Centre for
Data Analytics

kbs
Knowledge-Based
Systems Group

Insight

Towards Enriching CQELS with Complex Event Processing and Path Navigation

1st Workshop on High-Level Declarative
Stream Processing
September 22, 2015

Minh Dao Tran and Danh Le Phuoc

IRISH RESEARCH COUNCIL
An Chomhairle um Thaighde in Éirinn

TU WIEN
FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

DCU

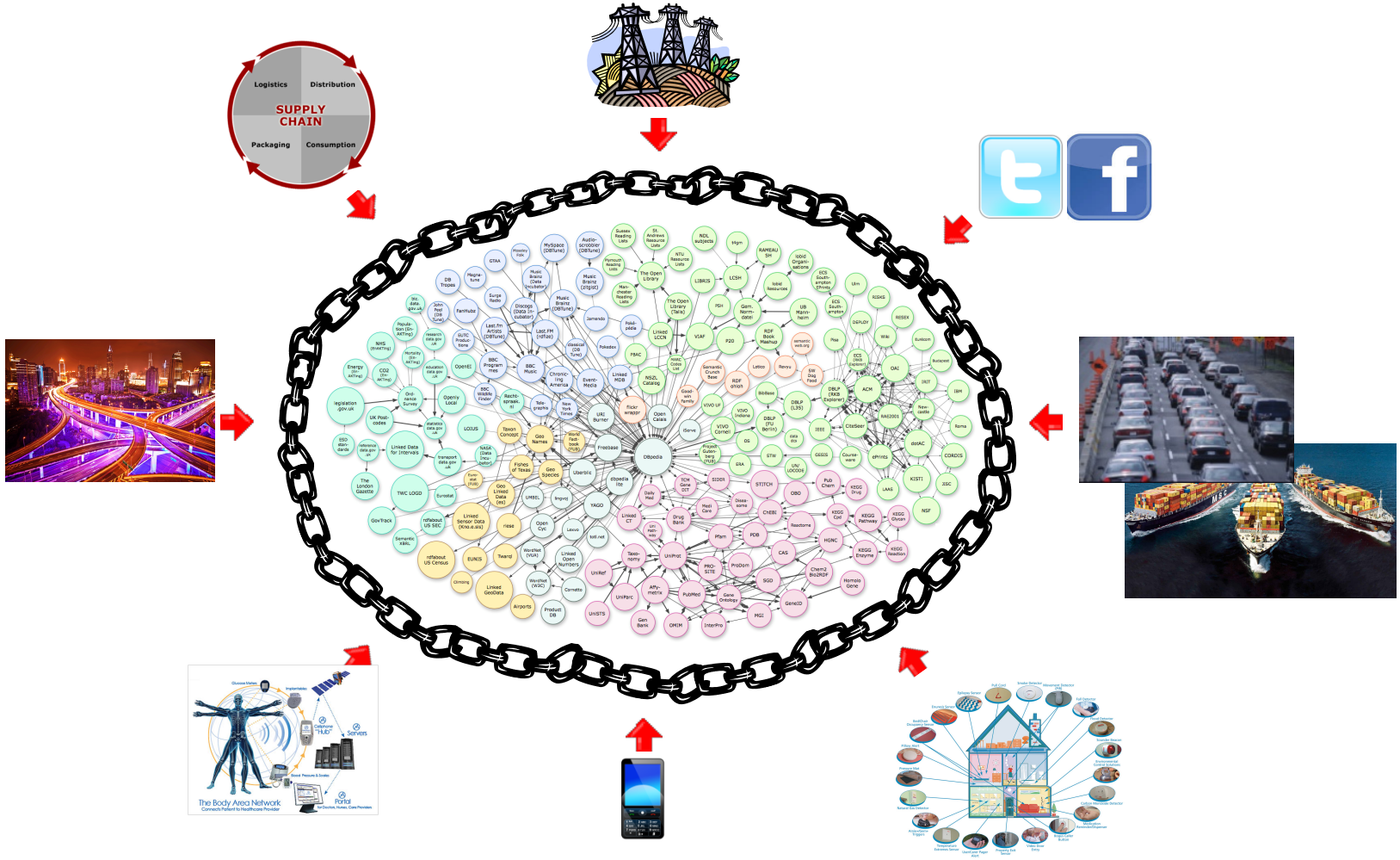
NUI Galway
OÉ Gaillimh

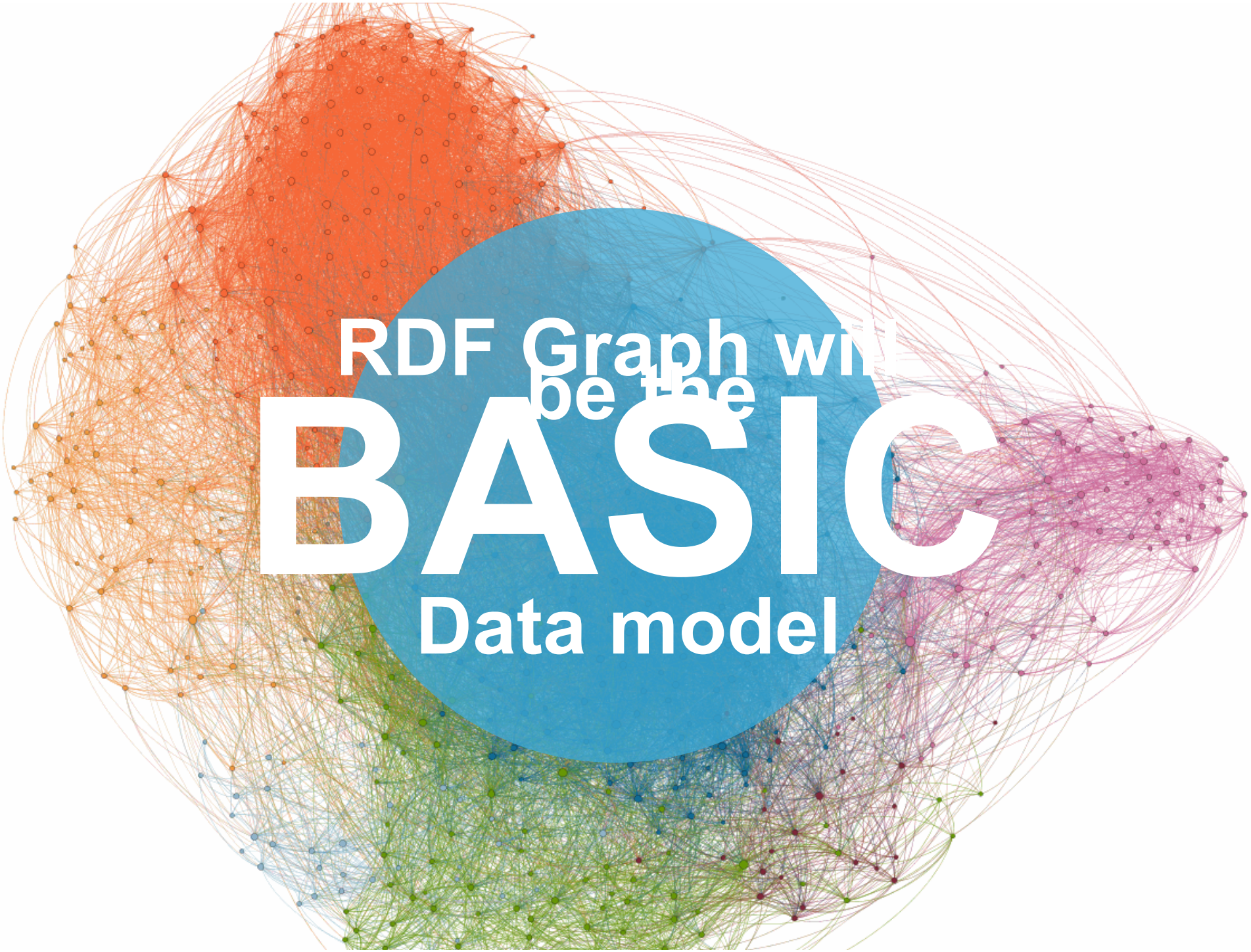
UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

