends **CQL** to process RDF Graphs
- Introduces the notions of **Window** and **Event Pattern**

An Example

```
REGISTER STREAM <fire>
CONSTRUCT { ?o a :FireObservation ; :sensedBy ?s .}
FROM NAMED WINDOW <w1> [RANGE 5m,STEP 5m] ON STREAM <temp>
WHERE { WINDOW <w1> {
  ?s :observes ?o ; ?o :value ?t
  FILTER (?t > 50C) }}
```

Continuous Reasoning

Deductive

Ontology Streams

-An Ontology Stream is an **partially** ordered sequence of pairs (A_i, t_i) where

-**A_i**, is a set of a **ABox** axioms w.r.t. a static **TBox** T.

-**t_i** is a timestamp.

Windowed Ontology Streams

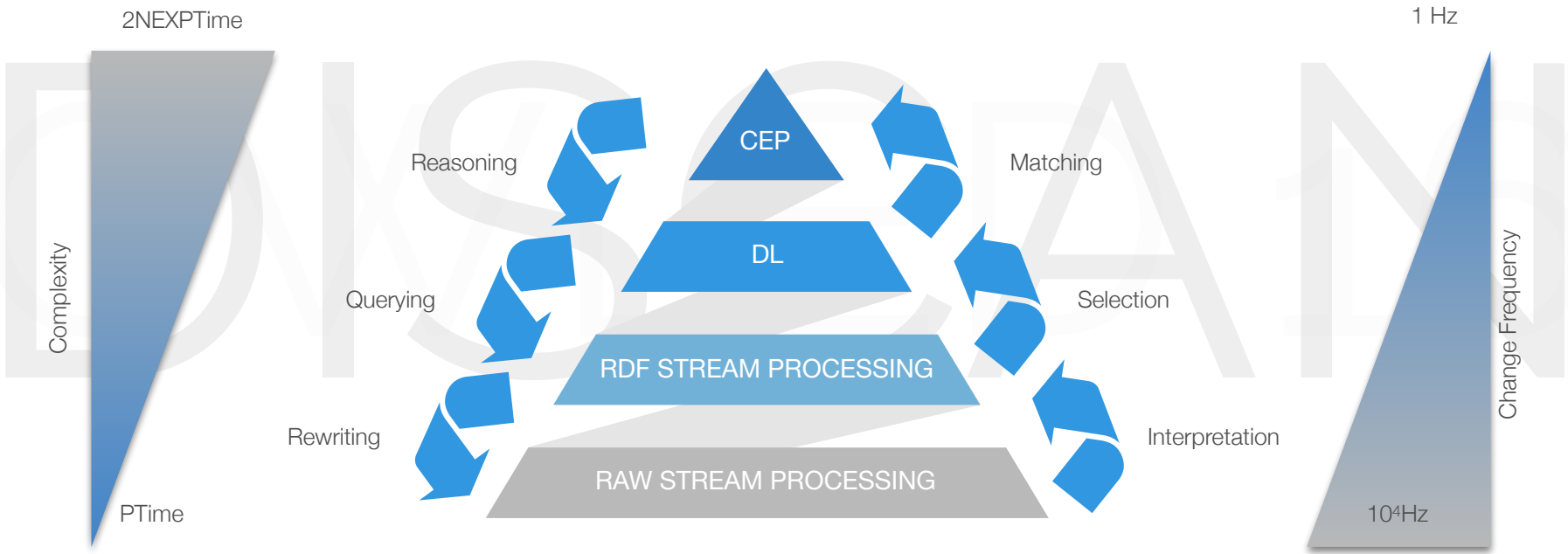
-An Windowed Ontology Stream $S_{[o,c]}$ is the union of all the **Abox** axioms Sets A_i with o<i<c

-**Continuous Reasoning** can be reduced to traditional **ontological reasoning** over a windowed ontology stream

