Micro Archives as Rich Digital Object Representations*

Helge Holzmann
L3S Research Center
30167 Hannover, Germany
holzmann@L3S.de

Mila Runnwerth
German National Library
of Science and Technology (TIB)
30167 Hannover, Germany
Mila.Runnwerth@tib.eu

ABSTRACT

Digital objects as well as real-world entities are commonly referred to in literature or on the Web by mentioning their name, linking to their website or citing unique identifiers, such as DOI and ORCID, which are backed by a set of meta information. All of these methods have severe disadvantages and are not always suitable though: They are not very precise, not guaranteed to be persistent or mean a big additional effort for the author, who needs to collect the metadata to describe the reference accurately. Especially for complex, evolving entities and objects like software, pre-defined metadata schemas are often not expressive enough to capture its temporal state comprehensively. We found in previous work that a lot of meaningful information about software, such as a description, rich metadata, its documentation and source code, is usually available online. However, all of this needs to be preserved coherently in order to constitute a rich digital representation of the entity. We show that this is currently not the case, as only 10% of the studied blog posts and roughly 30% of the analyzed software websites are archived completely, i.e., all linked resources are captured as well. Therefore, we propose Micro Archives as rich digital object representations, which semantically and logically connect archived resources and ensure a coherent state. With Micrawler we present a modular solution to create, cite and analyze such Micro Archives. In this paper, we show the need for this approach as well as discuss opportunities and implications for various applications also beyond scholarly writing.

KEYWORDS

Web Archives; Crawling; Data Representation; Scientific Workflow

ACM Reference Format:
Helge Holzmann and Mila Runnwerth. 2018. Micro Archives as Rich Digital Object Representations. In *Proceedings of 10th ACM Conference on Web Science (WebSci '18).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3201064.3201110

*This work is partly funded by the European Research Council under ALEXANDRIA (ERC 339233)

WebSci '18, May 27–30, 2018, Amsterdam, Netherlands

ACM ISBN 978-1-4503-5563-6/18/05…$15.00
https://doi.org/10.1145/3201064.3201110

1 INTRODUCTION

In the area of digital libraries and in the scholarly domain in general exist many digital identifiers used to reference objects and entities in literature, most prominently, the *Digital Object Identifier* (DOI)[13]. These identifiers are commonly backed by a set of metadata that describe the referenced object. While meta information are easy to create and maintain for fixed objects, such as scientific publications, which do not change anymore after they have been published and assigned their DOI, this approach does not scale well for more dynamic entities.

As one such subject, we consider software, an omnipresent good in science that is often referenced in literature. Software is constantly being developed and can have a different state in every moment, especially if it is open source and being developed by a large community. In such cases, it is difficult to permanently keep corresponding metadata up to date. Even more challenging, a software that is developed by thousands of developers, with every developer working on a small piece of it, is nearly impossible to be precisely expressed by a fixed set of metadata values. Further is such a representation in many cases not what a reader requires to fully understand the referenced asset. Way more useful would be a description, documentation, or even the source code in case of software. We found in previous work that most of these information already exist on the Web [12].

From an author's perspective who wants to reference some entity or object that is not explicitly prepared for this, the collection of all required meta information to comprehensively describe the referenced asset means a big additional effort. Instead, we often see very vague references in literature, e.g., only a name, sometimes with the version or date. Similarly, references to Web resources, such as blog articles, are made as a footnote containing the URL. However, even if the date of visit is specified, this is not very helpful as the referenced blog post or linked resources may already have changed by the time it is read.

Many of these problems could be solved if we had richer presentations of the cited objects. If the reader does not only see the name, version and author of a referenced software, but can actually read the documentation at the time when the author accessed it. For that reason, we propose *Micro Archives*: microscopic collections of archived resources on the Web that describe a single entity or object, cohesively preserved for future reference. While existing Web archives already provide the necessary infrastructures to preserve all required resources individually, Micro Archives can be considered a logical and semantic connection of such resources to provide a holistic view onto a cited object. Furthermore, metadata that may be available in unstructured or semi-structured form as part of such a Micro Archive can be dynamically extracted and presented as needed whenever required.