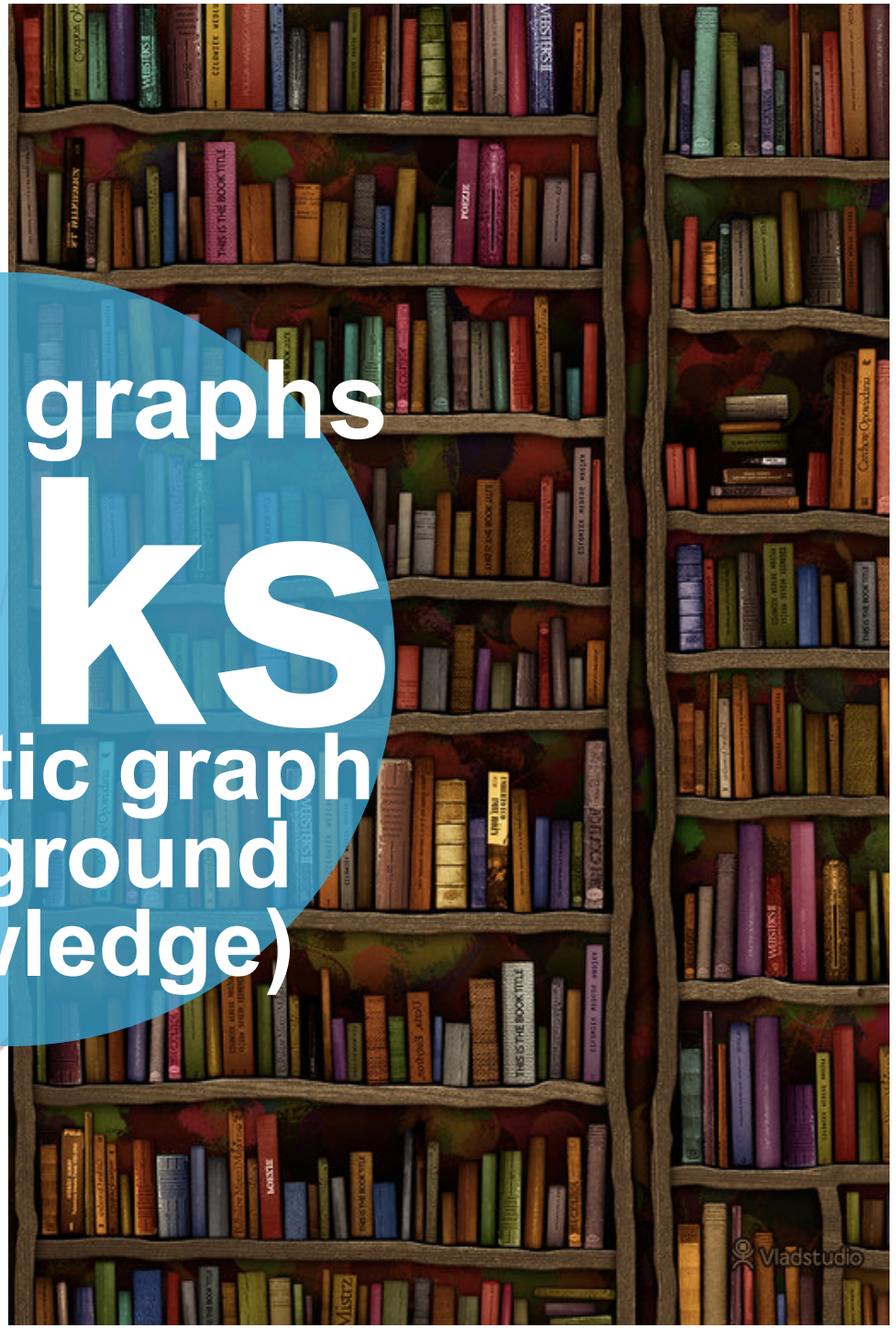
UCD
DUBLIN

SFI
Science Foundation Ireland

The Quest for Unified Data Representation !!!