Ontology Based Event Processing

Joint work with P.Bonte, E. Mannens, F. De Turck, F. Ongenae

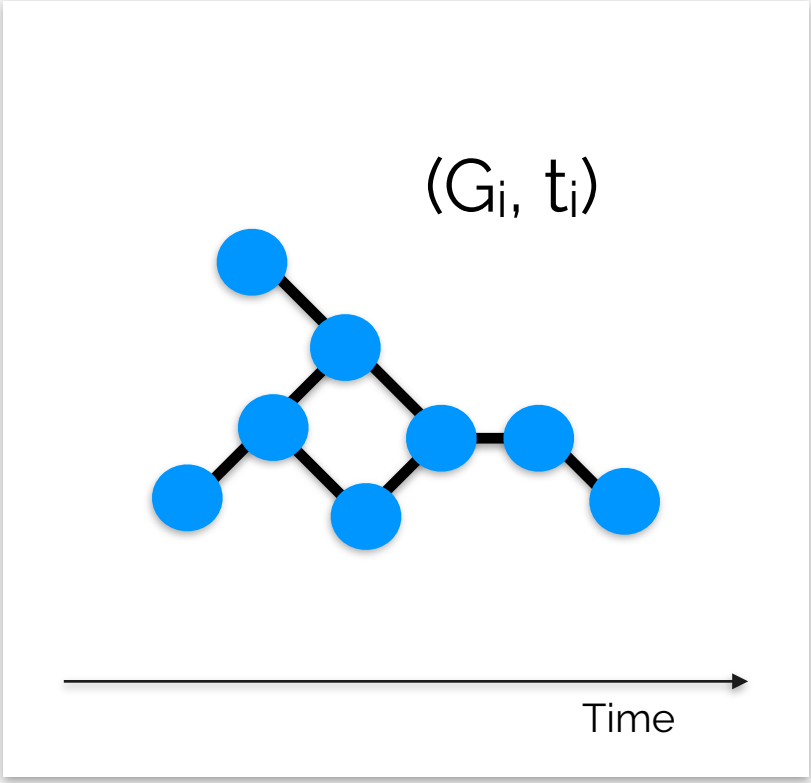
Cascading Reasoning Approach



Stuckenschmidt, H., Ceri, S., Della Valle, E., & Van Harmelen, F. (2010). Towards expressive stream reasoning

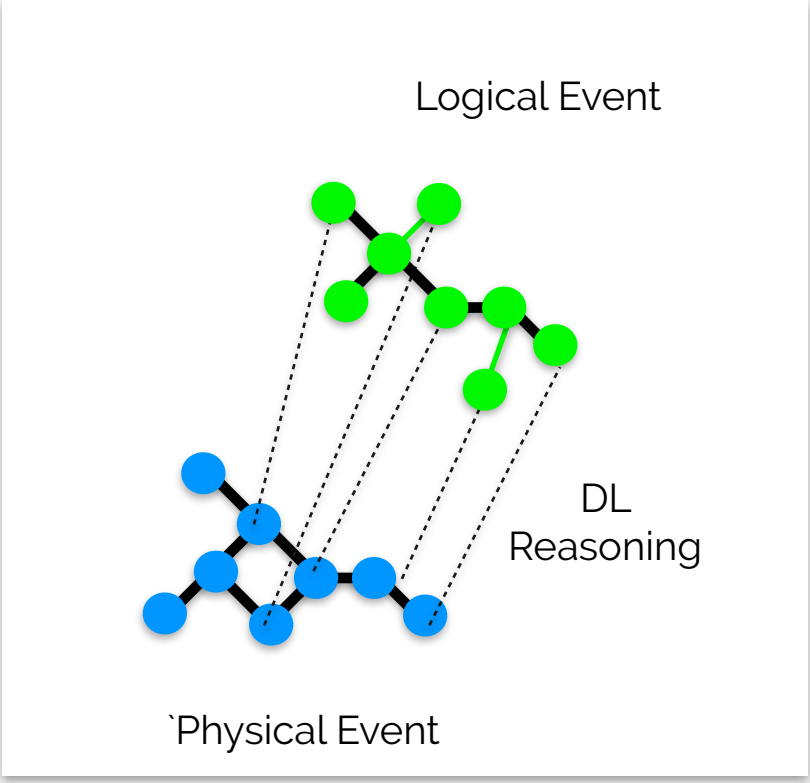
Data Integration

We assume **RDF Stream** as common data model



Events!

first-class objects in the language



```
EVENT OfficeTemperaturEvent
subClassOf TemperaturEvent
and (observationResult some
(hasValue >= 40)) and
(hasLocation some Office)
```

Logical Modeling

Logical Event Specifications

```
EVENT FireEvent {
MATCH TemperaturEvent
SEQ SmokeDetectionEvent
WITHIN (5m) }
```

