

First impressions of the W3C workshop on Graph Data

SDI.Next: Linked Spatial Data in Europe

March 12, 2019

Amersfoort, The Netherlands


Ivan Herman, W3C

Warning: these are really the first impressions

- The workshop took place last week, ended on Wednesday...
- ...and I has a few days off after the event 😊

The facts

- The Workshop on “Web Standardization for Graph Data”:
 - took place in Berlin, 4-6 March 2019
 - there were ≈ 100 participants
 - one keynote (from Amazon), ≈ 20 presentations, and a bunch of short presentations
 - lots of discussions, panels
 - program, submissions, etc, are available via: <https://www.w3.org/Data/events/data-ws-2019/>



W3C Workshop on Web Standardization for Graph Data

Creating Bridges: RDF, Property Graph and SQL

Monday 4th March to Wednesday 6th March 2019, Berlin, Germany ([venue](#))

Home Page	Call for Participation	Workshop Agenda	Minutes	Position Statements	Report
---------------------------	--	---------------------------------	-------------------------	-------------------------------------	------------------------

Introduction

Data is increasingly important for all organisations, especially with the rise of IoT and Big Data. The falling costs for storage and processing is driving interest in extracting competitive value from ever larger amounts of data through analytics and data hungry AI algorithms. In addition, organisations are seeking to exploit opportunities for sharing data within emerging digital ecosystems. W3C has an extensive suite of standards relating to data that were developed over two decades of experience. These include core standards for RDF, the Semantic Web and Linked Data.

A W3C Workshop is now planned for early 2019 on emerging standardisation opportunities, e.g. query languages for graph databases and improvements for handling link annotations (i.e. embracing property graphs), support for enterprise-wide knowledge graphs, different forms of reasoning that are suited to incomplete, uncertain and inconsistent knowledge, AI and Machine Learning, approaches for transforming data between different vocabularies with overlapping semantics, signed graphs, what's next for remote access to data and information services. In addition, W3C hosts many Community Groups working on data standards and we are interested in what is needed to better support work on vocabulary standards.

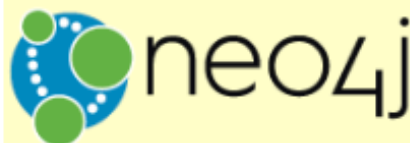
See this Workshop's [Call for Participation](#). Further background is given below.

Graph Databases and Link Annotations

Businesses relied on relational databases (RDBMS) for many years using SQL for query and update. More recently we have seen the rise of NoSQL databases that address the need for flexible handling of unstructured data with key-value stores, document stores, and graph databases. One example is [CouchDB](#) which uses JSON for data storage with ready support for replication for speedy access at different sites. NoSQL is good when you need agility to deal with ever changing data models.

1 Host


W3C gratefully acknowledges Neo4J for hosting this workshop.



2 Sponsors

Platinum, Gold and Silver Sponsorships available

Platinum Sponsors



Gold Sponsors

Lots of sessions...

Monday		Tuesday				Wednesday	
			Interoperation	Problems & Opportunities	Standards Evolution		
		09:00				09:00	Introduction & Reports from Tuesday's sessions
09:30	Chairs/PC Synch	09:30	Graph Data Interchange	Easier RDF and next steps	SQL and GQL	09:30	
10:00		10:00				10:00	
		10:30	Break			10:30	Break
		11:00				11:00	
		11:30	Graph query interoperation	Composition, patterns and tractability	Triumphs and tribulations	11:30	Extending, Incubating, Initiating
		12:00				12:00	
	OPEN	12:30	Lunch			12:30	CLOSE
13:00	Intro & Keynote	13:00					
13:30		13:30					
14:00	Venues & Vectors	14:00	Specifying a Standard	Queries and computation	Rules and reasoning		
14:45	Break	14:30					
15:00	Coexistence or Competition	15:00	Break				
15:30		15:30					
16:00		16:00	Graph models and schema	Temporal, spatial and streaming	Outreach and education		
16:30	Lightning Talks	16:30					
17:00		17:00	Preview of next day				
17:45	Preview of next day	17:15	Posters				

Issues leading to the Workshop 1.

- Increasing importance of graph-based data and databases in general (used in machine learning, Internet of Things, Big Data, etc.)
- The concept of Property Graphs has come to the fore, alongside RDF, for graphs
 - there is a need to find a way to see how these technologies coexist
 - discussions are ongoing on the pro-s and cons of RDF vs. PG
 - *PG is part of the graph data landscape for good!*
- RDB/SQL is also very much present in this area, too

Issues leading to the Workshop 1.