RDF Graph wi
be the
BASIC
Data model

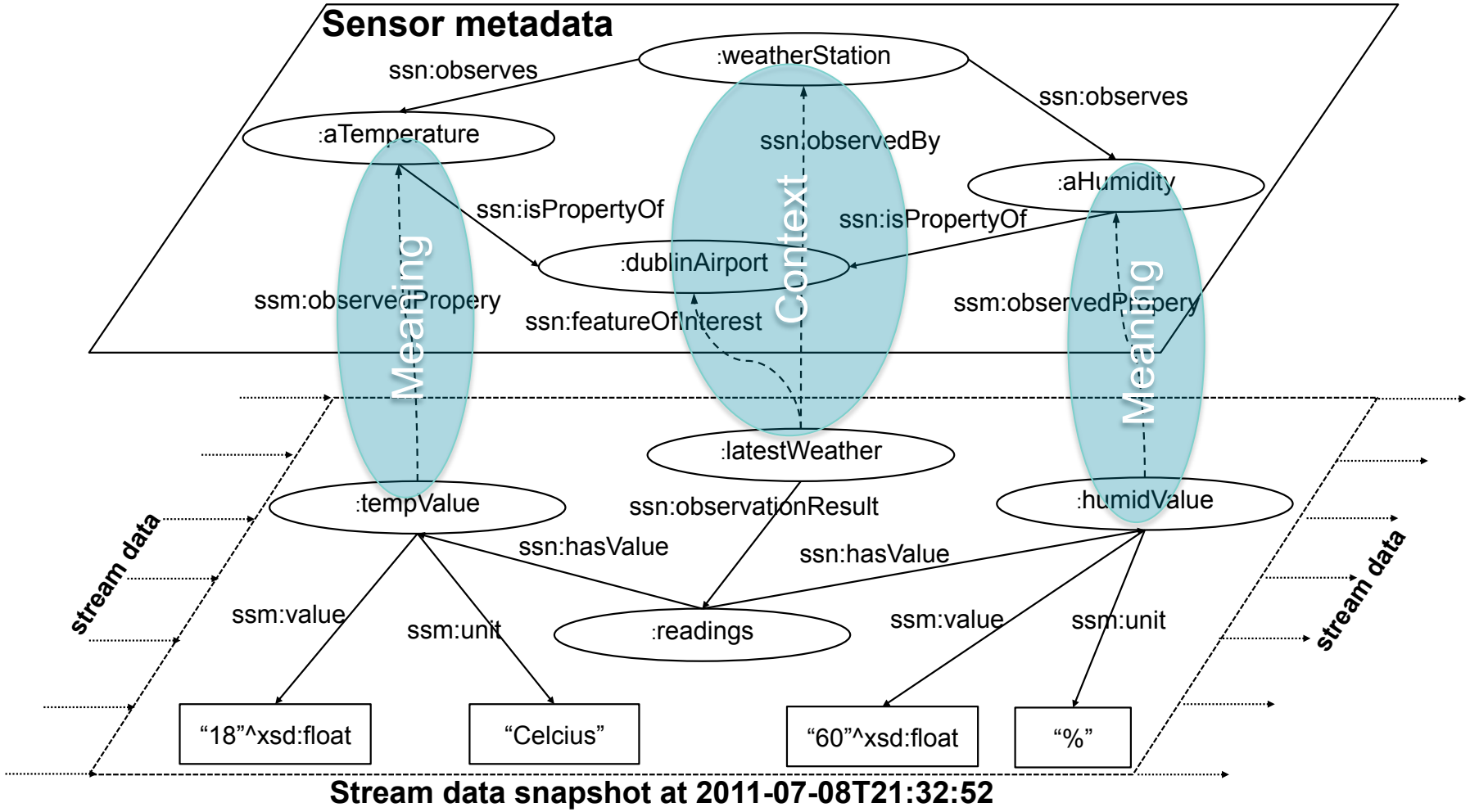


tream graphs

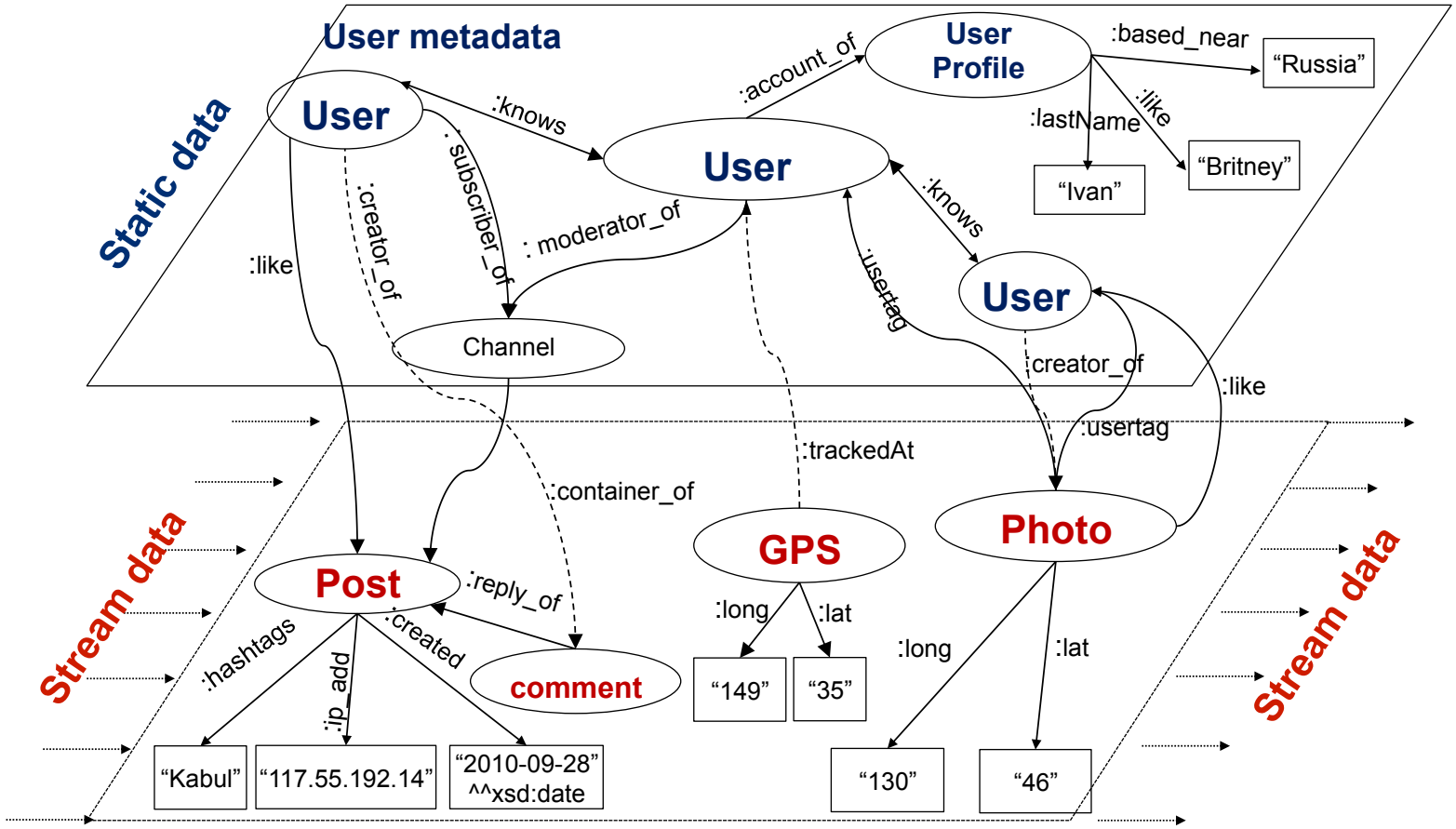
links

with static graph (Background knowledge)

RDF Stream Data layout



RDF Stream Data Layout (cont.)