Semantic Complex Event Processing

Patterns

```
EVENT FireEvent {
MATCH TemperaturEvent
SEQ SmokeDetectionEvent
WITHIN (5m)
IF {
   EVENT TemperaturEvent
             {?loc0 hasValue ?v}
   EVENT SmokeDetectionEvent
            {?loc1 hasValue ?v
    FILTER (?smokeLevel == 3) }}
```

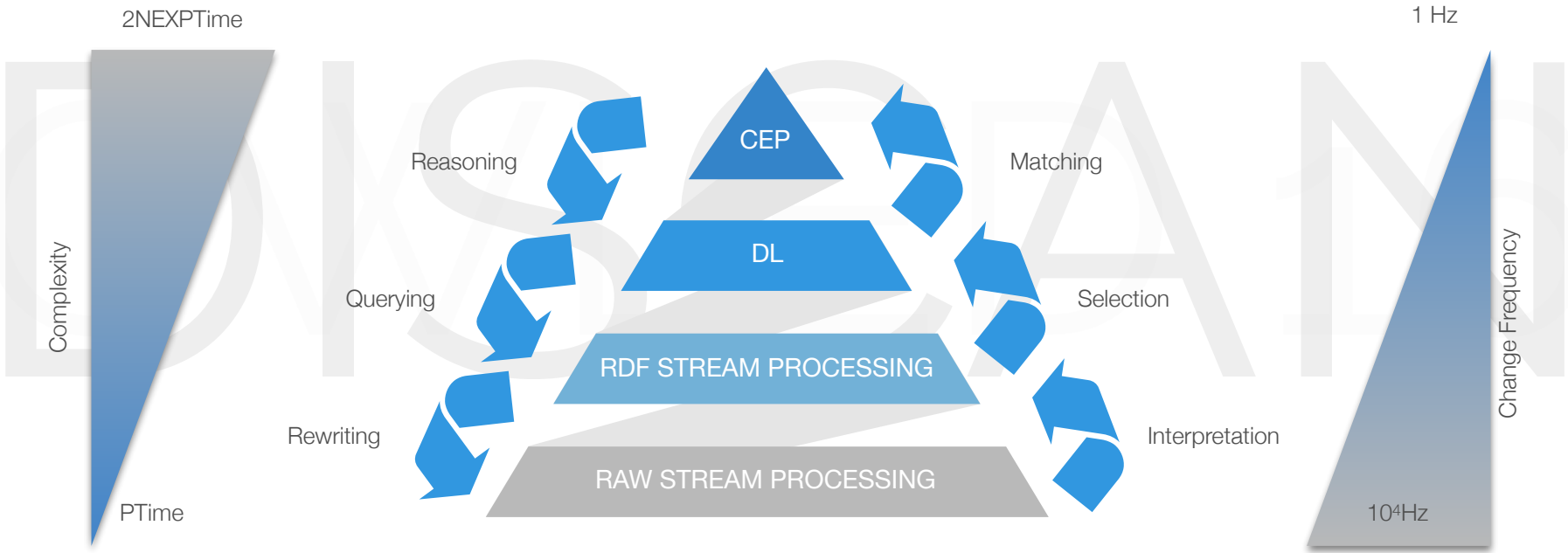
Semantic Complex Event Processing

In OBEP

Future Works

Ontology Based Streaming Data Access

Cascading Reasoning



Stuckenschmidt, H., Ceri, S., Della Valle, E., & Van Harmelen, F. (2010). Towards expressive stream reasoning

Rewriting and Interpreting

RDF
STREAM PROCESSING

RAW
STREAM PROCESSING

- including **continuous semantics** will **enable** continuous querying over **virtual streaming sources**;
- including **time** operators like windows will **enable query rewriting** into **continuous query languages**

Stream Reasoning Applications

Anatomy of a **Streaming Application**

- **Input** Streams
- **Output** Streams
- **Continuous** Tasks

WASP

Web Stream Processing Application

The Web is Streaming

VoCaLS - Vocabulary and Catalog for Linked Streams

- VOCALS allows to **describe streams** and streaming endpoints in a machine readable form

- VOCALS enables **stream services description**, fostering interoperability between **producers** and **consumers**.

- VOCALS let track **stream transformation provenance** describing the continuous tasks operating on streams.

Questions?

Email: riccardo.tommasini@polimi.it
Twitter: @rictomm
Github: riccardotommasini
Web1: riccardotommasini.com
Web2: streamreasoning.org