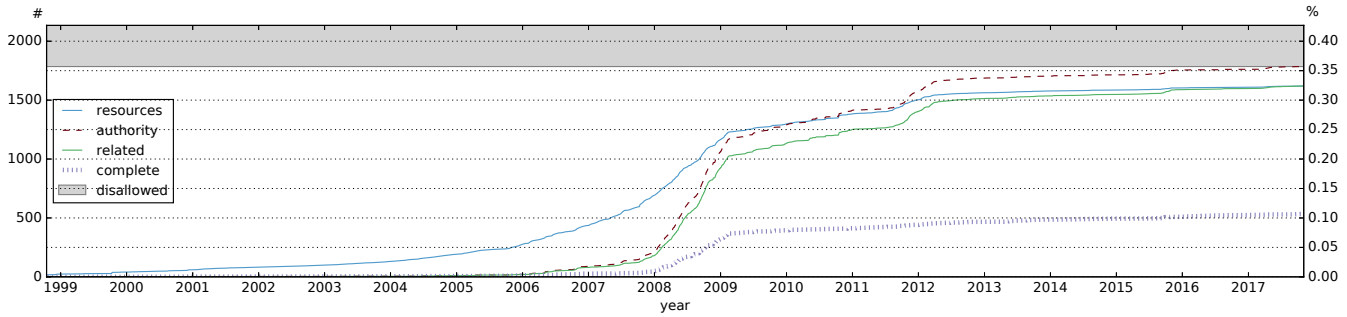

Figure 1: Web Archive Timeline: Blogs.

In the following, we present Micrawler, a modular proof-of-concept prototype that implements the entire pipeline of creating, archiving, analyzing, presenting and citing Micro Archives, along with a practical example of how our approach can be used within the scientific publication workflow. Further, we showcase two use case scenarios, i.e., 1) blog articles, 2) software, which we have investigated in terms of inconsistencies that could be fixed with Micrawler in the future. Finally, we will highlight the opportunities created by Micro Archives in various areas and stress why we think the presented concepts are an inevitable step in our digital world.

2 RELATED WORK

Piwowar et al. [21] provided evidence that enhanced access to research data lead to an increased number of citations. Although there has been quite some work on research data and its use in literature [17, 18, 22] as well as on Web archives as containers for cultural, personal or scientific entities [15, 19, 20], there is not much on combining both aspects as we intent with our work. Dynamic research data, such as software, has been neglected for a long time because of its volatility and its development process that cannot be suitably mapped by traditional metadata. Only recently, several initiatives have emerged to foster the use of software in a scientifically sound manner, such as the *Software Sustainability Institute*, *Software Heritage* or *FORCE11*[1][4, 7, 8, 23]. However, we are the first to propose the incorporation of Web archives for this purpose.

Web archives have been of growing interest as they allow to explore the Web with regard to a dimension that is often neglected in common tasks, like search and entity linking, but also the use of the Web in science: time. These valuable collections allow to study the Web and its development over time [2, 10]. Further, it has become a dire need to preserve scientific information before it vanishes from the Web [3, 16, 24]. However, access capabilities are still limited [6]. Works that attempt to improve this, deal with the efficient processing of Web archive data at scale [9] as well as temporal search and ranking [5, 11]. While these approaches can be used to retrieve temporally relevant and related resources for a given entity in an automatic manner, Micro Archives aim at making such semantic, temporal connections more explicit and sustainable.

3 CASE STUDIES

We have investigated two use case scenarios for which Micro Archives would immediately create a major benefit in their scientific use, i.e., blog articles and software. The question we raise is: How complete and coherent is the archived Web with respect to related resources linked on the corresponding webpages? Micrawler can improve the coherence of Web archives by making sure for an object or entity cited today, all related resources are archived today as well, resulting in a Micro Archive.