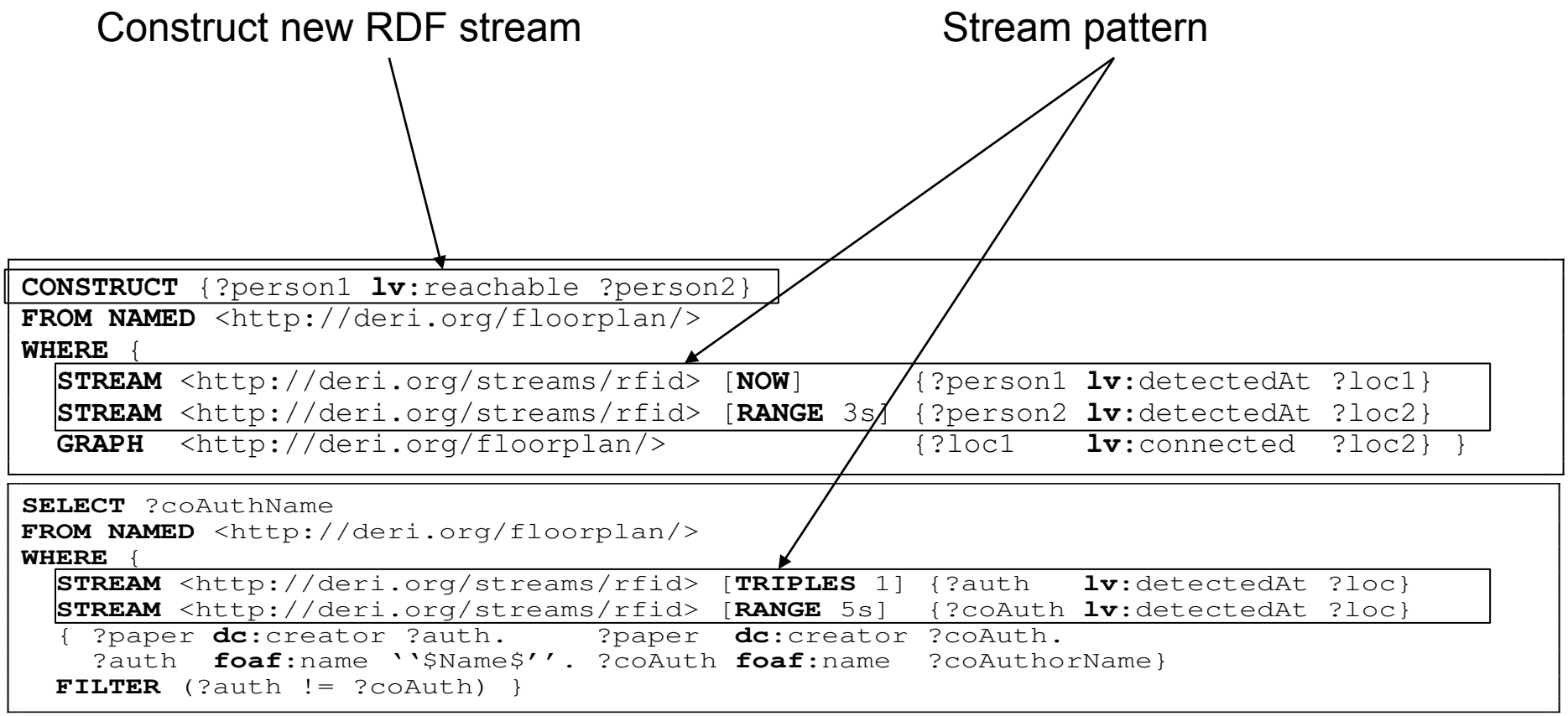
Social network data stream in RDF

Stream data: GPS, Posts & comments, Photos

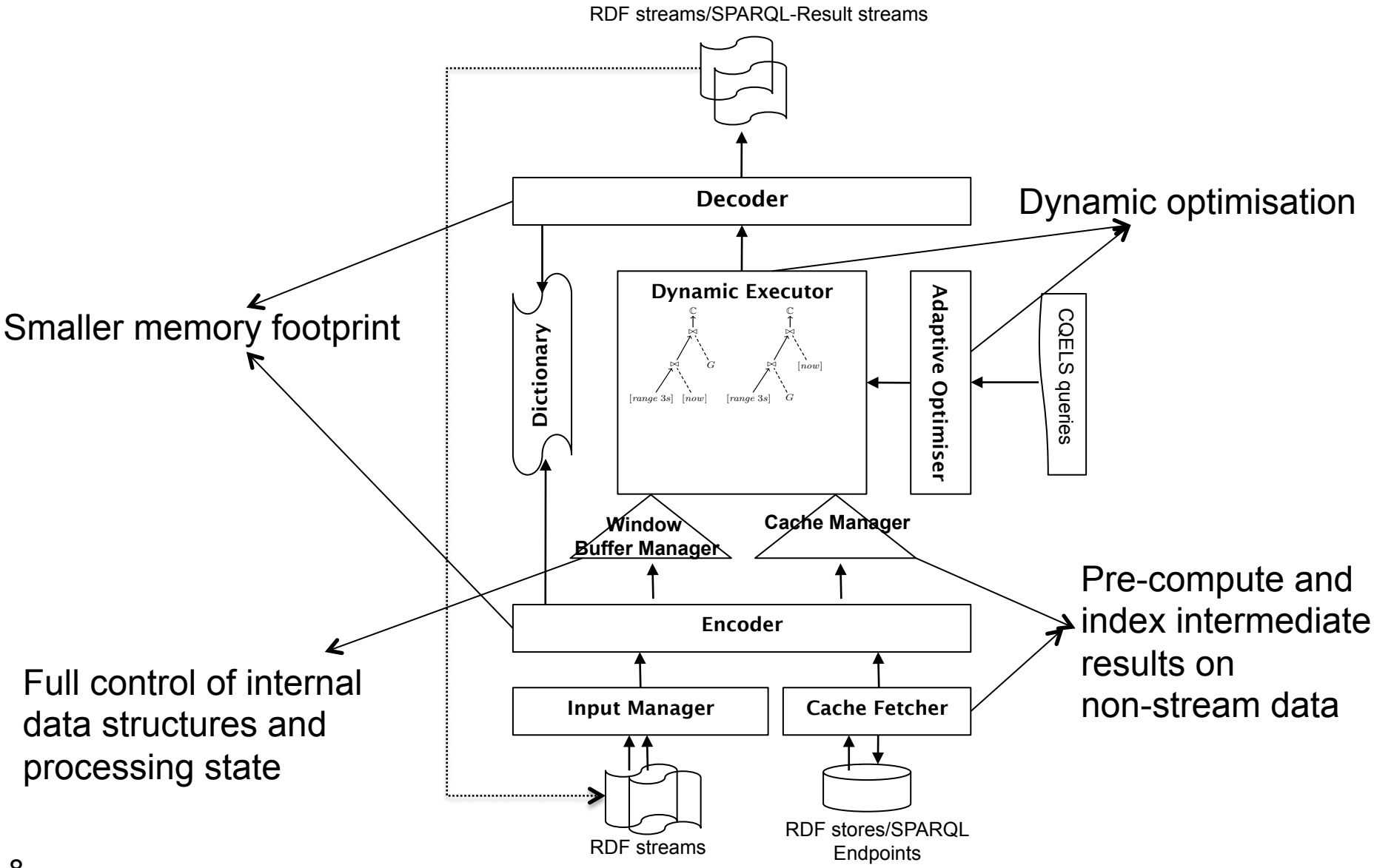
Static data: User metadata (user profile, users' relationships)