In theory...

- SQL could be extended to do everything for graphs
- SPARQL could be extended to do everything for PG and tables
- A property graph GQL that handles tables and graphs could do everything SQL can do

Issues leading to the Workshop 1.

In practice...

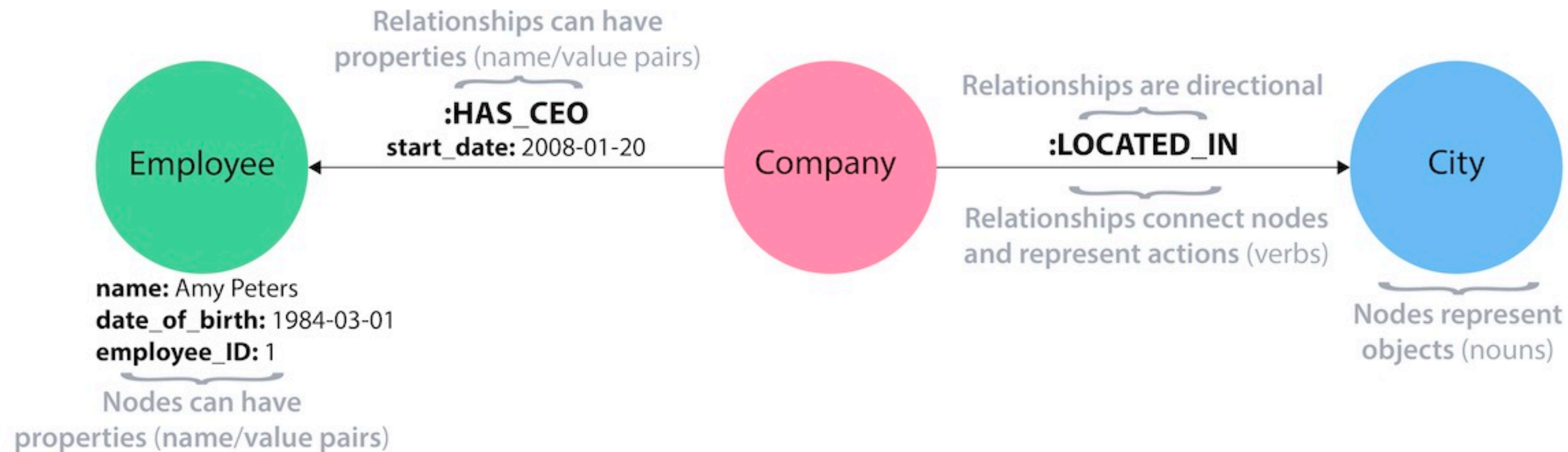
- That would lead to paralysis, or endless war
- Data communities have very deep social and product roots, and large to huge user bases
- Like humans, they can't get personality transplants

Issues leading to the Workshop 2.

- There are also concerns with RDF
 - general acceptance is still relatively slow (although there are great successes)
 - there are many minor (or major...) technical issues with RDF & Co that need housekeeping

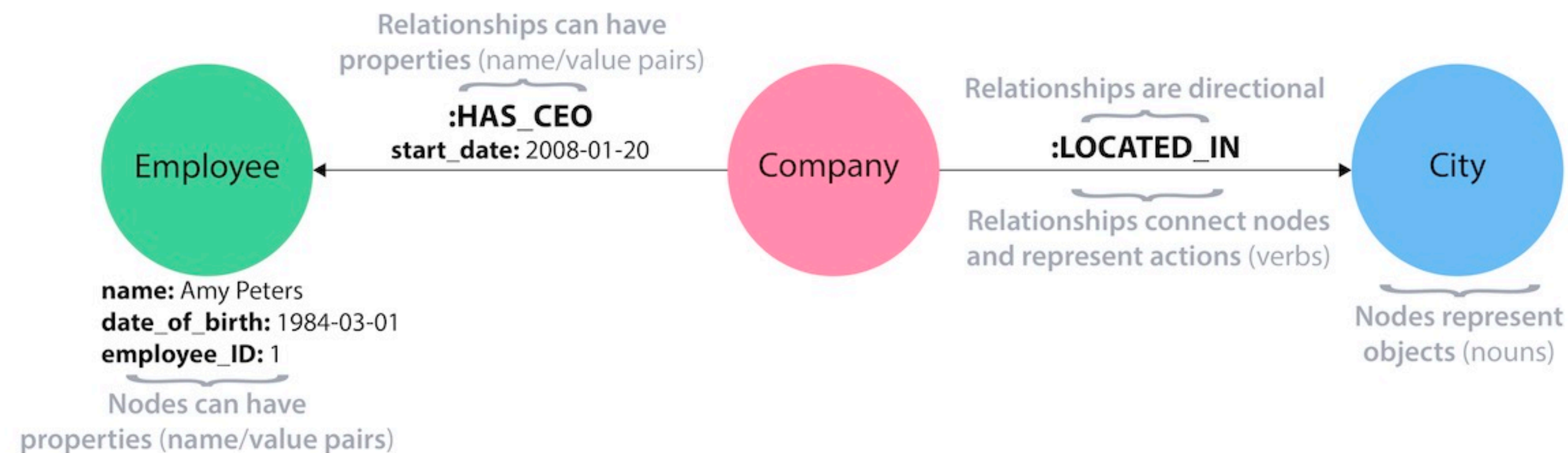
(“RDF”, in the presentation, is a shorthand for full RDF suite, i.e., RDF, RDFS, OWL, SPARQL, SHACL, etc.)

A few words about Property Graphs



Property Graphs

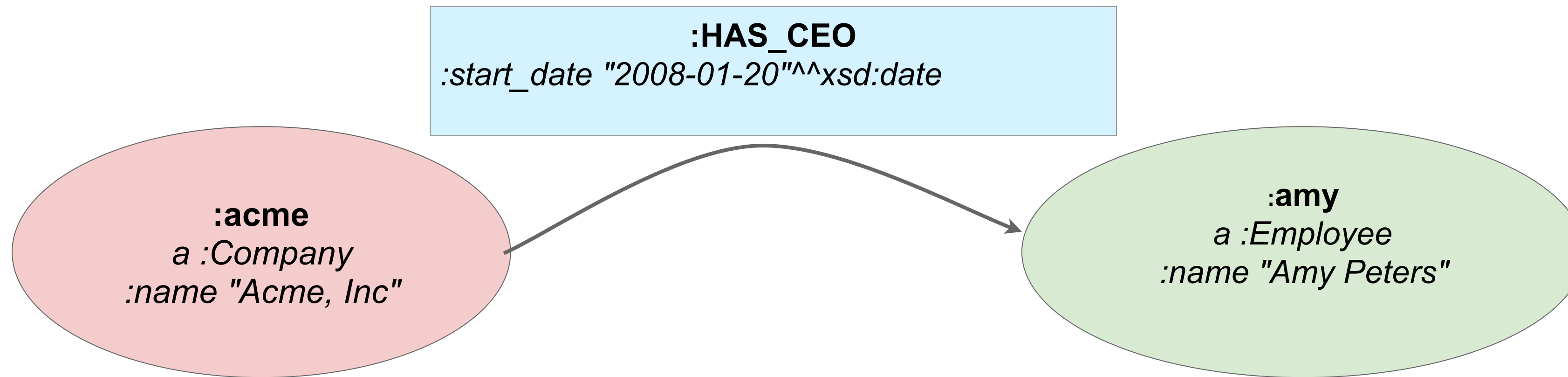
- Framework for representing data and metadata with a graph of nodes and links
 - both nodes and links may have name/value pairs
 - otherwise referred to as “properties”
 - nodes are “just” nodes, not necessarily URL-s
- Link annotations are very useful to assign temporal, spacial, provenance, etc, information easily



Property graphs have a real success

- Non-SQL database vendors, like Neo4j, base their business on this
- Major database providers (Oracle, Amazon's Neptune,...) incorporate PG *as well as* RDF stores
- There are a number of smaller (including open source) implementations (e.g, TinkerPop)
- There are a number of query languages (declarative and imperative), but not *one* winner (yet)
- There is work in the ISO/SQL community to incorporate PG, and define query languages

PG can be represented in RDF



- For example:
 - using reification
 - some sort of an intermediate node (usually BNode) to represent the link
 - use a named graph with a single triple
 - extend RDF to include, somehow, a triple as an entity (e.g., “RDF*”)

PG can be represented in RDF: RDF*

- Instead of something like:

```
Company has_ceo Employee .  
s rdf:type df:Statement .  
s rdf:subject Company .  
s rdf:predicate has_ceo .  
s rdf:object Employee .  
s :start_date "2008-01-20"^^xsd:date .
```

- Use

```
<<Company has_ceo Employee>> :start_date "2008-01-20"^^xsd:date .
```

- To be seen whether this is a syntactic sugar, or RDF should be extended

PG can be represented in RDF

- All these representations do exist in real products
- All have pros and cons
- There is *no* generally accepted way of doing that, i.e., none of those solutions are interoperable.

Why are PG-s interesting for the RDF community?

- They are around on the market...
- They represent, in some ways, a level of abstraction that is easier to understand:
 - by collapsing the “properties” into some sort of labels, the real, “core” aspect of a graph becomes more visible
 - helps in grasping the “essence” of a dataset without being lost in details (date, provenance, tags, etc.)
- adopting a “PG style” would be actually helpful to make RDF more understandable

“...historically, property graphs were somewhat of a reaction to the complexity of RDF. A complex standard will not be accepted by the developer community” (Juan Sequeda)



- The value of RDF is well proven, but...

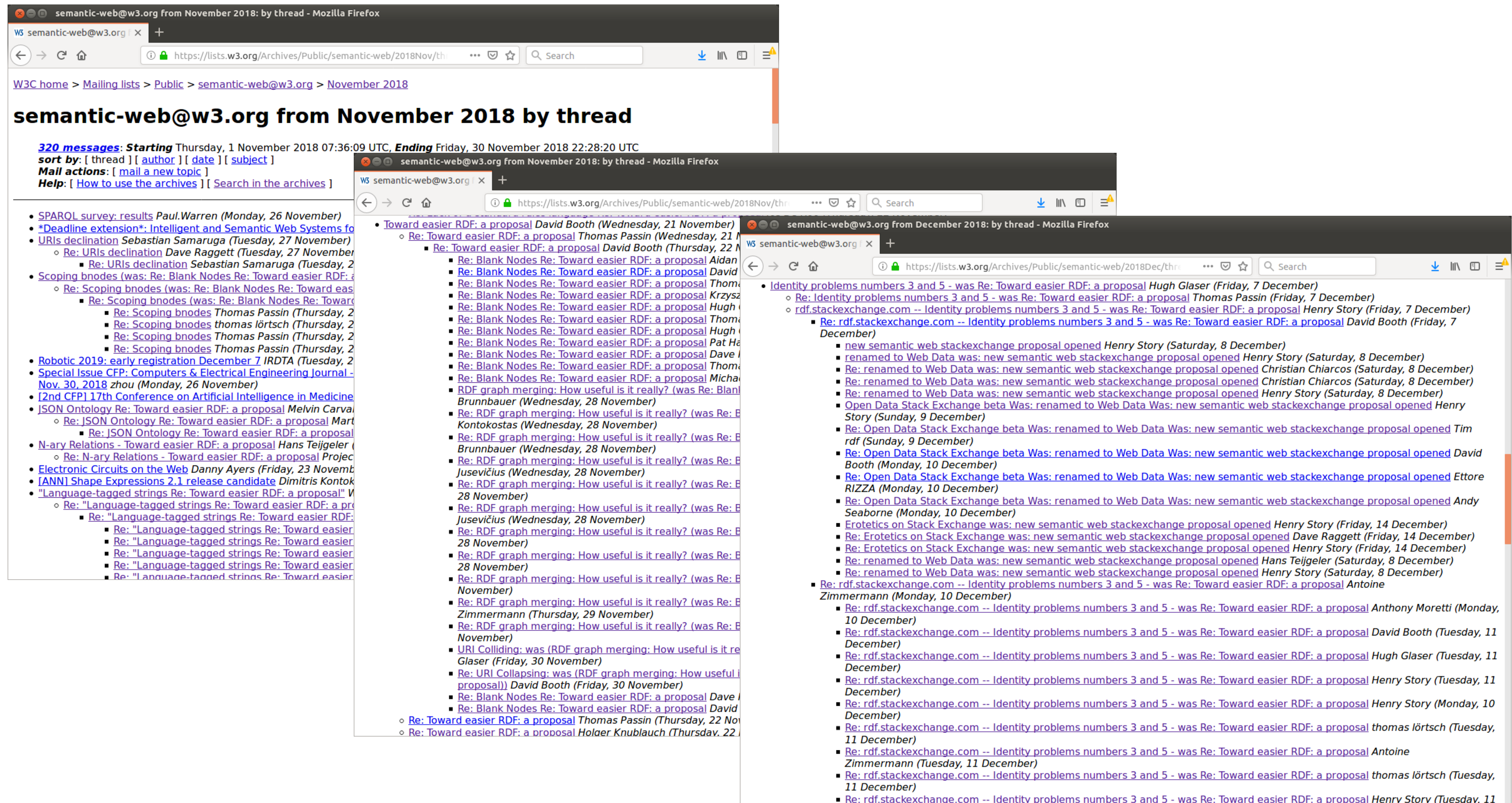


- The value of RDF is well proven, but...
- Too hard for **average** development teams

Issues leading to the Workshop: “EasierRDF” initiative

- Email and github discussion initiated by David Booth
- See: <https://github.com/w3c/EasierRDF>

Over 600 messages!



EasierRDF github site: 50+ issues

The image shows two overlapping browser windows from Mozilla Firefox. The background window displays the GitHub repository page for 'w3c/EasierRDF'. The repository description is 'Making RDF easy enough for average developers'. It shows 35 commits and 1 branch. The README.md file is visible, stating the repository's purpose for experimental work on making RDF accessible to average developers. The foreground window shows the 'Issues' page for the same repository, listing 50 open issues and 3 closed issues. The issues are categorized by labels such as 'language features', 'tools', 'education', 'big ideas', and 'related standards'. The issues listed include:

- Standardized n-ary relations (and property graphs) - Category: language features
- Moribundity of Tools - Category: tools
- SPARQL-friendly lists - Category: language features
- Blank nodes - Category: language features
- Beginner friendly tutorials / documentation - Category: education
- Idea: Higher-level RDF language - Category: big ideas
- Overview of an RDF triple store - Category: tools
- Lack of Technology Framing - Category: education
- SPARQL Triplestore and Reasoning Performance - Category: tools
- Lack of a Good Editor - Category: tools
- SPARQL: The unnamed/default graph should have a standard name - Category: related standards
- Lack of standard RDF canonicalization - Category: related standards

Some result of the email/github discussion

- Technical issues
 - lack of n-ary relations
 - blank nodes
 - missing canonicalization/signature of graphs
 - RDF is too low (“assembly”) level
 - no generally accepted and *simple* rule system
- Non-technical issues
 - lack of beginner level good tutorials
 - no equivalence to, say, MDN
 - no (not yet?) proper integration with Javascript
 - moribundity of tools, registries, lots of abandonware
- Connection to Property Graphs?



Results of the Workshop

Results of the Workshop: many ideas came up

- RDF*/SPARQL* documentation to be submitted to W3C, with possible incubation towards a standard
- extend JSON-LD with PG
- Standards work around PG
 - an abstract (standard) model for Property Graphs†
 - standard mapping between Property Graphs and RDF
 - standard mapping between Property Graphs and Relational Data†
 - W3C Community Group for Graph Query Language (GQL)†
- RDF improvements
 - RDF for stream processing
 - RDF for time, for geographical data, ...
 - solve all the technical and outreach problems in RDF 😊

† Final work probably *not* at W3C

But... this can lead to chaos

- It would lead to unstructured, unrelated work, not necessarily in the right order
- Final decision and further actions:
 - a workshop report should come out in 3-4 weeks
 - set up a *W3C Business Group*:
 - look at the bigger story around data: data is strategic asset for companies. What are the features and mappings that are of importance?
 - derive a prioritized list of technical issues to be solved to fulfill those needs
 - spin off task forces, community groups, etc, to look at the technical issues that are of major importance
 - liaise with other organizations (e.g., ISO) for the activities that are to be done elsewhere
 - look at outreach possibilities in general

Watch this space,
interesting things will happen!

Some links

- Workshop home page:
 - <https://www.w3.org/Data/events/data-ws-2019/>
- All submissions
 - <https://www.w3.org/Data/events/data-ws-2019/papers.html>
- Workshop agenda with links to slides
 - <https://www.w3.org/Data/events/data-ws-2019/schedule.html>
- These slides:
 - <https://www.w3.org/2019/Talks/Amersfoort-IH/Presentation.pdf>

Thank you for your attention