3.1 Datasets and Methodology

The retrospective analysis of blog articles was done using the *TREC Blogs'08*[2] collection. This corpus consists of 28,488,766 blog posts, collected between 2007 and 2008 for the *TREC 2008 Blog Track*. Hence, we can assume the blog articles to be published during that time period. Although some older ones are included as well, there are definitely no posts composed later than Feb 2009.

As it is more difficult to relate software to a specific point in time, we study its state as of today. For this analysis, we collected all 22,022 URLs[3], each corresponding to a single software, as listed on *swMATH*[4], a catalog and information service for mathematical software.

All webpages linked from any of the processed URLs are considered related. Although maybe not complete, we found that many software websites link to corresponding documentation, artifacts, source code and other related artifacts from their homepage [12]. These resources were gathered from the archived snapshot of the corresponding software or blog page. In case of software, we picked the latest captures, and for the retrospective study of blog articles, we picked the earliest snapshot that was available in the Internet Archive's Wayback Machine[5].

As the process of retrieving an archived snapshot for an URL with all its linked resources is quite time consuming, we limited our analysis to a random sample of 5,000 objects from each dataset. A single unit of 1 represents a completely archived object with all related resources, the percentage is relative to these. Partially archived objects would be represented by a corresponding floating

[1] https://www.force11.org/about/manifesto
[2] http://ir.dcs.gla.ac.uk/test_collections/blogs08info.html
[3] state at Dec 7, 2017
[4] http://www.swmath.org
[5] http://web.archive.org

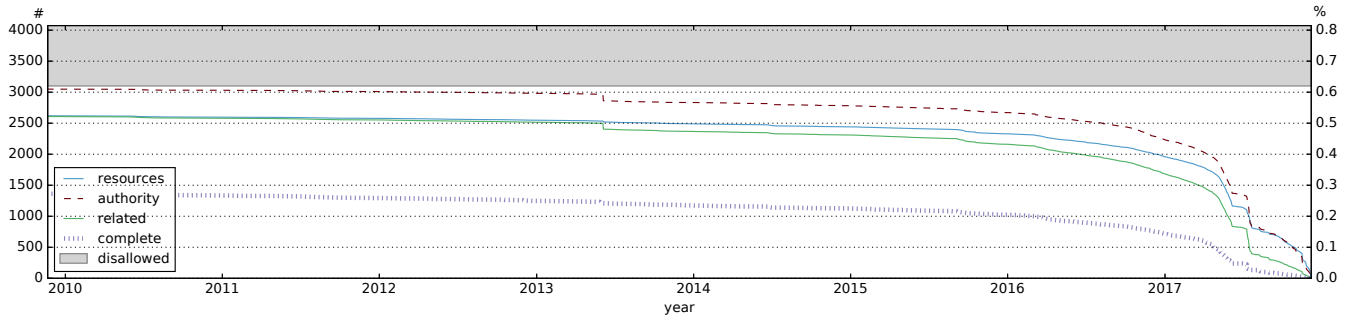

Figure 2: Web Archive Timeline: Software.

point unit. For better readability, the plots have been limited to the 2,134 blog posts and 4,074 software websites for which at least the authority page available, i.e., the actual blog post or representative webpage of a software. Together with the related sources we ended up with a total of 243,336 URLs that we had to fetch for blogs and 123,060 URLs for software, resulting in 48 related resources per blog article and 24 related resources per software on average.

Another fraction was covered by the Web archive, but disallowed themselves from being archived through a policy specified in their `robots.txt`. For these, the corresponding objects could not be studied, neither can they be captured with our proposed approach. There are depicted in our plots by the gray bar at the top. For an authority that is archived, but that links to pages that are disallowed, these related resources were ignored.