Continuous Query Execution over Linked Stream -CQELS

CQELS Language (CQELS-QL) – an extension to SPARQL 1.1

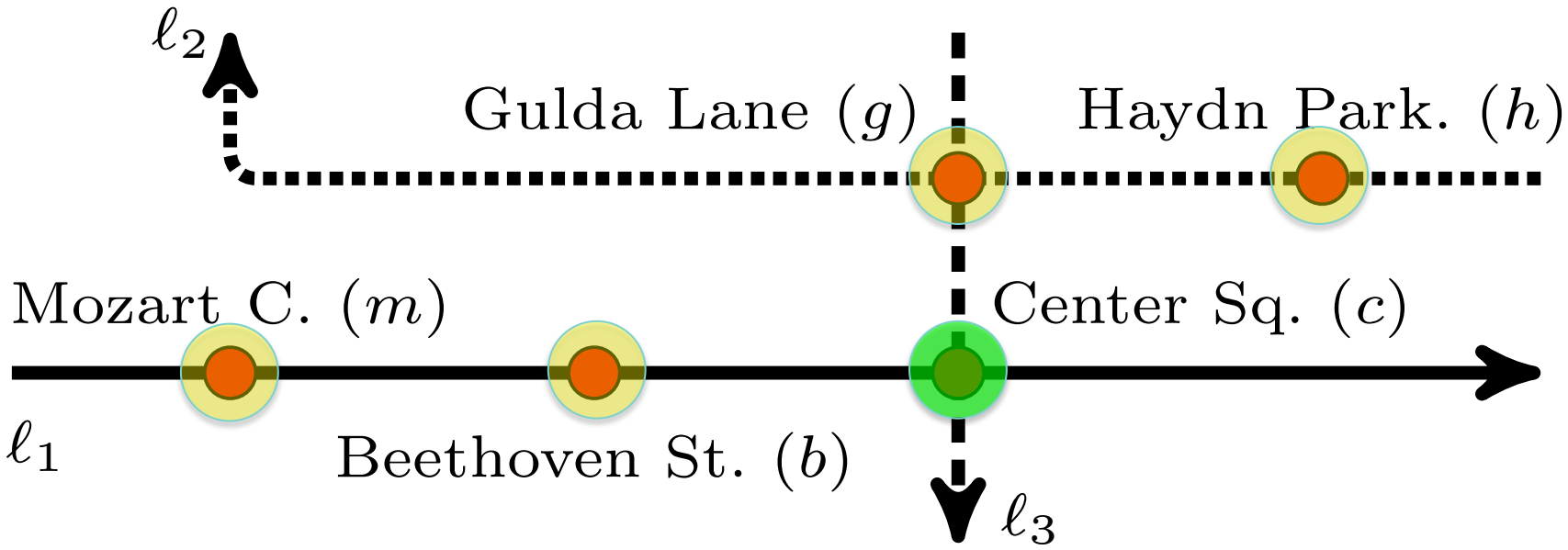


CQELS Execution Framework

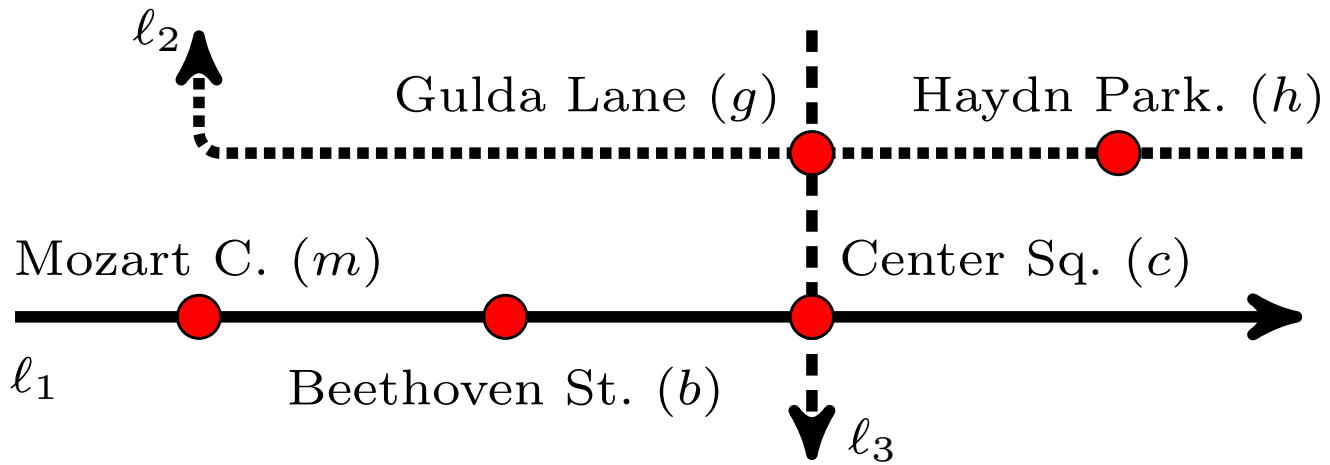


Complex Event Detection: Linked/ Connected Event

"notify about recent *repeated delays* of subways *following* by *no arrival* at stops that according to the network plan *can be reached by subways*"

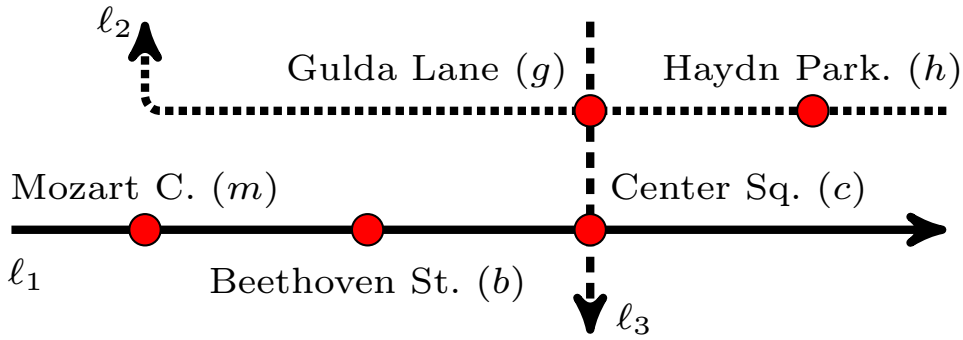


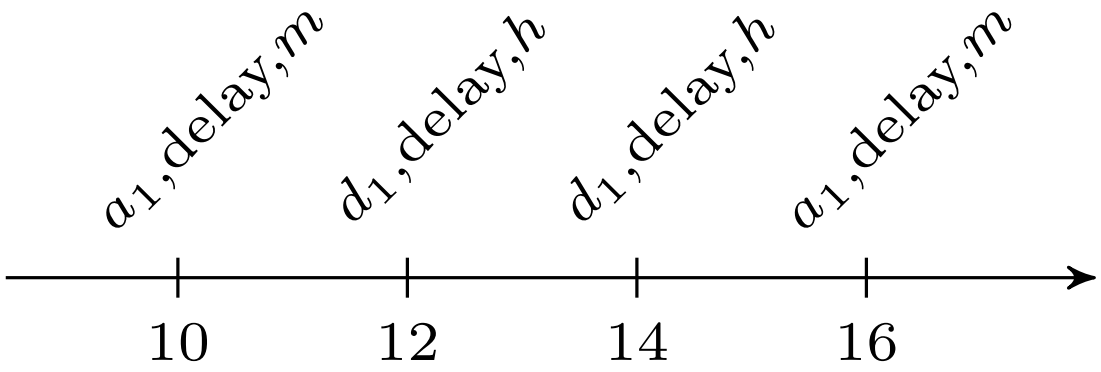
Network plan in RDF



$$D = \left\{ \begin{array}{llll} \texttt{:conn}_1\ \texttt{:beg :m}, & \texttt{:conn}_1\ \texttt{:end :b}, & \texttt{:conn}_1\ \texttt{:means :subway}, & \texttt{:conn}_1\ \texttt{:dur :3m}, \\ \texttt{:conn}_2\ \texttt{:beg :b}, & \texttt{:conn}_2\ \texttt{:end :c}, & \texttt{:conn}_2\ \texttt{:means :subway}, & \texttt{:conn}_2\ \texttt{:dur :2m}, \\ \texttt{:conn}_3\ \texttt{:beg :h}, & \texttt{:conn}_3\ \texttt{:end :g}, & \texttt{:conn}_3\ \texttt{:means :subway}, & \texttt{:conn}_3\ \texttt{:dur :3m}, \\ \texttt{:conn}_4\ \texttt{:beg :g}, & \texttt{:conn}_4\ \texttt{:end :c}, & \texttt{:conn}_4\ \texttt{:means :tram}, & \texttt{:conn}_4\ \texttt{:dur :5m}, \\ & & \vdots & \\ \texttt{:a}_1\ \texttt{rdf:type :subway}, & & \texttt{:d}_1\ \texttt{rdf:type :subway}, & \\ & & \vdots & \end{array} \right\}$$

RDF Stream





$$S = \begin{array}{ll} \langle\{(\mathtt{a}_1, \mathtt{delay}, \mathtt{m})\}, [10, 10]\rangle, & \langle\{(\mathtt{d}_1, \mathtt{delay}, \mathtt{h})\}, [12, 12]\rangle, \\ \langle\{(\mathtt{d}_1, \mathtt{delay}, \mathtt{h})\}, [14, 14]\rangle, & \langle\{(\mathtt{a}_1, \mathtt{delay}, \mathtt{m})\}, [16, 16]\rangle, \ldots \end{array}$$

CQELS-QL limitations

Can not express Event ordering

```
1 SELECT      ?s
2 FROM        ex:transportationMap
3 FROM NAMED WINDOW :W ON ex:publicTransport [RANGE 10m]
4 WHERE {
5   WINDOW :W { ?v :delayAt ?s }
6   ?v rdf:type :subway.
7 }
```

Can not express graph path/networked connection

nSPARQL path navigation operators

$$[\![\texttt{self}::a]\!]_G = \{(a,a)\}$$

$$[\![\texttt{next}::a]\!]_G = \{(x,y) \mid \exists z\colon (x,z,y) \in G\}$$

nSPARQL operators

$$[\![axis^{-1}::a]\!]_G = \{(x,y) \mid (y,x) \in [\![axis::a]\!]_G\}, \text{ where } axis \in \{\texttt{next}, \texttt{node}, \texttt{edge}\}$$

SPARQL 1.1 grammars

$$exp ::= axis \mid axis :: a (a \in IBL) \mid axis :: [exp] \mid exp/exp \mid exp|exp \mid exp^* \mid exp^+$$

SPARQL 1.1 syntaxes

```
?s (next-1::beg[next::means/self::subway]/next::end)+ :c.
```

Sequence operators for CEP

$$\mathrm{SEQ}(A_1, \ldots, A_n) \equiv \exists\, t_1 < t_2 < \ldots < t_n \colon A_1(t_1) \wedge A_2(t_2) \wedge \ldots \wedge A_n(t_n).$$

$$\mathrm{SEQ}(A_1, \ldots, A_{i-1}, !A_i, A_{i+1}, \ldots, A_n) \equiv$$
$$\exists t_1 < \ldots < t_{i-1} < t_{i+1} < \ldots < t_n \colon A_1(t_1) \wedge \ldots \wedge A_{i-1}(t_{i-1}) \wedge A_{i+1}(t_{i+1}) \wedge \ldots \wedge A_n(t_n)$$
$$\wedge\, (\forall t_i \in (t_{i-1}, t_{i+1}) \colon \neg A_i(t_i)).$$

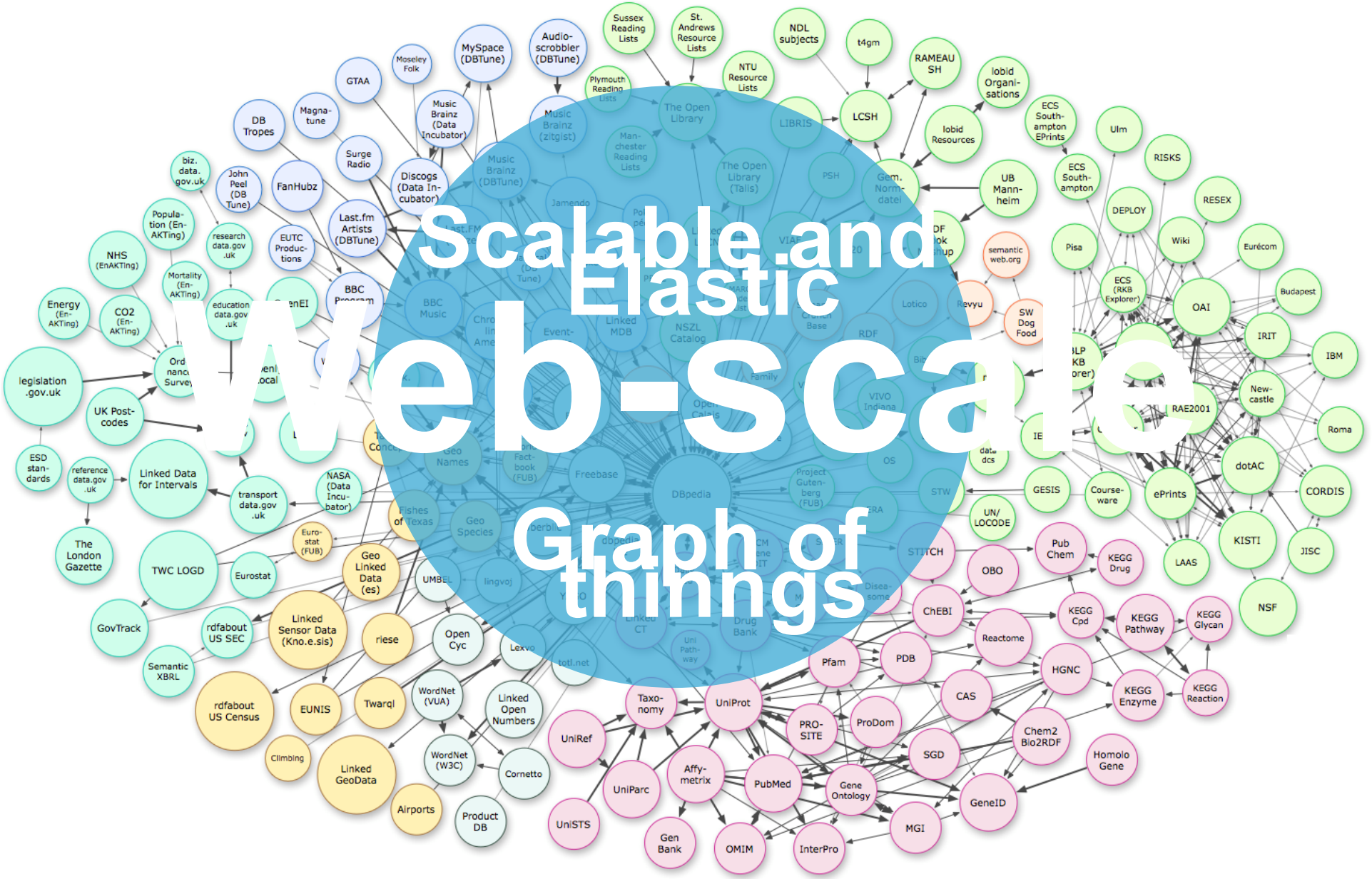
```
FROM NAMED WINDOW :W ON ex:publicTransport [RANGE 10m]
WINDOW :W {
 SEQ({?v :delayAt ?s}, {?v :delayAt ?s}, !{?v :arriveAt ?s})
}
```

Put all together

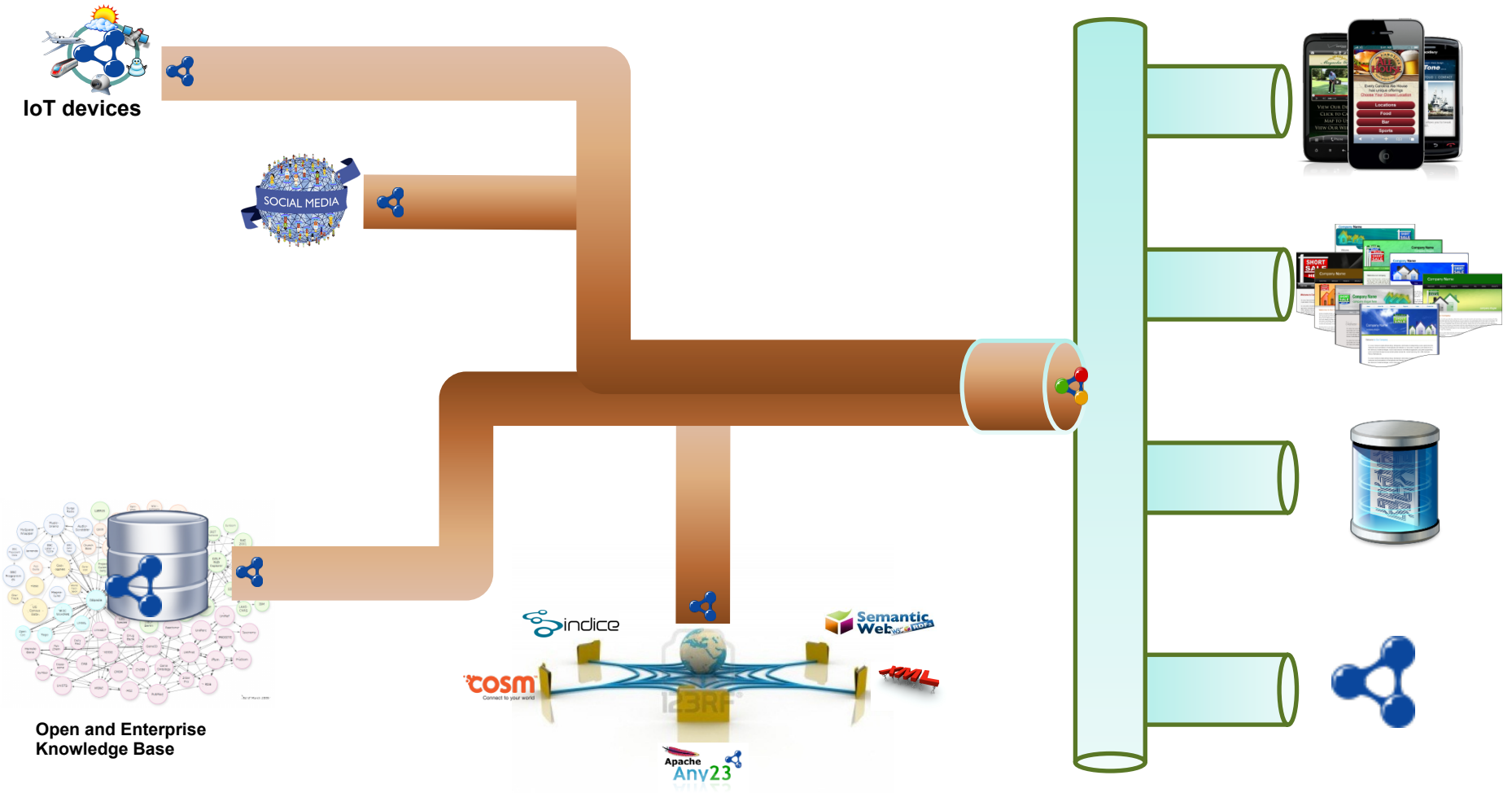
```
1  SELECT      ?s
2  FROM        ex:transportationMap
3  FROM NAMED WINDOW :W ON ex:publicTransport [RANGE 10m]
4  WHERE {
5   WINDOW :W {
6    SEQ({?v :delayAt ?s}, {?v :delayAt ?s}, !{?v :arriveAt ?s})
7   }
8   ?v rdf:type :subway.
9   ?s (^:beg/[:means :subway]/:end)+ :c.
10 }
```

Why nSPARQL and SEQ for CQELS engine implementations?

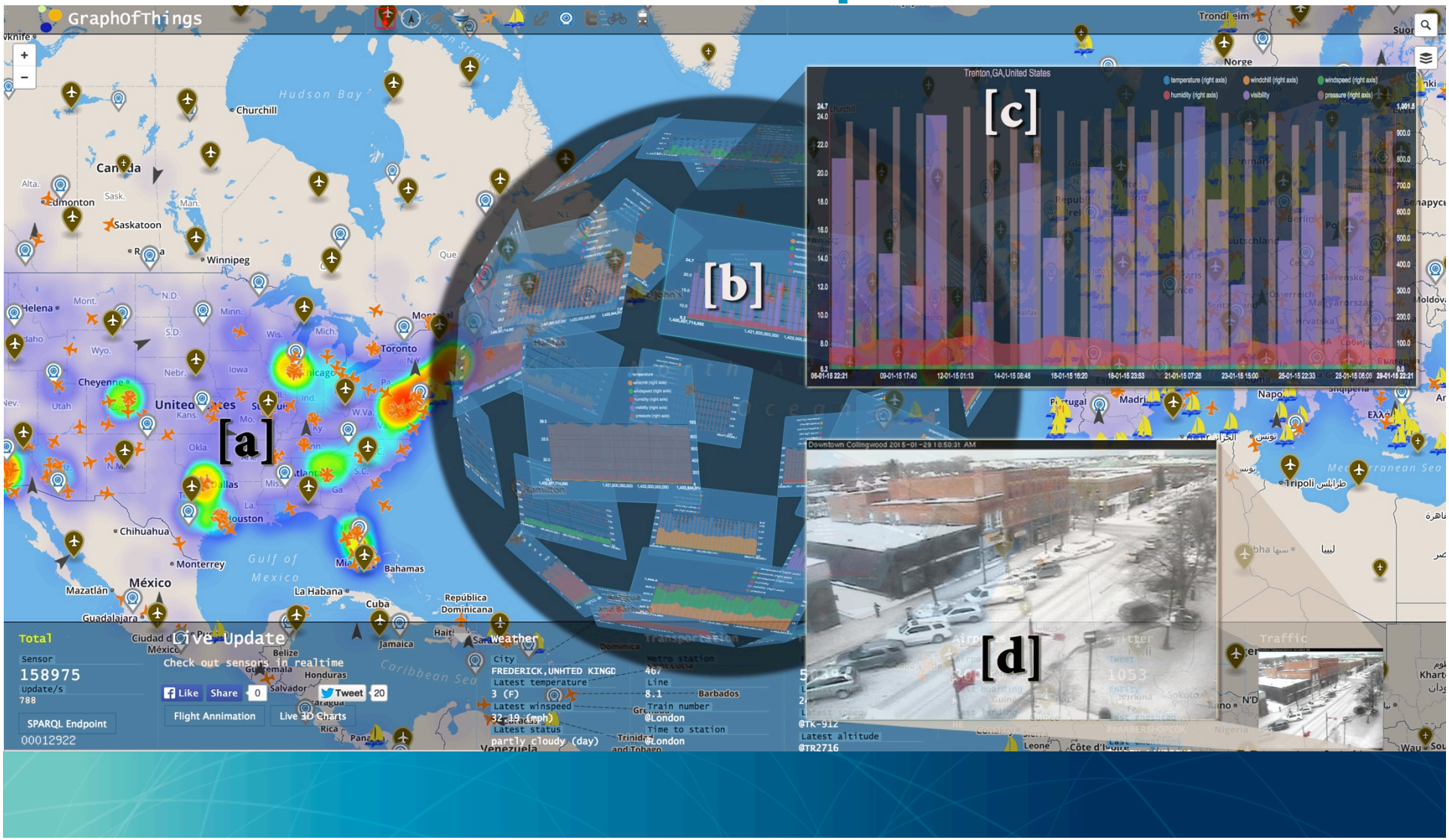
- **New attractive features to CQELS-QL:**
  - Path Navigation through the RDF graph/Stream
  - Support RDFS reasoning
  - Support SEQ for CEP
- **Efficient Implementation**
  - Polynomial
  - Efficient algorithms


Scalable and Elastic
Web-scale
Graph of thinngs

Unified Data Integration Bus

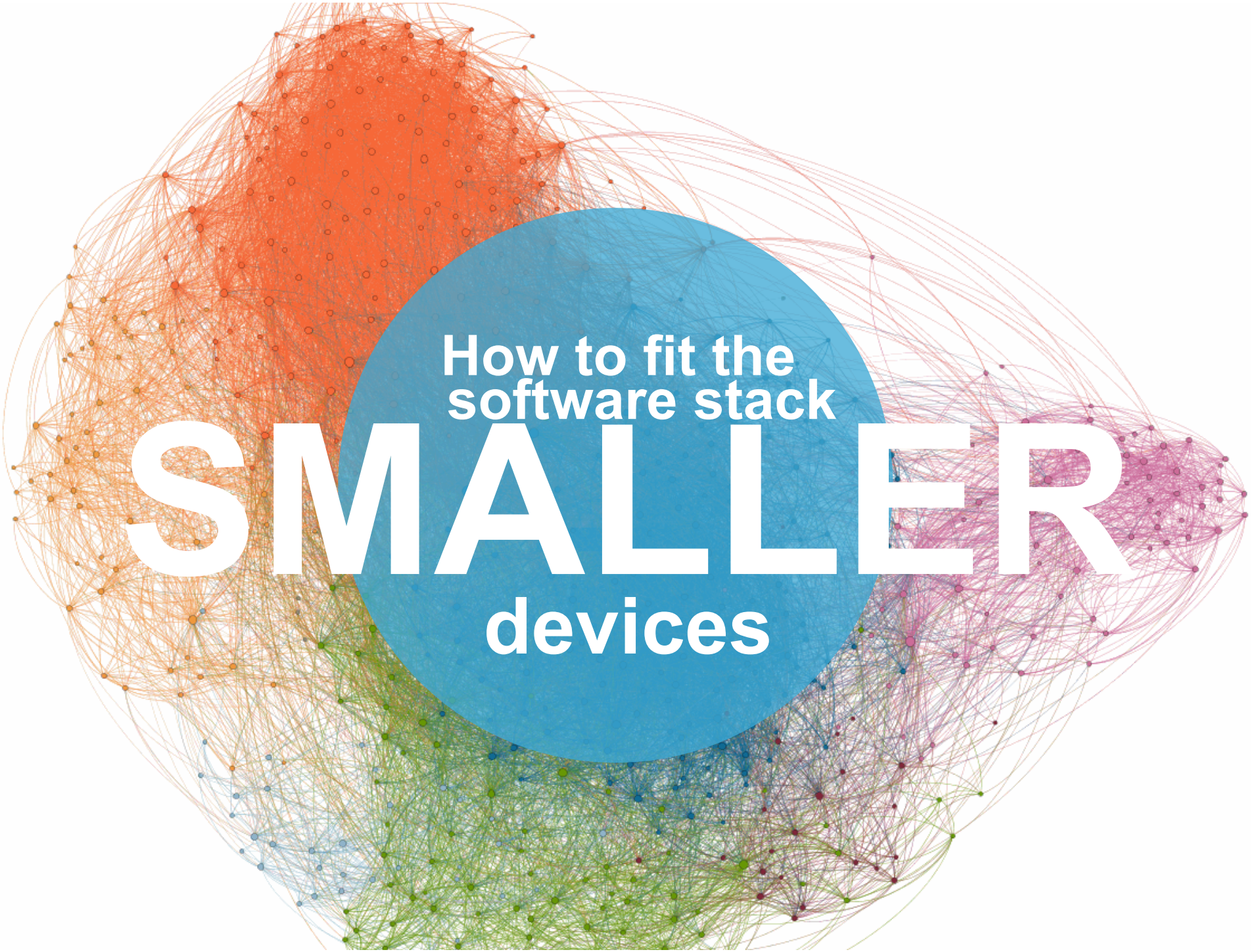


Interactive Exploration of Things as Web-based Use Case in HTML/ Javascript

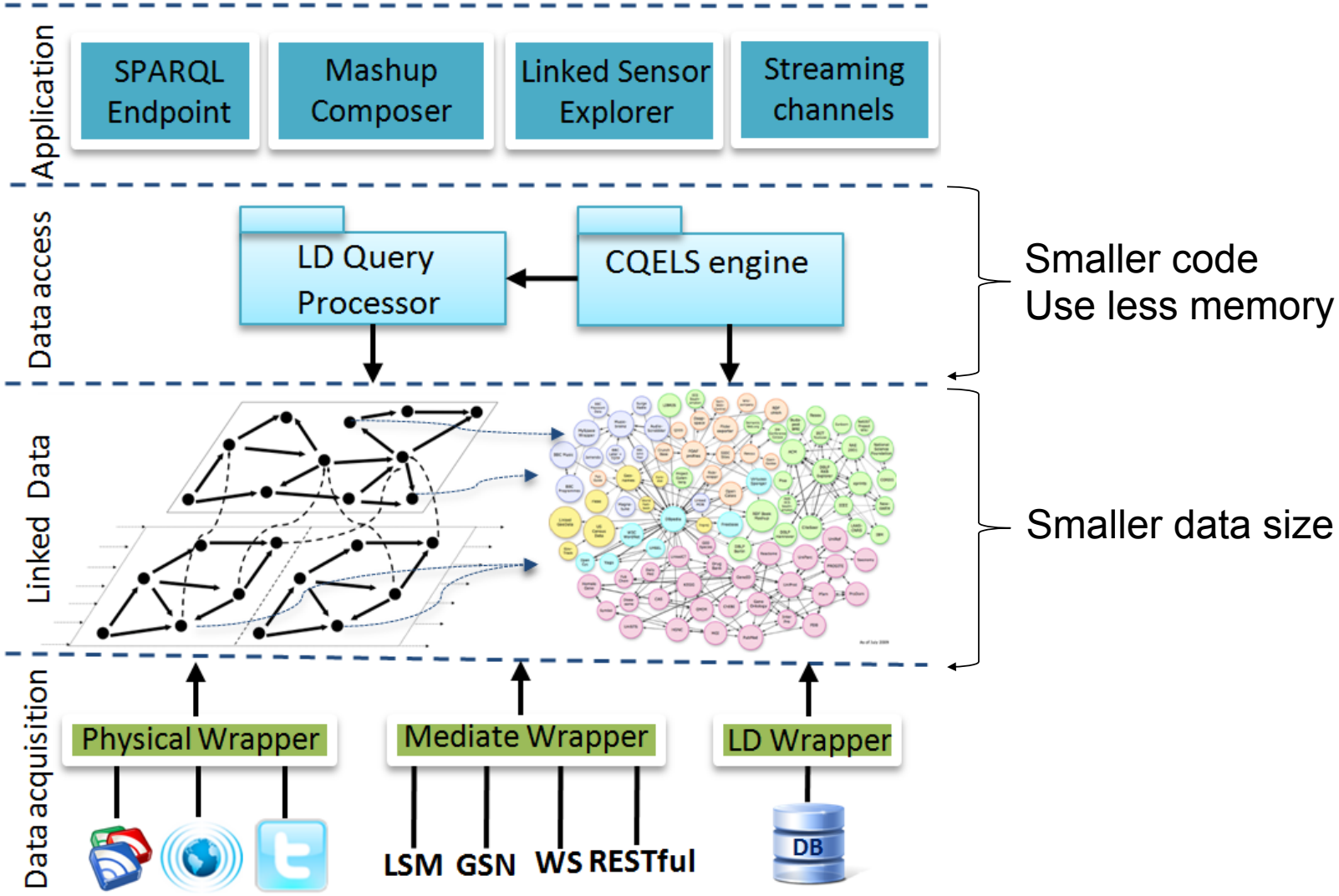


Setup for http://graphofthings.org/

- ✓ **Serves** 400,000+ things (8.5 billion data entries ≈ 140 billion RDF triples)
- ✓ **Hardware**: a cluster of 7 servers(shared 10 GBps network backbone with 10Gbps)
- ✓ **Software** libraries:
  - CQELS Cloud
  - Apache Zookeeper 3.4.5-cdh4.2
  - Apache Storm 0.9.2
  - ElasticSearch 1.5.2
  - OpenTSDB 2.0
  - HBase 0.98.4.


How to fit the
software stack
SMALLER
devices

Smaller software stack for smaller graph?



Embedded CQELS engine

- Current attempt
  - Code foot print : 4MB-8MB
  - Data size : 4-8 million RDF triples
  - Memory foot print : 4MB-64MB
  - Supporting OSs : Android, embedded Linuxes
  - Hardware : RaspberyPI, BeagleBone, Intel Galileo, Android phones, Tablets, etc
- There are plenty room for making it smaller, faster and more scalable!!!

Conclusion & Implementation plan

- **New attractive but efficient extension to CQELS-QL: CQELS-CEP**
  - Path Navigation through the RDF graph/Stream
  - Support RDFS reasoning
  - Support SEQ for CEP
- **Implementation**
  - New features will be available in next release of CQELS
  - Visual Query Editor for CQELS-CEP
  - Open Source in LGPL license