# Towards Building Live Open Scientific Knowledge Graphs

Anh Le-Tuan[*]
anh.letuan@tu-berlin.de
Germany

Carlos Franzreb[†‡]
carlos.franzreb@fokus.fraunhofer.de
Germany

Danh Le-Phuoc[*‡]
danh.lephuoc@tu-berlin.de
Germany

Sonja Schimmler[†‡]
sonja.schimmler@fokus.fraunhofer.de
Germany

Manfred Hauswirth[*†]
manfred.hauswirth@tu-berlin.de
Germany

## ABSTRACT

Due to the large number and heterogeneity of data sources, it becomes increasingly difficult to follow the research output and the scientific discourse. For example, a publication listed on DBLP may be discussed on Twitter and its underlying data set may be used in a different paper published on arXiv. The scientific discourse this publication is involved in is divided among not integrated systems, and for researchers it might be very hard to follow all discourses a publication or data set may be involved in. Also, many of these data sources—DBLP, arXiv, or Twitter, to name a few—are often updated in real-time. These systems are not integrated (silos), and there is no system for users to query the content/data actively or, what would be even more beneficial, in a publish/subscribe fashion, i.e., a system would actively notify researchers of work interesting to them when such work or discussions become available.

In this position paper, we introduce our concept of a live open knowledge graph which can integrate an extensible set of existing or new data sources in a streaming fashion, continuously fetching data from these heterogeneous sources, and interlinking and enriching it on-the-fly. Users can subscribe to continuously query the content/data of their interest and get notified when new content/data becomes available. We also highlight open challenges in realizing a system enabling this concept at scale.

## CCS CONCEPTS

• **Information systems → Graph-based database models**.

## KEYWORDS

knowledge graph, scientific publications dataset, open data

**ACM Reference Format:**
Anh Le-Tuan, Carlos Franzreb, Danh Le-Phuoc, Sonja Schimmler, and Manfred Hauswirth. 2022. Towards Building Live Open Scientific Knowledge Graphs. In *Companion Proceedings of the Web Conference 2022 (WWW '22 Companion), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3487553.3524624

[*]Technical University Berlin.
[†]Weizenbaum Institute, Berlin.
[‡]Fraunhofer Institute for Open Communication Systems, Berlin.


*WWW '22 Companion, April 25–29, 2022, Virtual Event, Lyon, France*

ACM ISBN 978-1-4503-9130-6/22/04…$15.00
https://doi.org/10.1145/3487553.3524624

## 1 MOTIVATION

For researchers, it is essential to keep track of the current research in their field and the impact it generates. To do so, many tools for literature search exist: Researchers can browse **research output**, i.e., publications, data sets and code, etc.. In addition, they can monitor the scientific discourse, i.e., **communication over research output**. For example, a researcher wants to keep track of a certain conference, what papers are published in the proceedings, and what associated data sets and code are available. Or a researcher wants to follow a certain topic, what papers are published on this topic, and which of these papers are discussed on Twitter or in other online forums.

Fortunately an increasing number of systems allow researchers to retrieve the required information. In order to keep track of current research, researchers typically have to manually browse the various sources of interest. For instance, in the computer science domain the relevant sources would be conference proceedings on the ACM and IEEE websites, pre-prints on arXiv or CEUR-WS, data on Zenodo, and source code on GitLab or GitHub. They would also use tools like DBLP, Papers With Code, Google Scholar or Semantic Scholar which retrieve information from other sources and provide an integrated view, e.g., on the papers of a specific researcher irrespective the publication venue. There also exist several scientific knowledge graphs. Two prominent examples are OpenAlex[1] and ORKG[2]. To monitor communication over research output, media sources like news articles or blogs, and (academic) social networks like Twitter, LinkedIn or ResearchGate would have to be consulted.

Even for this rather simple scenario, we can identify two main challenges: (1) (Parts of) the required information is provided by different systems and this information often is not interlinked. For instance, an author of a certain paper is not easy to find and identify correctly on Twitter. (2) An enormous amount of information is provided in near real-time. For instance, a vivid discussion about an interesting topic on Twitter might produce 1000 or more Tweets per hour.

A simplistic ad-hoc answer to this problem may be to apply systems like Google Alerts. However, these "solutions" suffer from a number of drawbacks: (1) the information base is often too general (outside the scientific discourse), (2) new sources cannot be integrated by users, (3) information is updated slowly, (4) the dynamically created connections graph, a valuable resource itself, is not freely available, and (5) no powerful, structured query interface exists. Another commercial solution is Altmetric[3], which offers a

[1]The open catalog to the global research system, https://openalex.org/
[2]The Open Research Knowledge Graph, https://www.orkg.org/
[3]https://www.altmetric.com/

set of tools that collect and sort data from different sources to provide a single view of the activity that surrounds academic projects. This tool also suffers from the drawbacks (2)-(5) mentioned above.

To really address this problem, we present our vision of a live open knowledge graph which can integrate an extensible set of existing or new data sources in a streaming fashion, continuously fetching data from heterogeneous sources, and interlinking and enriching them on-the-fly. Users can subscribe to continuously query the data of their interest and get notified when new content/data in their area of interest becomes available. The outlined system is currently being implemented as a proof-of-concept prototype and will be developed into a fully-fledged system in the future.

In Section 2, we present our conceptual model, and in Section 3 we give insights in our actual data schema. In Section 4 we describe our continuous data curation pipeline, and in Section 5 we present our near real-time publish/subscribe queries. We provide open challenges in Section 6 and give our conclusions in Section 7.

## 2 CONCEPTUAL MODEL

On a conceptual level, we base our approach on a simple model for research output and communication over research output. This is an abstraction of the actual data schema, which is based on established ontologies, such as Linked Data Notifications [2] (whose original ideas can be traced back to the SIOC ontology [1] and have been consumed into schema.org [9]), and the data models that are used in different systems.

We use DBLP, arXiv, meta-data of scientific conferences and Twitter as a representative example in this paper. All these systems provide a flexible API which can be used for our approach: DBLP provides information about recently published papers, arXiv is used to access the actual papers, and Twitter is used to retrieve discussions that are related to the papers. Figure 1 exemplifies the data schema in an abstracted manner: We see the live posts stream – including a new paper on arXiv, a new entry in DBLP and two new tweets – and the corresponding knowledge graph. A snapshot, which is based on the actual data schema, can be seen in Figure 2.

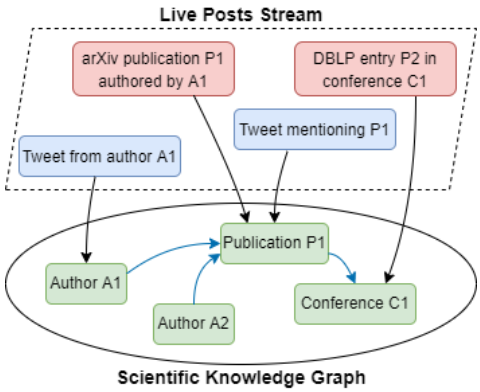


**Figure 1: Abstracted example of the conceptual model regarding a query about the activity surrounding author A1**

We use this simple conceptual model to define a high-level architecture which integrates individual sources and systems on an abstract level. The architecture is then mapped onto the individual APIs. Using a high-level architecture and data schema based on standardized ontologies and relations allows us to integrate further systems easily and formulate (continuous) queries in a common, high-level language.

## 3 DATA SCHEMA FOR SCIENTIFIC DISCOURSE STREAMS

Our data schema design aims to exploit terms from established vocabularies such as Bibliographic Ontology (BIBO) [4], Citation Typing Ontology (CiTO) [13], schema.org [9] and SIOC ontology to model classes and their properties around authors, publications, citations, discussions (tweets, blog posts, etc.), and social interactions (likes, retweets, reshares, quotes, etc.). Moreover, to capture the spatial and temporal aspects of the stream of scientific discourse, related vocabularies and publishing practices such as VoCaLS [14] and W3C/OGC spatial Web best practices [15] can be used.

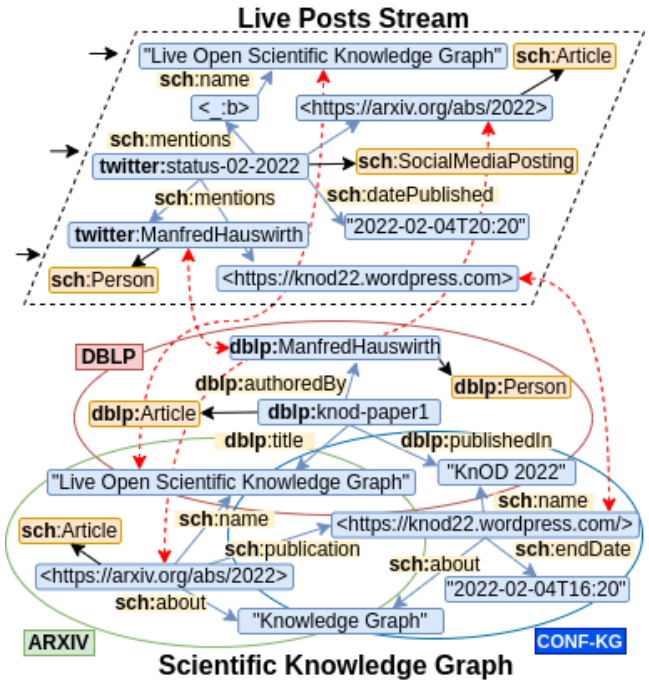


**Figure 2: Snapshot of the live scientific knowledge graph**

Figure 2 illustrates an example how the data is annotated with the current version of our data schema. In the figure, the black arrows represent the "is-a" relationship which is equivalent to the <***rdf***:*type*> property and the blue arrows depict other property relationships.

The upper part of Figure 2 shows the example of a live discussion on Twitter. We use the vocabulary from schema.org to annotate the tweet and to describe the information learned from the tweet. For instance, the URL of the tweet <**twitter**:status-02-2022> is annotated as an instance of <***sch:****SocialMediaPosting*>. The property

[4] https://dublincore.org/specifications/bibo/bibo/

<***sch***:*mentions*> is used to link the tweet to the information it contains. The mentioned information can be a reference such as the URL <**twitter**:ManfredHauswirth> which refers to an author, or the text "Live Open Scientific Knowledge Graph" that refers to the title of a publication. Furthermore, we use the property <***sch***:*datePublished*> to denote the time the tweet was published.

The lower part of Figure 2 depicts an example of the integrated view of three data sources (silos): DBLP, arXiv, and CONF-KG. DBLP contains the information of the recently published scientific paper and it is annotated with the DBLP ontology (red ellipse). For instance, the URL <**dblp**:knod-paper1> refers to a publication and is annotated with class <***dblp:****Article*>. The author and title of the paper is linked to the paper with the property <***dblp***:*authoredBy*> and <***dblp***:*title*>, respectively. The property <***dblp***:*publishedin*> provides the name of the conference where the paper is published. arXiv data (green ellipse) and CONF-KG data (blue ellipse) provide the meta-data of the preprint version of publications and conferences, and they are annotated with schema.org. For instance, the URL <https://arxiv.org/abs/2202> of the preprint is described as an instance of <***sch***:*Article*>. The property <***sch***:*publication*> links the paper to the conference <https://knod22.wordpress.com/>. The title of the paper and the name of the conference are linked to the paper and the conference with the property <***sch***:*name*>. The property <***sch***:*about*> is used to describe the domain ("Knowledge Graph") of the paper and the conference.

Using schema.org gives us the flexibility to integrate live social discussions with a static scientific knowledge-based graph which are collected from related datasets such as DBLP. The DBLP ontology provides direct links to map its concepts to the concepts in schema.org. In our example, the information described with schema.org in the tweet can be linked directly to the knowledge graph. Therefore, we are able to answer queries that require integrating information from different data sources which otherwise would be silos. We highlight these links with dashed lines in red in Figure 2. The queries we enable with this integration we discuss in Section 5.

## 4 CONTINUOUS DATA CURATION PIPELINE

The pipeline starts with the user subscribing a continuous query to the system. A query can refer to a publication the user wants to track, a conference of interest, or a research topic (without limiting the general expressiveness of the queries). The system parses the query and initializes the corresponding knowledge graph, whose structure was described above. For efficiency purposes, also the management system itself can issue "seed queries" to pre-fetch popular streams or stream patterns into the system.

The system then deploys harvesters, which look for relevant information in the configured data sources (e.g., Twitter, arXiv, or DBLP). The information found by the harvesters is treated as events, which are immediately added to the knowledge graph in the appropriate place. This process is illustrated in Figure 3.

For instance, if a tweet mentions an interesting paper for the user, the system adds it to the knowledge graph as a new node and connects it to the paper. Tweets are represented according to schema.org and the SIOC ontology, as done in TweetsKB [5]. Another example is a new pre-print appearing in arXiv under a topic a user is interested in. This paper would also be added dynamically to the knowledge graph as depicted in Figure 3.

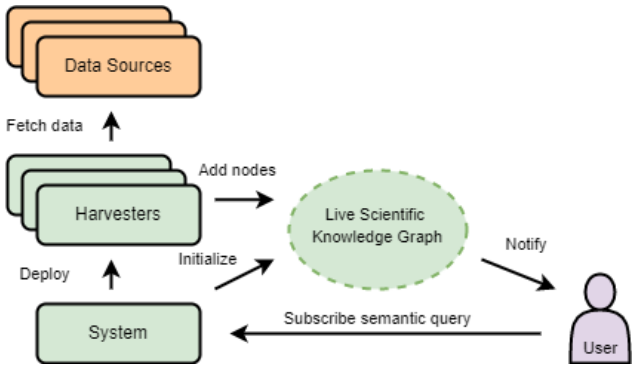


**Figure 3: Schema of the pipeline**

The knowledge graph is extended by each new event, adding new publications, people or conferences. The author of the tweet mentioned in the previous paragraph would also be added to the knowledge graph. In this way, we can track how often they interact with the queried entity, e.g., they publish a paper, cite a paper, or tweet about a paper. We can also map a person's account in different websites with databases like Wikidata, where Twitter identifiers, DBLP identifiers and more can be found for popular researchers.

The live knowledge graph is represented as a stream RDF data model, as described in Section 3. The RDF-based updates of the stream can be pushed continuously into the system from harvesters to be enriched, stored and indexed. This graph includes complex temporal and event patterns that can be queried with one-time queries or with continuous queries in a publish/subscribe fashion.

## 5 NEAR REAL-TIME PUBLISH/SUBSCRIBE SEMANTIC QUERIES

The unified graph data model interlinks different data streams into a unified data stream; thus, a user can issue a continuous query over these streams using a SPARQL-like query language such as CQELS-QL [4, 10]. Different from a typical one-shot SPARQL query to a SPARQL endpoint, a query in CQELS-QL is processed in a publish/subscribe fashion. The CQELS query processor continuously computes the updated information related to the query and delivers matching results incrementally when new ones become available. As an example of a CQELS-QL query, the query in Listing 1 retrieves *"the top-5 papers discussed on Twitter within 48 hours after an event in the Knowledge Graph research community."*

The grammar of the CQELS-QL is similar to the SPARQL grammar. As showed in Listing 1, the query poses graph query patterns to knowledge graphs such as DBLP and CONFKG via the traditional 'GRAPH' keyword (lines 11-16). The new keyword here is 'STREAM' that enables continuous graph query patterns to be expressed. The keywords such as 'WINDOW', 'ON' or 'AFTER' are used to define the temporal relations (lines 11-16). The keyword 'WINDOW' is used to specify the time window of interest, 'ON' to specify which time predicate is used for including events in that window. 'AFTER' is used to specify the 'time reference' for the

window, in our example the time when the conference took place. The graph pattern inside the 'STREAM' block (lines 6-10) uses the SPARQL grammar for basic graph patterns.

```
1  PREFIX sch:<http://schema.org>
2  PREFIX dblp:<http://dblp.org/rdf/schema#>
3  SELECT ?title, count(?post) AS ?nPost
4  WHERE {
5   STREAM <Twitter> WINDOW [48H ON sch:datePublished
6                                AFTER ?eventTime]{
7      ?post a sch:SocialMediaPosting.
8      ?post sch:mentions ?paper.
9      ?post sch:datePublished ?time. }
10  GRAPH <DBLP> {
11     ?paper dblp:publishedIn ?eventName.
12     ?paper dblp:title ?title.}
13  GRAPH <CONFKG> {
14     ?event sch:name ?eventName.
15     ?event sch:about "Knowledge Graph".
16     ?event sch:endDate ?eventTime.}
17 }
18 GROUP BY ?paper
19 ORDER BY ?nPost
20 LIMIT 5
```

**Listing 1: An Example Top-k Query in CQELS-QL**

When the query above is subscribed, the system periodically sends the updated list of five papers with their titles and the number of times they have been mentioned in the last 48 hours. Alternatively, the system can also eagerly notify any changes of this 'top-5' list such as counts, order, or titles of papers made in this list. This top-k kind of queries can be made more complicated to easily build alerting features like those of Google Scholar or ResearchGate. A more complex example is that a user can use one's research profile to compose a personalised feed of scientific discourse based on her previous publications and citations or even research results. In this case, it may require some few nested queries of this kind.

Note that the publish/subscribe mechanism of the above query decouples the query specification step from the data curation pipeline. In particular, the data streams can be ingested and enriched after such queries have been entered to the system. Subsequently, more data or more enriched links and entities can be fused incrementally into the system. As the output of each query is a stream, it can be used as an input to other queries.

## 6 OPEN CHALLENGES

**Flexible graph query language:** While CQELS-QL and similar languages provide a powerful graph query language over graph streams for users and developers, there is the challenge to improve and further develop the system according to user requirements and learning from (new) features offered by other systems, e.g., Google Scholar or ResearchGate. Such features have to be able to understand a user's relevant data such as research profiles, social networks and research interests, e.g., via their published work, to build up graph query networks which specify *most recommended scientific discourses*. To enable these features at scale, several technology and research challenges regarding data management, e.g [6, 12] and information retrieval, e.g [3, 11] for a 'near real-time' infrastructure must be addressed.

**User-friendly graph query language:** Defining queries might also be a challenge for users. To support the user, a well-defined set of generally useful queries should be available, which can be easily re-formulated. In the best of the end user world, a simple graphically query editor would be useful.

**Integrating new data sources:** There exists a variety of additional data sources that could be integrated into the system. Another challenge that arises is the choice of data sources, which heavily depends on the domain at hand. Furthermore, to date, not all these data sources provide a real-time capable interface.

**Retrieving data in near real-time:** The next challenge is in online information retrieval at this large scale (hundred millions of publications/authors and millions of updates per day) as our setup involves many interesting research problems amid the influx of related scientific data sources and the fast developments of related technologies. For example, entity resolution [11] is a very important tool to extract entities and relationships to generate graph nodes and edges for the graph data streams in Section 3. However, how to deal with this scale of data while still guaranteeing time-sensitive constraints in generating results is an open challenge.

**Incorporating data in near real-time:** From a data management perspective, every time new data arrives, the knowledge graph needs to be updated immediately (near real-time) to notify the affected query subscribers of the updates in a timely fashion. For scalability reasons, it is not sensible to recompute all the queries for each single update, but instead apply an incremental evaluation approach. However, incremental evaluation over basic graph query patterns can be NP-complete in general [7]. But the very same research also shows that the incremental evaluation over graph queries can be polynomial if certain restrictions are imposed over the graph query patterns. Also, there are some graph query engines, e.g., [8], which can deal with updates efficiently. Henceforth, it is interesting to investigate whether the graph update patterns of the system pose any new challenges for current graph data systems, and if any new open problems will arise.

**Checking and filtering query results:** Depending on the queries performed by the users, and the data sources chosen, it might be the case that a huge amount of data is retrieved, which is potentially of low quality or even wrong, e.g., fake information, out-dated data, old versions of papers, etc.. A fully fledged system should include some filtering options, and also some tools to check the quality of the data and to identify these types of information.

## 7 CONCLUSIONS

The disjoint discourse among heterogeneous, not integrated sources of research output and communication over research output is a major obstacle for scientific progress as it hinders the automated generation of an up-to-date representation of the relevant sources in an area. Researchers are left to their own search skills and a few semi-automatic tools that integrate some of the sources. In this paper, we have suggested a more comprehensive and global integration that supports an integrated view over distributed research output and communication over research output. As a specifically useful functionality, our approach allows users to state their interests and register them as continuous queries, so that they get notified as soon as sources relevant to them become available. As a "by-product" our system will incrementally build a live open knowledge graph, which can be viewed even as the major contribution of our work as it will be reusable in many contexts.

## ACKNOWLEDGMENTS

This work was funded by the German Research Foundation (DFG) under project no. 460234259 (NFDI4DataScience) and under grant no. 453130567 (COSMO project) and the Federal Ministry for Education and Research under grant nos. 01IS18025A and 01IS18037A (Berlin Big Data Center and Berlin Institute for the Foundations of Learning and Data) and 16DII128 (Weizenbaum Institute, "Deutsches Internet-Institut").

## REFERENCES

[1] John G Breslin, Andreas Harth, Uldis Bojars, and Stefan Decker. 2005. Towards semantically-interlinked online communities. In *European semantic web conference*. Springer, 500–514.

[2] Sarven Capadisli, Amy Guy, Christoph Lange, Sören Auer, Andrei Sambra, and Tim Berners-Lee. 2017. Linked data notifications: a resource-centric communication protocol. In *European Semantic Web Conference*. Springer, 537–553.

[3] Vassilis Christophides, Vasilis Efthymiou, Themis Palpanas, George Papadakis, and Kostas Stefanidis. 2021. An Overview of End-to-End Entity Resolution for Big Data. *ACM Comput. Surv.* 53, 6 (2021), 127:1–127:42. https://doi.org/10.1145/3418896

[4] Daniele Dell'Aglio, Minh Dao-Tran, Jean-Paul Calbimonte, Danh Le Phuoc, and Emanuele Della Valle. 2016. A Query Model to Capture Event Pattern Matching in RDF Stream Processing Query Languages. In *Knowledge Engineering and Knowledge Management - 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings (Lecture Notes in Computer Science, Vol. 10024)*, Eva Blomqvist, Paolo Ciancarini, Francesco Poggi, and Fabio Vitali (Eds.). 145–162. https://doi.org/10.1007/978-3-319-49004-5_10

[5] Pavlos Fafalios, Vasileios Iosifidis, Eirini Ntoutsi, and Stefan Dietze. 2018. Tweetskb: A public and large-scale rdf corpus of annotated tweets. In *European Semantic Web Conference*. Springer, 177–190.

[6] Wenfei Fan, Tao He, Longbin Lai, Xue Li, Yong Li, Zhao Li, Zhengping Qian, Chao Tian, Lei Wang, Jingbo Xu, Youyang Yao, Qiang Yin, Wenyuan Yu, Kai Zeng, Kun Zhao, Jingren Zhou, Diwen Zhu, and Rong Zhu. 2021. GraphScope: A Unified Engine For Big Graph Processing. *Proc. VLDB Endow.* 14, 12 (2021), 2879–2892. http://www.vldb.org/pvldb/vol14/p2879-qian.pdf

[7] Wenfei Fan, Chunming Hu, and Chao Tian. 2017. Incremental Graph Computations: Doable and Undoable. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*, Semih Salihoglu, Wenchao Zhou, Rada Chirkova, Jun Yang, and Dan Suciu (Eds.). ACM, 155–169. https://doi.org/10.1145/3035918.3035944

[8] Wenfei Fan, Chao Tian, Ruiqi Xu, Qiang Yin, Wenyuan Yu, and Jingren Zhou. 2021. Incrementalizing Graph Algorithms. In *SIGMOD '21: International Conference on Management of Data, Virtual Event, China, June 20-25, 2021*, Guoliang Li, Zhanhuai Li, Stratos Idreos, and Divesh Srivastava (Eds.). ACM, 459–471. https://doi.org/10.1145/3448016.3452796

[9] Ramanathan V Guha, Dan Brickley, and Steve Macbeth. 2016. Schema. org: evolution of structured data on the web. *Commun. ACM* 59, 2 (2016), 44–51.

[10] Danh Le-Phuoc, Thomas Eiter, and Anh Lê Tuán. 2021. A Scalable Reasoning and Learning Approach for Neural-Symbolic Stream Fusion. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 4996–5005. https://ojs.aaai.org/index.php/AAAI/article/view/16633

[11] George Papadakis, Ekaterini Ioannou, Emanouil Thanos, and Themis Palpanas. 2021. *The Four Generations of Entity Resolution.* Morgan & Claypool Publishers. https://doi.org/10.2200/S01067ED1V01Y202012DTM064

[12] Botao Peng, Panagiota Fatourou, and Themis Palpanas. 2021. ParIS+: Data Series Indexing on Multi-Core Architectures. *IEEE Trans. Knowl. Data Eng.* 33, 5 (2021), 2151–2164. https://doi.org/10.1109/TKDE.2020.2975180

[13] Silvio Peroni and David Shotton. 2012. FaBiO and CiTO: ontologies for describing bibliographic resources and citations. *Journal of Web Semantics* 17 (2012), 33–43.

[14] Riccardo Tommasini, Yehia Abo Sedira, Daniele Dell'Aglio, Marco Balduini, Muhammad Intizar Ali, Danh Le Phuoc, Emanuele Della Valle, and Jean-Paul Calbimonte. 2018. VoCaLS: Vocabulary and Catalog of Linked Streams. In *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 11137)*, Denny Vrandecic, Kalina Bontcheva, Mari Carmen Suárez-Figueroa, Valentina Presutti, Irene Celino, Marta Sabou, Lucie-Aimée Kaffee, and Elena Simperl (Eds.). Springer, 256–272. https://doi.org/10.1007/978-3-030-00668-6_16

[15] Linda van den Brink, Payam M. Barnaghi, Jeremy Tandy, Ghislain Atemezing, Rob Atkinson, Byron Cochrane, Yasmin Fathy, Raúl García-Castro, Armin Haller, Andreas Harth, Krzysztof Janowicz, Sefki Kolozali, Bart van Leeuwen, Maxime Lefrançois, Joshua Lieberman, Andrea Perego, Danh Le Phuoc, Bill Roberts, Kerry Taylor, and Raphaël Troncy. 2019. Best practices for publishing, retrieving, and using spatial data on the web. *Semantic Web* 10, 1 (2019), 95–114. https://doi.org/10.3233/SW-180305