Each plot contains four lines to show the coverage of the studied objects in the Web archive over time: **resources** represents an object as fraction of its archived resources, **authority** considers the authority pages only, **related** denotes the fraction of *resources* for an object only if the *authority* is archived, and **complete** shows the completely archived ones.

3.2 Results: Blogs

The timeline in Figure 1 shows the results of our study of blog articles. Due to the time of the dataset, which was collected around year 2008, we can observe a major growth in the archive around this time as expected. However, as shown by the resources line, some of the related resources were already preserved long before the blog posts were published, e.g., in 2006 around 5% of the links in an article on average. This makes sense as they have to be online before they are referenced by a blog.

The steep increase of the archived resources to 25% together with the growth of the actual articles (authority pages) indicates that the blogs reference rather recent resources, assuming that they were captured by the archive not too long after publication. This is encouraged by the fact that they were archived slightly before the blog posts, hence, the archive discovered them not through the articles but independently of them.

Once the authority URLs are archived as shown by the dashed line, the related resources go up as well, suggesting these were already archived before that point. However, although this is a positive finding, it only goes from around 20% at the beginning of 2009 to slightly over 30% today on average for the resources related to the archived authorities, a unfortunately small fraction. The gap to the completely archived articles stays rather large and only reaches about 10% today. This makes us wonder whether actually a coherent and useful impression of the archived blog articles with their hyperlinked references can be obtained from the studied Web archive.

3.3 Results: Software

Software on the other hand was studied from its current state, going back until the latest snapshot of a resource had been archived. Positive is the steep growth on the very right of the timeline, resulting in almost 50% of all software authority websites archived already only about one year back from now, at the beginning of 2017. Unfortunately, there is not much gain by going back in time and even in 2010 and before not more than slightly over 60% are archived overall. Similar to blogs, the line of complete snapshots is rather low. A noticeable difference to the timeline of blogs is that the lines of overall resources and related resources are much closer at any time. That means only a few related resources are recaptured more recently than the corresponding authority page. In contrast to blogs, it is quite likely that these are only discovered by the archive crawler through the software websites.

4 USE CASE SCENARIO

As our case studies have shown, the coherence among related resources in Web archives is not sufficient to reference a consistent state of the represented object. This is what we intent to improve with the introduction of Micro Archives. The following steps outline a common workflow to create and cite a Micro Archive.

Specifying Micro Archives. In order to use a Micro Archive as digital representation of any object, it first needs to be defined. Anyone can specify a Micro Archive with the required set of resources: their URL along with labels and possibly comments. A Micro Archive specification should include the name of the represented object as well as additional properties, such as the type, e.g., blog, software, person, company, etc.. Such crawl specifications can be shared, refined as well as reused. Predefined specifications can be provided or extracted from suitable services, such as repositories or directories, accessible through a dedicated link to cite included items. In case of software, this could be any service that is aware of the relevant URLs, such as a software catalogs like *swMath* (s. Section 3.1). A click on this cite link could immediately trigger the

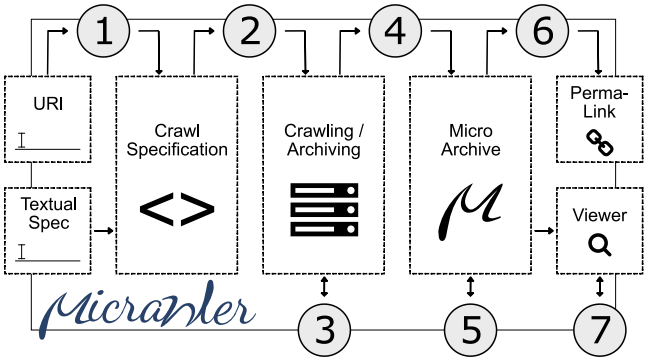

Figure 3: Micrawler Architecture and Extension Points / Related Services.

archiving process (using a software like Micrawler, s. Section 5). To create a Micro Archive of a blog post, the specification can be automatically derived from the links in the post itself.

Crawling / Archiving. Based on the given crawl specification all related resources should be crawled and archived at the same time or with as little delay as possible. Whether only the given URLs are captured or used as seeds for a broader crawl depends on the type of application. The archiving process can be performed by any Web archive, treating each resource as an independent item. Depending on the type of resource, even different archives may be used, like Web archives for webpages, but more software-specific archive for the raw source code. The resulting Micro Archive now serves as an additional layer that connects these captured resources and takes care of a coherent state among them.

Presentation / Citing. Once created, the Micro Archive is anchored to the time when it was crawled and represents the corresponding object or entity through the resources that were part of the specification. For future reference, a unique handle that is assigned to the Micro Archive, would now be sufficient to cite the preserved state of the represented object. This may be a short URL or more specific identifiers, such as a DOI or others.

5 MICRAWLER

Micrawler (*Micro Crawler*) is a reference implementation and proof-of-concept prototype to perform the aforementioned steps of creating and citing Micro Archives. It runs the entire pipeline from specifying over crawling to citing and analyzing Micro Archives. Figure 3 gives an overview of the steps performed by Micrawler and how these connect to the modules as explained in the following. The codebase of Micrawler is open source and published under https://github.com/helgeho/Micrawler. The running prototype has been deployed to http://tempas.l3s.de/Micrawler.

(1) **Spec Proxy:** A specification (spec) of what to crawl/preserve can be provided to Micrawler textually or a URI to load/extract a spec from. The *spec proxy* is in charge of deriving the textual spec from the given resource. Our current prototype implements a few special cases, such as software listed on *swMATH* (s. Section 3.1), for which a corresponding spec is generated from the included software website and linked resources.

(2) **Crawl Queue:** While in many cases the exact list of URLs as provided by the spec is crawled, this service allows to amend this list just before the crawl is started, e.g., to include deep links into certain websites. For software with a GitHub page in the spec, our demo adds the corresponding URL to GitHub's metadata API to preserve these valuable information.

(3) **Archiving/Crawl Service:** Each URL in the queue is now sent to an archive to be preserved. Such a service may be the *Save Page Now* feature of the Internet Archive's Wayback Machine, which we use in the current implementation. Alternatively, each URL could be send to a different service, e.g., source code might be stored at a more specialized service, like *Software Heritage*[6].

(4) **Archive Meta Service:** After all resources in the queue have been preserved, the created Micro Archive is documented by enriching the original spec with corresponding metadata for each capture in the archive. The *archive meta service* retrieves this information, such as the exact timestamp from the used archive.

(5) **Analyzers:** For different types of archives, Micrawler can be configured with different analyzers, to dynamically identify and derive additional information of the archived entity from the archived resources, such as a version number in case a software or information about the author in case of blog articles.

(6) **Persistence Provider:** To be shared and cited, the created spec that describes a Micro Archive and points to the archived resources has to be stored persistently. In this step, the *persistence provider* should assign a persistent identifier to the Micro Archive and guarantee permanent access. Therefore, our current prototype should not be used in production. With the assigned identifier, Micrawler generates BibTEXand BibLATEXto be used scientific publications as follows [1]:

```
@misc{SageMath,
  title = {{SageMath}},
  type = {software},
  howpublished = {\url{http://tempas.l3s.de/micrawler/permalink/8bcbcec}},
  note ={Archived using Micrawler: 2018-01-10T09:03:35.000Z}
}
```

(7) **Viewer:** Depending on which archiving services are used, suitable viewers need to be configured accordingly. Web archives commonly provide an instance of the Wayback Machine to replay archived resources in its original state. Viewers are called and opened by Micrawler when a resource of a Micro Archive is clicked.

6 OUTLOOK AND OPPORTUNITIES

Our case study has shown that only 10% of the studied blog posts and roughly 30% of the analyzed software websites are archived completely, i.e., all linked resources are captured as well. With Micrawler and Micro Archives we presented novel concepts to increase these numbers in the future to enable coherent citations. While this is the primary use case, we see a lot of potential in such microscopic collections by establishing the missing semantic and logical link among the resources on the Web combined with a temporal embedding:

[6] https://www.softwareheritage.org

Supporting Web Archives. An infrastructure around Micrawler that allows for sharing and maintaining crawl specifications as well as existing Micro Archives in combination with a headless implementation that can be triggered programmatically may support Web archives by ensuring coherent snapshots at relevant times. For instance, such a database that is aware of the resources related to an entity would enable publishers or libraries to trigger a snapshot whenever a mention of the entity is detected in a new publication, e.g., all websites and social media accounts of a person can be captured whenever he or she is mentioned in the news. Web archives itself can incorporate this information to prioritize related resources of a page at crawl time as well as use it to improve their access capabilities.

Temporally Relevant Collections. A huge issue in the research field of *Temporal Information Retrieval* [14] and temporal Web archive search [11] is the lack of a ground truth dataset for temporally relevant search results of a query. Micro Archives as a first step towards structuring the Web as well as Web archives in a semantical way constitute exactly such collections for the corresponding entities as queries across time. Hence, a central, curated database as described above, which allows for the retrieval of existing Micro Archives along with the snapshots of related resources would be of importance for these applications and finally enable proper evaluation of temporal retrieval systems. In addition to this, these collection can also be of direct use for the users of Web archives to discover lost webpages from the past.

Structuring the Web. Micro Archives add a semantical as well as a logical structure to Web archives, which represent single entities or objects at different points in time. The identification of such structures along with the existence of archived snapshots for corresponding resources opens up new opportunities in studying the Web. For instance, Web graphs that are typically constructed based on single URLs, hosts or domains, may now be formed according to objects and entities based on their related resources. Scientists would be able to study relations among entities not just based on textual information, which are hard to extract, but based on related resources across time. The coherent snapshots ensure a temporal coverage and realistic topologies in the sub-graphs, which are currently widely broken due to the present incompleteness of Web archives.

Rich Information. A very ambitious and visionary aspect of Micro Archives, is the complete reconstruction of represented entities. Wikipedia is a great example of how entities can be represented on the Web. It is not only used for reading and learning about facts, but even to link and disambiguate entity mentions on the Web or in machine learning tasks. However, Wikipedia articles are not written from scratch, they are rather compiled of information found all around the Web, indicated by the many references in these articles. Thus, collections and temporal snapshots of related resources that are representative for an entity may allow for automatic generation of such articles or semantic representations like in knowledge bases. Furthermore, these representation are temporal and thus, can reflect the evolution of corresponding entities.

REFERENCES

[1] 2018. SageMath. http://tempas.l3s.de/micrawler/permalink/8bcbcec. (2018). Archived using Micrawler: 2018-01-10T09:03:35.000Z.

[2] Teru Agata, Yosuke Miyata, Emi Ishita, Atsushi Ikeuchi, and Shuichi Ueda. 2014. Life span of web pages: A survey of 10 million pages collected in 2001. In *JCDL*.

[3] Scott Ainsworth, Ahmed Alsum, Hany SalahEldeen, Michele C. Weigle, and Michael L. Nelson. 2011. How much of the web is archived?. In *JCDL*.

[4] Roberto Di Cosmo and Stefano Zacchiroli. 2017. Software Heritage: Why and How to Preserve Software Source Code. In *iPRES*.

[5] Miguel Costa, Francisco Couto, and Mário Silva. 2014. Learning Temporal-dependent Ranking Models. In *SIGIR*.

[6] Miguel Costa, Daniel Gomes, Francisco Couto, and Mário Silva. 2013. A Survey of Web Archive Search Architectures. In *WWW Companion*.

[7] