THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Why Data Citation is a Computational Problem

Citation for published version:
Buneman, P, Davidson, S & Frew, J 2016, 'Why Data Citation is a Computational Problem' Communications of the ACM, vol. 59, no. 9, pp. 50-57. DOI: 10.1145/2893181

Digital Object Identifier (DOI):
10.1145/2893181

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Communications of the ACM

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 05. Apr. 2019

Why data citation is a computational problem

Peter Buneman
University of Edinburgh

Susan Davidson
University of Pennsylvania

James Frew
University of California, Santa Barbara

March 7, 2016

Abstract

Most information is now published in complex, structured, evolving datasets or databases. There is increasing demand that this digital information should be treated in the same way as conventional publications and be appropriately cited. While principles and standards have been developed for data citation, they are unlikely to be used unless we can couple the process of extracting information with that of providing a citation for it. We discuss how to *generate citations automatically* for data in a database given how the data was obtained – the *query* – as well as the content – the *data*. We show how the problem of generating a citation is related to a well-understood problem in databases and describe this in two examples with radically different citation requirements.

1 Introduction

Citation is the basis for traditional scholarship. We use a citation to identify the cited material, to help retrieve it, to give credit to the creator of the material, to date it, and so on. In the context of printed materials, such as books and journals, citation is well understood. However, the world is now digital. Most of our scholarly and scientific resources are held online and many are in some kind of database, i.e. a structured, evolving collection of data. For example, most biological reference works have been replaced by curated databases, and vast amounts of basic scientific data – geospatial, astronomical, molecular, etc. – are now available on-line. There is strong demand [9, 21] that we should accord the same scholarly status to these databases and cite them appropriately, but how can we do this effectively?

It is because of the structure and evolution of databases that citation is a challenge. Attributes such as ownership or authorship may change for different parts

of the database. Even for a simple collection of files, we may want to find good methods of citing subsets of these files; that is we want to do better than cite the whole collection or generate a huge number of citations to individual files. We need at least to specify how a citation is to be extracted from the database.

A citation is a collection of "snippets" of information, such as authorship, title, ownership, date, etc that are specified by the database administrators and which may be prescribed by some standard. However, if we expect people to cite digital data, simply providing principles and standards for citation is not enough – we must also *generate* the citations. Even when making conventional citations to the literature, we typically avoid typing in citations. Instead we look for the citation in some database of citations (e.g. the ACM Digital Library[1] or DBLP[2]) and insert it into our document using a reference manager (BibTeX, Mendeley, Zotero, etc.) or by copy-paste. In the context of citing databases, if the citation is not available or if the standard appears complicated, we are almost certain to omit the citation or provide an inaccurate one. In short, *unless we couple the process of generating a citation with the act of extracting the data, the advocacy of data citation will have limited effect.*

How then are we to generate citations for data extracted from a database? Following our broad use of the term "database", we use the term "query" to mean any mechanism used to extract the data – for example, a set of file names, an SQL query, a URL, a special purpose GUI, etc. The problem we then need to solve is simply formulated as follows:

Given a database D and a query Q, generate an appropriate citation.

It is often the case that the curators, authors or publishers of a database have good ideas about how their data should be cited. However, it is unlikely that they will know how to associate a citation with some complex SQL query, and even less likely that the user of the data, whose query was generated by some user interface, will understand what is wanted. We need to extract the citation *automatically* from the query Q and the database D, which raises two questions:

- Does the citation depend on both Q and D, or just on the data $Q(D)$ extracted by Q from D?
- If we have appropriate citations for some queries, can we use these to construct citations for other queries?

If the retrieved data is simply a number or an image, we cannot expect to find the citation in the retrieved data. Moreover, if the query returns nothing, it may be worthy of citation – but what citation is associated with the empty set? We need at least context information; we need both Q and D.

[1] `http://dl.acm.org/`
[2] `http://dblp.uni-trier.de/`

The answer to the second question is important because authors and publishers frequently have ideas on how to cite certain parts of the database, i.e., they can provide citations for certain queries, but they do not know what to do about other queries.

We should point out that numerous organizations [11, 14, 16, 24] have advocated data citation and developed principles [2, 7, 8, 9, 14, 15, 23, 24] that refine and standardize the notion [1, 2, 4, 7, 13, 23]. The purpose of these standards is mostly to prescribe and to describe the information in a citation.

A major, but not the only, purpose of a citation is to identify the cited material, and citation is often linked to persistent identifiers such as DOIs[3], ARKs[4], or URIs[5]. These identfiers, while they may have certain fixed properties, do not guarantee *fixity* – that the cited material remains unchanged. Beyond observing that citations should reference the appropriate version, we do not address fixity in this paper; nor do we address the closely related topic of provenance which, in addition to archiving, involves a record of the whole process of data extraction. For a discussion of these issues and a prototype system that combines citation and provenance, see work by Pröll and Rauber [19, 20].

In the remainder of this paper, we propose a general approach to citation generation (Section 3), and illustrate it in the context of two very different scientific databases (Section 2).

2 Sample Scientific Datasets

To illustrate the computational issues of data citation, we describe two scientific datasets that differ widely both in their structure and in how they should be cited.

2.1 GtoPdb

The IUPHAR/BPS Guide to Pharmacology (GtoPdb) [18][6] is a relational database that contains expertly curated information about drugs in clinical use and some experimental drugs, together with information on the cellular targets of the drugs and their mechanisms of action in the body. The resource is particularly useful to researchers who hypothesize that a particular cellular mechanism is involved in a physiological process of interest, and want to find tools (drugs) to impose a specific activation level on the pathway to test their hypotheses.

Users view information through a hierarchy of web pages. The top level divides

[3] http://dx.doi.org/10.1000/182
[4] http://confluence.ucop.edu/display/Curation/ARK
[5] http://www.ietf.org/rfc/rfc3986
[6] http://www.guidetopharmacology.org/

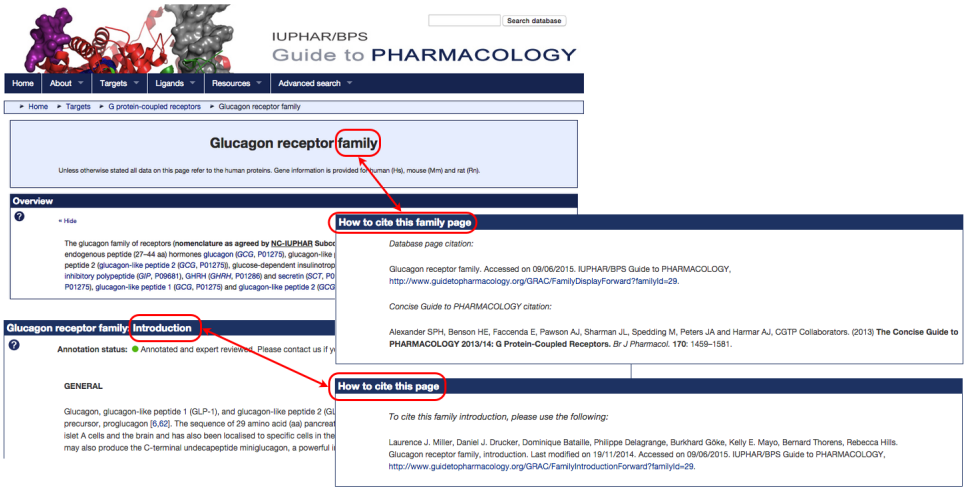

Figure 1: GtoPdb Family and Introductory pages with independent citations

information by "families" of drug targets that reflect typical pharmacological thinking; lower levels divide the families hierarchically into sub-families and so on down to individual drug targets and drugs. At the lowest level are expert-created overviews and, for some entries, pages containing details of chemical and genetic structures and properties. Despite its underlying relational implementation, GtoPdb can therefore be thought of as a structured hierarchy.

Information in GtoPdb is generated by hundreds of expert contributors, and different database entries are associated with different lists of contributors. While the suggested citation for GtoPdb as a whole (the root) is a traditional paper written by its curators, a citation to a subtree of GtoPdb includes the contributors who generated the content, see Figure 1. Note that the citation also includes the path from the root to the subtree (the query), and this is important as a few targets are members of more than one sub-family, and therefore the path that was taken through the hierarchy to a target page may vary, and the citation may depend on that path. Queries against GtoPdb may return a boolean value or the empty set, and to cite this fact – for example to determine the relevant contributors – one clearly needs the query. An important and useful property of GtoPdb is that nearly all the information needed to construct a citation, such as names of contributors, is in the database itself.

2.2 MODIS

MODIS (MODerate-resolution Imaging Spectrometer) is an electromechanical optical imaging system currently flying aboard NASA's Terra and Aqua satellites. Each MODIS sensor images the entire surface of the Earth every one to two days, as a strip approximately 2000 km wide beneath the satellite's orbit.

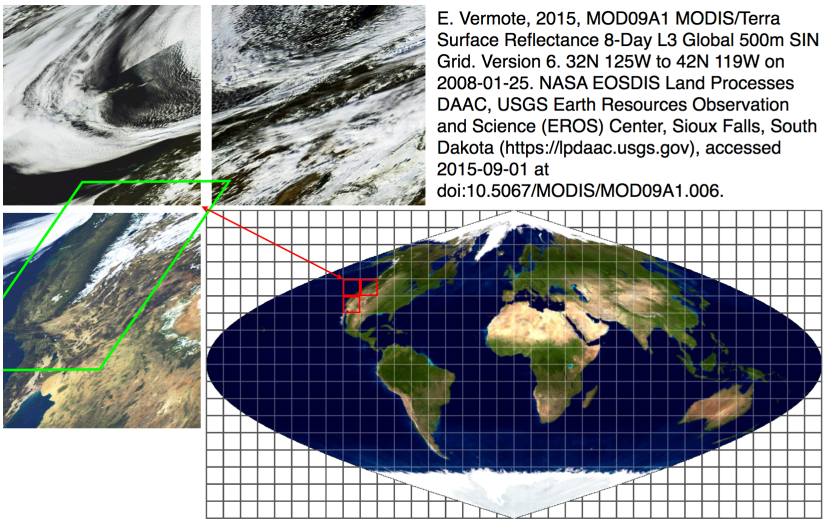

Figure 2: The MODIS grid: highlighted tiles (red) of spatial extent for California (green), with citation

The MODIS sensor records the top-of-atmosphere radiance in several spectral bands, but MODIS data products typically process these values into Earth surface properties such as reflectance, snow cover, ocean color, etc.

MODIS data products are distributed as *granules*: fixed-sized subsets representing either an interval (typically 5 min) of the satellite's orbit, or a tile within a standard map projection of all or part of the Earth (see Figure 2). Each MODIS granule is created, stored, and distributed as a single Hierarchical Data Format (HDF) file. MODIS data product search and access systems typically identify and return entire granules, not subsets thereof.

Each MODIS data product defines a granule naming convention, typically incorporating the product identifier, a version number, date-times of acquisition and generation, and (if applicable) a tile identifier. A granule name is thus a unique identifier for the granule, but is not in itself a complete citation, for two reasons. First, applications of MODIS data products frequently use multiple granules, and there is no standard way to refer to a set of granules other than by complete enumeration. Second, applications of MODIS data products frequently focus on spatiotemporal regions of interest that are not precisely aligned with granule boundaries; thus, an application's query against a MODIS data product may not be precisely reflected in the corresponding set of product granules. For example, compare the latitude-longitude bounding box for California in Figure 2 with the non-rectangular set of MODIS tiles that intersect the box. While enumerating this set is important for provenance, a spatio-temporal bounding box is a compact description of the coverage, which – if expressed in a common

co-ordinate system – allows easy searching for studies relevant to a particular region. Such bounding boxes are a common feature of geospatial citations; indeed a spatial bounding box is one of the optional fields in the DataCite schema [4].

3 Towards a Solution

We now address the problem of generating a citation for a query Q on database D. As we saw with GtoPdb and MODIS, the citation will depend on both Q and D. This would appear to be a major problem, since anything that involves the analysis of a query or program is likely to be computationally expensive, if not undecidable. However, as we will see, the problem may be alleviated if we have a base of citations for *views* of D – *citable units* – which may then be used to generate citations for other queries. From a practical perspective, it is unlikely that a data publisher will be able to associate a citation with an arbtrarily complex query; however it should be possible for them to say "For this part of the database, the citation should look like this". If several "parts of the database" can be formalized as a view, then we have a basis for generating citations.

3.1 Views and Citable Units

The standard notion of a database view is: given a database schema S, a *view* is some function V which, when applied to any instance of S (i.e., any database that conforms to S), produces a database in some other schema S'. Note that the input and output database schemas do not have to be in the same data model: We could, for example, have an XML view of a relational database. Views have been used in traditional database architectures to describe "areas of responsibility" for parts of a database. What we propose here is to use them to create "citable units"[7]

Figure 3 shows a simplified[8] representation of GtoPdb as a hierarchy, which is how it is published as web pages and understood by many contributors and users. There are four different classes of nodes in the hierarchy: the root, families, introductions (to families), and targets. Each of these nodes defines a view which is the subtree beneath it, and the GtoPdb curators have specified a different citation for each class. The higher levels of the hierarchy have citations with collaborators (editors or curators) and the lower levels with contributors. The curators of GtoPdb would like to carry citation down to the level of tables and tuples, but currently a citation for any other node in the hierarchy is the citation for the nearest ancestor of that node.

[7] See Appendix B for further discussion of citable units and Appendix C for database views.

[8] To simplify presentation, we assume that families are all directly under the root in GtoPdb. In reality, some families may be grouped together as subfamilies of another family.

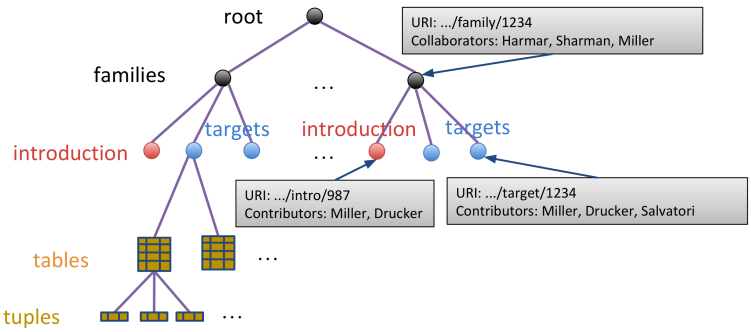

Figure 3: The GtoPdb hierarchy showing the citable views and some partial citations.

This is a promising start for defining citations for the hierarchical (web) presentation of the database, but recall that the underlying database is relational. How do we use these ideas to provide a citation for some SQL query against the database? We can turn this into a question about views. Suppose we have a database schema S, a view V over S and a query Q. If Q can be expressed as a query over V, then the citation associated with V is a *candidate citation* for Q. More formally, if, there is a query Q' such that, for all instances D of S, $Q(D) = Q'(V(D))$, then the citation for V is a candidate citation for Q.

The view (the subtree) for each node in the hierarchy is given by a simple query on the underlying database. For example, there is a TARGET table whose primary key is a target identifier TID. For any value x of TID, and for any table that has TID as a foreign key, we select the rows that contain x. We now get a set of tables, each of which is a subset of the rows of the table in the original database. This is a view defined by x and each such value of TID defines a distinct *target view*. A similar construction works for families: there is a FAMILY table whose primary key is a target family identifier FID. For any value x of TID, and for any table that has FID as a foreign key, we select the rows that contain x. However, we also include in this view the union of tables of subfamilies of FID or (in the case of lowest level families), the union of target tables contained in FID. Each value of FID defines a distinct *family view*.

So the question of what citation to use for a relational query boils down to whether it can be answered using one of these relational views. Unfortunately, while simple to state, the problem of rewriting a query using a view (or set of views) has been extensively studied in the context of query optimization, maintenance of physical data independence, and data integration [10, 12, 5]. The general problem is no simpler than program equivalence, which is undecidable; however, for answering *conjunctive queries* over *conjunctive views* the problem is NP-complete with practically efficient solutions. However, even if we are in a restricted situation where the problem is solvable, there may be 1) no

views that support a given query; 2) more than one candidate view; or 3) the query may be expressible as a function on two or more candidate views, e.g., $Q(D) = Q'(V1(D), V2(D))$.

In spite of these potential issues, the formulation is useful in many practical cases, in particular when the views form a hierarchy that allows us to choose the "best" view from a candidate set.

3.2 Hierarchies of views

A hierarchy of views is formed by a view refinement (subview) relationship: Given two views W and V of the same database, we call W a *subview* of view V if there is a view W' such that $W(D) = W'(V(D))$ for all instances D of the database. Trivially, each view of the database is a subview of the view returning the database itself. What we want for a citation is a *smallest* view V for which Q is a subview.

In the context of GtoPdb, there is a natural view hierarchy: The view for target TID is a subview of any family view which contains the target TID. In the hierarchical view of the data (Figure 3) the tree for TID is a subtree of the tree for FID; in the relational representation, each table in TID is a subset of the corresponding table in FID. Note that each view corresponds to a simple SQL query (a conjunctive query) over the relational representation, and that for this class of views the problem of answering a query using a view is possible.

To specify simple views in a hierarchical structure, we can use a path language such as XPath.[9] For example, in GtoPdb there are three classes of view: one for the family page, one for the family introduction page, and one for the target page. We can specify them as follows:

Family view:	/Root/Family[FamilyName=$$f]
Introduction view:	/Root/Family[FamilyName=$$f]/Introduction
Target view:	/Root/Family[FamilyName=$$f]/Target[TargetName=$$t]

Each of these specifies a *class* of views, parameterized by variables indicated by $$. For the family and introduction view, each value of `$$f` gives us a view (a node in the tree) and for the target view we need both `$$f` and `$$t`. We shall refer to these views as *parameterized* views.

When someone uses the web interface to GtoPdb, each page they visit is specified by a path from the root. For example:

/Root/Family[FamilyName="Melatonin"]/Target[TargetName="MT1"]/LigandTable

This can be answered using the Target view defined above. It can also be answered by following the link in the Family view to "MT1", however the former is more specific and would therefore be the preferred citable unit. Recall that

[9] http://www.w3.org/TR/xpath/

the citations for the two views could be different, as illustrated by the grey boxes in Figure 3.

Equally, suppose someone had queried the underlying database with a simple selection on the Family table with Name = 'Calcitonin'. Given that each citatable view in GtoPdb is a set of conjunctive queries, it is easy – and in this case easy – to determine that this could be answered using the Family view for Calcitonin.

As we have seen, it is possible that a query could be answered in two ways, perhaps through the union of several Target views or through one Family view. This could be resolved through a policy by the data publisher or by presenting the alternatives to whoever wants to construct the citation.

3.3 Generating citations

Having set up a basis for identifying an appropriate citation, how do we generate one? Here we propose a simple rule-based language in which we use XPath syntax to define *patterns* that are matched against a hierarchy (the body of the rule) to produce the required citation (the head of the rule). Figure 4 shows a simple rule for generating a citation together with a citation that is generated by that rule. The right-hand side of the rule is an XPath-like expression that contains two kinds of variables: $\$\$x$ variables are the view parameters; and $\$x$ variables are bound once the $\$\$x$ variables have been matched. Here, the contributors, which depend on the family and the version number, which is unique to the database, are extracted.

The left-hand side of the rule contains the citation in whatever syntax is preferred. Here, we have assumed a simple JSON-style syntax, but the syntax could be in one of the numerous citation "styles", or some more generic syntax such as BibTeX [10] or DataCite [4]. In this example we have assumed that the database name and the URI are constants in the citation.

The sample result in Figure 4 is the citation for the simple path
/Root/Family[FamilyName="Calcitonin"]
It is also the citation for a simple SQL selection on the Family table with Name = 'Calcitonin' the SQL query above. In these cases, it is rather easy to determine that that it can be answered using the appropriate relational version of the Family view.

3.4 Citations and MODIS

From a database perspective, MODIS is much simpler than GtoPdb. It is a hierarchically organized collection of products (e.g. surface reflectance products) consisting of a set of granules, which we assume for now are tiles (see Figure 2).

[10] http://www.bibtex.org/

```
{ Title: "IUPHAR/BPS Guide to Pharmacology", Version: $v,
  Family: $$f, Contributors: $a, URI: "www.iuphar.org" }
←−
/Root[VersionNumber: $v]/Family[FamilyName: $$f]/Introduction[Contributor-list: $a]

{ Title: "IUPHAR/BPS Guide to Pharmacology", Version: 26, Family: "Calcitonin",
  Contributors: ["Debbie Hay", "David R. Poyner"], URI: "www.iuphar.org" }
```

Figure 4: A citation specification and a sample result for GtoPdb

A typical retrieval will ask for a set of tiles that cover a certain region of the Earth's surface and whose time stamp is within a given interval – a spatio-temporal bounding box of granules. For example, supposing one were interested in the surface reflectance for California on 25 January 2008, the granules could be specified by a bounding box whose latitude and longitude are the ranges [32,42] and [-125, -119][11] and time 2008-01-25.

The query to retrieve these granules can be expressed as a range query. If we group MODIS products into a hierarchy, our spatio-temporal query may be expressed in a path language as follows:

```
/root/product[ProdName="surface reflectance"]/file[Lat ≥ 32 and Lat ≤ 42 and
                                                   Lon ≥ -125 and Lon ≤ -119 and
                                                   Time = 2008-01-25]
```

This closely reflects the retrieval capabilities of many MODIS product distribution systems. To describe this common bounding box retrieval pattern, an appropriate parameterized view would be:

```
/root/product[ProdName=$$p]/file[ Lat ≥ $$minlat and Lat ≤ $$maxlat and
                                  Lon ≥ $$minlon and Lon ≤ $$maxlon and
                                  Time ≥ $$mint and Time ≤ $$maxt]
```

A key difference between GtoPdb and MODIS is where the information needed to construct the citation is stored. In GtoPdb it is in the database, while in MODIS it is mostly kept elsewhere. This is easily solved by having functions in the citation rule that query an appropriate metadata repository with parameters extracted from the matching rule. For example, in Figure 5, m_auth() is a function that, given a product and version, queries the metadata for authorship. To our knowledge, there is currently no such organized metadata repository in MODIS, but having one would clearly be beneficial.

The version and access time (DATE function) are also not part of the view definition but can be calculated when the query is executed. Note that in

[11] This is approximately the green box in Figure 2.

```
{ author: m_auth($p,$$v), m_year:($p,$$v), title: m_title($p), version: $v,
  bounding-box : [$$minlong, $$minlat, $$maxlong, $$maxlat], interval: [$$mint, $$maxt],
  organization: m_org($p), url: m_url($p), accessed: DATE(), doi = m_doi($p,$$v}
⟵
/root/product[ProdName=$p]/version[vnum=$$v]
      /file[Lat ≥ $$minlat and Lat ≤ $$maxlat and
            Lon ≥ $$minlon and Lon ≤ $$maxlon and
            Time ≥ $$mint and Time ≤ $$maxt]

{ author: "E. Vermote", title: "MOD09A1 ... SIN Grid", version: 6,
  bounding-box: [-125, 32, -119, 42], interval: [2008-01-25, 2008-01-25],
  organization: "NASA EOSDIS ... South Dakota", URL: "https://lpdaac.usgs.gov",
  accessed: "2015-09-01", doi: "10.5067/MODIS/MOD09A1.006" }
```

Figure 5: A citation rule for MODIS

MODIS, when newer analysis software becomes available, the entire database of products is re-analyzed yielding a complete new version; old versions are not kept. While this is undesirable from the standpoint of provenance and reproducibility, the citation carries useful information even though its referent may not exist.

4 Conclusions

We have addressed a critical issue in the adoption of data citation: automatically generating a citation from the query and database that was used to obtain the data. A preliminary implementation of the rule-based citation language for hierarchical data is reported in [3]. What we describe here is quite general and applies to any database with a well-defined query language. Rewriting queries through views was originally developed for query optimization and subsequently exploited in data integration. The idea of using them for data citation bears some relationship to that of using them to define security levels in a database [6].

We believe that using database views to specify citable units is key to both specifying and generating citations. It is important for any data publisher who wants their data to be properly cited to define these views, and to ensure that the data necessary to generate the citation from them is available. We have shown how this can be done for two quite different scientific databases, and we believe that the idea can work on forms of data such as RDF [22] and databases that are deployed in other fields such as the humanities. We have looked briefly at some examples, and the main issue is that the data needed to generate the citation may not be available, either in the database or some metadata repository[12].

[12] See the use cases and linked data sections of Appendix A

We focussed on one specific computational problem in this paper, but it is almost impossible to do this in isolation from other topics such as citation standards. For example, the citation snippets required by the curators of our two examples do not quite conform to the DataCite metadata schema [4]: although DataCite has an entry for a spatial bounding box, it does not have one for a temporal interval as required by MODIS. It is an interesting problem to check that a citation rule generates results that are consistent with a given citation schema.

We also mentioned archiving (fixity) and provenance as related computational challenges, but there are many others. We have tacitly assumed a rather conventional view of citations and how they will be used, but there are many ways in which this may radically change, e.g., the 10,000 author paper or the paper with 10,000 references. Maybe, by analogy with PageRank [17], there should be some notion of transitivity of credit in citation. These are all likely to require new ideas from computer science.

Acknowledgements: Tony Harmar, who led the development of GtoPdb, introduced us to the problem of data citation. We are also indebted to Sarah Cohen Boulakia, Jamie Davies, Wenfei Fan, Andreas Rauber, Joanna Sharman, Gianmaria Silvello and the reviewers much useful input. This work is supported by NSF IIS 1302212 and EPSRC SOCIAM EP/J017728/1.

References

[1] Micah Altman and Gary King. A proposed standard for the scholarly citation of quantitative data. *D-Lib Magazine*, 13(3/4), March/April 2007.

[2] Alex Ball and Monica Duke. How to cite datasets and link to publications. `http://www.dcc.ac.uk/resources/how-guides/cite-datasets`, June 2012.

[3] Peter Buneman and Gianmaria Silvello. A rule-based citation system for structured and evolving datasets. *IEEE Data Eng. Bull.*, 33(3):33–41, 2010.

[4] DataCite. DataCite Metadata Schema for the Publication and Citation of Research Data. `http://schema.datacite.org/meta/kernel-3/doc/DataCite-MetadataKernel\_v3.1.pdf`, October 2014.

[5] Alin Deutsch, Lucian Popa, and Val Tannen. Query reformulation with constraints. *SIGMOD Record*, 35(1):65–73, 2006.

[6] Wenfei Fan, Chee-Yong Chan, and Minos Garofalakis. Secure xml querying with security views. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 587–598. ACM, 2004.

[7] Data Observation Network for Earth (DataONE). Data citation and attribution. https://www.dataone.org/citing-dataone.

[8] International Council for Science (ICSU) Committee on Data for Science and Technology (CODATA). Data citation standards and practices. http://www.codata.org/task-groups/data-citation-standards-and-practices,
2010.

[9] FORCE11. Data citation synthesis group: Joint declaration of data citation principles. https://www.force11.org/datacitation, 2014.

[10] Alon Y. Halevy. Answering queries using views: A survey. *VLDB J.*, 10(4):270–294, 2001.

[11] Bryan Lawrence, Catherine Jones, Brian Matthews, Sam Pepler, and Sarah Callaghan. Citation and peer review of data: Moving towards formal data publication. *International Journal of Digital Curation*, 6(2):4–37, 2011.

[12] Maurizio Lenzerini. Data integration: A theoretical perspective. In *Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, PODS '02, pages 233–246, New York, NY, USA, 2002. ACM.

[13] I. McCallum, H.-P. Plag, and S. Fritz. GEOSS data citation guidelines: Version 2.0. http://www.gstss.org/library/GEOSS_Data_Citation_Guidelines_V2.0.pdf, October 2012.

[14] Federation of Earth Science Information Partners (ESIP). Data citation guidelines for data providers and archives. http://doi.org/10.7269/P34F1NNJ.

[15] CODATA-ICSTI Task Group on Data Citation Standards and Practices. Out of cite, out of mind: The current state of practice, policy, and technology for the citation of data. *Data Science Journal*, 12:CIDCR1–CIDCR75, 2013.

[16] Coalition on Publishing Data in the Earth and Space Sciences (COPDESS). Statement of commitment from earth and space science publishers and data facilities. http://www.copdess.org/wp-content/uploads/2015/01/statementofcommitment.pdf, January 2015.

[17] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, November 1999. Previous number = SIDL-WP-1999-0120.

[18] Adam J Pawson, Joanna L Sharman, et al. The IUPHAR/BPS Guide to PHARMACOLOGY: an expert-driven knowledgebase of drug targets and their ligands. *Nucleic acids research*, 42(D1):D1098–D1106, 2014.

[19] Stefan Pröll and Andreas Rauber. Scalable data citation in dynamic, large databases: Model and reference implementation. In *Proceedings of the 2013 IEEE International Conference on Big Data*, pages 307–312, 2013.

[20] Stefan Pröll and Andreas Rauber. A scalable framework for dynamic data citation of arbitrary structured data. In *DATA 2014 - Proceedings of 3rd International Conference on Data Management Technologies and Applications, Vienna, Austria, 29-31 August, 2014*, pages 223–230, 2014.

[21] Research Data Alliance (RDA) Working Group on Data Citation. Making data citable: Case statement. `https://rd-alliance.org/group/data-citation-wg/case-statement/wg-data-citation-making-data-citable-case-statement.html`.

[22] Gianmaria Silvello. A methodology for citing linked open data subsets. *D-Lib Magazine*, 21(1/2), 2015.

[23] American Meteorological Society. Data archiving and citation. `http://www2.ametsoc.org/ams/index.cfm/publications/authors/journal-and-bams-authors/journal-and-bams-authors-guide/data-archiving-and-citation/`.

[24] American Geophysical Union. AGU publications data policy. `http://publications.agu.org/author-resource-center/publication-policies/data-policy/`, December 2013.

Appendix A An annotated bibliography

Many of these references were suggested by participants in the workshop on "Computational Challenges in Data Citation" held at the University of Pennsylvania on 17-18 April 2014.[13]

Please note that all the references in the appendices are to papers listed in the appendix, some of which also appear in the references of the main paper.

Standards and Principles

A1 Rauber, A., Pröll, S. Scalable Dynamic Data Citation. Position Paper, Working Group on Data Citation (WG-DC), Research Data Alliance (RDA), 2015-03-23 draft version. `http://rd-alliance.org/groups/data-citation-wg/wiki/scalable-dynamic-data-citation-rda-wg-dc-position-paper.html`

generalizes [B4, B5]

A2 Uhlir, P.F. (Ed.) *For Attribution- Developing Data Attribution and Citation Practices and Standards.* National Academies Press, Washington, DC, 2012. `http://doi.org/10.17226/13564`

workshop summary

A3 Bilder, G. DOIs unambiguously and persistently identify published, trustworthy, citable online scholarly literature. Right? Crossref Blog, September 20, 2013. `http://blog.crossref.org/2013/09/dois-unambiguously-and-persistently-identify-published-trustworthy-citable-online-scholarly-literature-right.html`

explores edge cases in DOI definition and registry implementations

A4 Chavan, V.S., Ingwersen, P. Towards a data publishing framework for primary biodiversity data: challenges and potentials for the biodiversity informatics community *BMC Bioinformatics 10(Suppl 14),* S2 (November 10, 2009). `http://doi.org/10.1186/1471-2105-10-S14-S2`

proposes an architecture for a distributed data publishing environment

A5 Mooney, H., Newton, M. The Anatomy of a Data Citation. *Journal of Librarianship and Scholarly Communication 1,* 1 (2012), eP1035. `http://doi.org/10.7710/2162-3309.1035`

argues that information professionals must promote data citation as "an essential component of data publication, sharing, and reuse."

[13] `http://datacitation.eri.ucsb.edu`

A6 Allen, L., Scott, J., Brand, A., Hlava, M., Altman, M. Credit where credit is due. *Nature 508* (17 April 2014), 312–313. http://doi.org/10.1038/508312a.

introduces a taxonomy of contributor roles, to facilitate fine-grained assignment of citation credit

Citation systems design and implementation

B1 Aalbersberg, I.J., Kähler, O. Supporting Science through the Interoperability of Data and Articles. *D-Lib Magazine 17,* 1/2 (January/February 2011). http://doi.org/10.1045/january2011-aalbersberg

discusses Elsevier's SciVerse ScienceDirect architecture

B2 Buneman, P. How to cite curated databases and how to make them citable. In *SSDBM 2006: 18th International Conference on Scientific and Statistical Database Management* (Vienna, 3–5 July 2006). IEEE Computer Society, Los Alamitos, CA, 2006, 195–203. http://doi.org/10.1109/SSDBM.2006.28

the original work on IUPHAR citation

B3 Altman, A., Crosas, M. The Evolution of Data Citation: From Principles to Implementation. *IASSIST Quarterly 37,* 1-4 (2013), 62–70. http://www.iassistdata.org/iq/evolution-data-citation-principles-implementation

background of the FORCE11 Data Citation Principles

B4 Pröll, R., Rauber, A., Scalable data citation in dynamic, large databases: Model and reference implementation. *Proceedings of the 2013 IEEE International Conference on Big Data*, pages 307–312, 2013.

B5 Pröll, R., Rauber, A., A scalable framework for dynamic data citation of arbitrary structured data. *DATA 2014 - Proceedings of 3rd International Conference on Data Management Technologies and Applications, Vienna, Austria, 29-31 August, 2014*, pages 223–230, 2014.

Data Citation and Linked Data

C1 Berners-Lee, T. Linked Data. (June 18, 2009); http://www.w3.org/DesignIssues/LinkedData.html.

rules for publishing data on the Web

C2 Berners-Lee, T. Cool URIs Don't Change. (1998); http://www.w3.org/Provider/Style/URI.html.

notes on designing URIs for stability and longevity

C3 Ayers, D., Völkel, M. Cool URIs for the Semantic Web. (December 3, 2008); http://www.w3.org/TR/cooluris/.

guidelines for using URIs with RDF

C4 Thompson, H.S. Naming on the Web: What scholars should want, and what they can have. In *CERN Workshop on Innovations in Scholarly Communication (OAI8)* (University of Geneva, 19–21: June 2013). http://indico.cern.ch/event/211600/session/3/contribution/2/attachments/331924/463111/scroll.pdf

overview of naming issues: binding, resolution, and management

C5 Heath, T., Bizer, C. Linked Data: Evolving the Web into a Global Data Space (1st edition). *Synthesis Lectures on the Semantic Web: Theory and Technology 1,* 1 (2011), 1–136. http://doi.org/10.2200/S00334ED1V01Y201102WBE001.

see chapter 6 "Consuming Linked Data", section 6.4 "Effort Distribution between Publishers, Consumers and Third Parties"

C6 Memento Guide - Resource Versioning and Memento. (January 19, 2015); http://mementoweb.org/guide/howto/

supporting the functionality of time-versioned URIs with the Memento protocol

C7 Van de Sompel, H., Nelson, M. Thoughts on Referencing, Linking, Reference Rot. (December 28, 2013); http://mementoweb.org/missing-link/.

requirements for robust citations to web resources

Data Citation and Reproducibility

D1 Peng, R. Reproducible Research in Computational Science. *Science 334,* 6060 (2 December 2011), 1226–1227. http://doi.org/10.1126/science.1213847.

introduction to the reproducibility problem

D2 Schopf, J. Treating data like software: a case for production quality data. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries* (Washington, DC, June 10–14, 2012). ACM, New York, 2012, 153–156. http://doi.org/10.1145/2232817.2232846.

argues that software release engineering principles should be applied to data

D3 Mesirov, J.P. Accessible Reproducible Research. *Science 327,* 5964 (22 January 2010), 415–416. http://doi.org/10.1126/science.1179653.

proposes a generic framework for supporting reproducible computational science

D4 Alper, P., Belhajjame, K., Goble, C., Karagoz, P. Enhancing and abstracting scientific workflow provenance for data publishing. In *Proceedings of the Joint EDBT/ICDT 2013 Workshops* (Genoa, Italy, March 18–22, 2013). ACM, New York, 2013, 313–318. http://doi.org/10.1145/2457317.2457370.

discusses the relationship between provenance and data citation

Use Cases

In this section we have included additional references to the examples in the main paper. In addition we have added references to databases that may present further challenges. Many RDF databases have been extracted from existing data sets, and in this process the data and metadata needed for citation have been lost; however the Experimental Factors Ontology [E5] is directly represented in RDF and presents an interesting challenge. Also it has been suggested to us that the Encoded Archival Description [E6] presents some interesting aspects of citation for semistructured data.

E1 Southan, C., Sharman, J.L., Benson, H.E., Faccenda, E., Pawson, A.J., Alexander, S.P.H., Buneman, P., Davenport, A.P., McGrath, J.C., Peters, J.A., Spedding, M., Catterall, W.A., Fabbro, D., Davies, J.A. The IUPHAR/BPS Guide to PHARMACOLOGY in 2016: towards curated quantitative interactions between 1300 protein targets and 6000 ligands. *Nucleic Acids Research* Advance Access (October 12, 2015). http://doi.org/10.1093/nar/gkv1037.

the most recent version of the GtoPdb (formerly IUPHAR) database

E2 NASA. MODIS Web. (2015); http://modis.gsfc.nasa.gov/.

main website for the MODIS instruments and data products

E3 NASA EOSDIS Land Processes DAAC, USGS Earth Resources Observation and Science (EROS) Center. Citing Our Data. (April 14, 2014); http://lpdaac.usgs.gov/citing_our_data.

how to cite USGS MODIS data

E4 Oak Ridge National Laboratory (ORNL) DAAC. Data Product Citation Policy. (2015); http://daac.ornl.gov/citation_policy.html.

how to cite ORNL MODIS data

E5 Malone, J., Holloway, E., Adamusiak, T., Kapushesky, M., Zheng, J., Kolesnikov, N., Zhukova, A., Brazma, A., Parkinson, H. Modeling Sample Variables with an Experimental Factor Ontology. *Bioinformatics 26,* 8 (2010), 1112–1118. `http://doi.org/10.1093/bioinformatics/btq099`.

an ontology for gene expression data

E6 Pitti, D.V., Encoded archival description: The development of an encoding standard for archival finding aids. *The American Archivist*, pp.268-283. 1997.

E7 Chavan, V. *Recommended practices for citation of data published through the GBIF network. Version 1.0.* Global Biodiversity Information Facility, Copenhagen, 2012. `http://links.gbif.org/gbif_best_practice_data_citation_en_v1`

how to cite global biodiversity data

E8 Duerr, R.E., Downs, R.R., Tilmes, C., Barkstrom, B., Lenhardt, W.C., Glassy, J., Bermudez, L.E., Slaughter, P. On the utility of identification schemes for digital earth science data: an assessment and recommendations. *Earth Science Informatics 4,* 3 (September 2011), 139–160. `http://doi.org/10.1007/s12145-011-0083-6`.

applicability of multiple identifier schemes for citing Earth science data

Background reading on Databases and XML

Most good textbooks on databases (e.g. [F1, F2]) cover both relational databases and the basics of XML. There is also plenty of on-line material related to these. Of relevance to the ideas in this paper are the systems that convert or represent relational databases in some hierarchical form such as XML. [F3] describes a sophisticated approach to this and also reviews what is practically available. Going in the other direction, a number of database systems provide for ingesting XML into tables, see [F4,F5].

F1 Ramakrishnan, R., Gehrke, J. *Database Management Systems (3rd Edition).* McGraw-Hill, 2002.

F2 Garcia-Molina, H., Ullman, J.D., Widom, J. *Database Systems: The Complete Book (2nd Edition)* Pearson, 2008.

F3 Benedikt, M., Chan, C.Y., Fan, W., Rastogi, R., Zheng, S., Zhou. A. DTD-directed publishing with attribute translation grammars. *VLDB*, 2002.

F4 Bohannon, P., Freire, J., Haritsa, J. R., Roy, P., Siméon, J. LegoDB: Customizing relational storage for XML documents. In *VLDB* 2002

F5 Teradata Database, Tools and Utilities Release 15.00 http://www.info.teradata.com/htmlpubs/DB_TTU_15_00/index.html#page/Teradata_XML/B035_1140_015K/XML_Shredding_Publishing.09.01.html

Views

G1 Fan, W. Chan, C.Y., Garofalakis, M.N. Secure XML Querying with Security Views. *SIGMOD* 2004.

G2 Halevy, A.Y. Answering queries using views: A survey. *VLDB J.*, 10(4):270–294, 2001.

G3 Abiteboul,S., Duschka, O.M. Complexity of Answering Queries Using Materialized Views. *PODS 1998* 1998.

G4 Deutsch, A., Popa, L., Tannen, V. Query reformulation with constraints. *SIGMOD Record*, 35(1), 2006.

G5 Lenzerini, M. Data Integration: A Theoretical Perspective. *PODS 2002*, 233–246, 2002

Archiving

Some relational database management systems provide for *time travel*, originally proposed in [H1] – the ability to see the database at any point in the past. Moreover this can be queried through a temporal extension of SQL [H2]. Such temporal extensions are now part of the SQL2011 standard and have been implemented by systems such as DB2. Unfortunately database developers often fail to make use of them. Moreover there is a concern that the long-term preservation of a database should be dependent on the maintenance of relatively complex database software. Work by Rauber on provenance and citation, cited in the main paper, provides an archiving system for tabular data.

For hierarchical data, [H3, H4] provide an approach that works by pushing temporal variation down into the hierarchy. Full web archiving [H5] will be of enormous benefit, but for our purposes this will need to be extended to the "deep" Web.

H1 Stonebraker, M. and Kemnitz, G. The POSTGRES next generation database management system. *Communications of the ACM*, 34(10), pp.78-92. 1991

H2 Snodgrass, R.T. *Developing Time-Oriented Database Applications in SQL* Morgan Kaufmann. 1999.

H3 Buneman, P., Khanna, S., Tajima, K. and Tan, W.C. Archiving scientific data. ACM Transactions on Database Systems *TODS*, 29(1), pp.2-42. 2004

H4 Müller, H., Buneman, P. and Koltsidas, I., XArch: archiving scientific and reference data. *ACM SIGMOD 2008* (pp. 1295-1298). 2008.

H5 Van de Sompel, H., Nelson, M.L., Sanderson, R., Balakireva, L.L., Ainsworth, S. and Shankar, H. Memento: Time travel for the web. *arXiv preprint arXiv:0911.1112.* 2009

Appendix B Citable units

Although the authors have heard the term "citable unit" in widespread use at meetings, there does not appear to be any authoritative description of the term. The term is used in some on-line blogs (themselves difficult to cite), but these appear to take the term as given rather than defining it. `http://serialmentor.com/blog/2015/1/2/what-constitutes-a-citable-scientific-work/` provides some criteria for a citation and appears to use "citable unit" to describe what we term the referent of the citation. `https://www.aje.com/en/author-resources/articles/nanopublications-and-mini-monographs` takes the term as given in order to describe a *nanopublication*.

We do not think that citable units coincide with the referents of DOIs. In our examples, the number of possible citatable units far exceeds the number of DOIs one would want to assign to a database. Also there is a subtlety, which we did not address, about a possible distinction between a citable unit and a citation. A citation such as "John Doe, The Impossibility of Reason, Elsinger 1954, chapter 3, paragraph 17" claims that the moon is made of green cheese." contains what one might regard as a citable unit – "John Doe, The Impossibility of Reason, Elsinger 1954" and a location – "chapter 3, paragraph 17" at which one finds the claim. Although the location is not really part of the citable unit, it is an invaluable part of the citation itself. Providing such location information is especially important for the substantial number of people who are employed as data curators to verify that citations are correct, in the sense that cited material has been correctly interpreted in the database. Finding where a particular claim is made in a long paper can be very time-consuming.

The provision of location information in a citation fits well with the machinery we have presented, but we did not discuss this in the paper.

Appendix C Views

The original motivation for answering queries using views was for efficiency: suppose one has already computed the answers (called *materialized views*) to one or more queries agains a database. Given a new query, it may be more efficient to compute the answer to that query using those views rather than applying it directly to the database. This presupposes that the query can be

factored through those views. A second stimulus comes from data integration and is closely linked to partial or incomplete information in databases. These are explained in a survey paper [G2]. The former stimulus, answering queries using materialized views, is closer to the problem presented in this paper. Another use of views, which does connect with our proposals, is for security: The "publishers" of the database associate a security level with each view. The security level needed to answer a query is determined by how the query can be answered using views. See [G1] for details

There is a huge literature on this topic. Complexity issues are treated in [G3]. Also relevant to data citation are how to rewrite in the presence of constraints [G4] and how it might work in other (non relational) representations of data [G5].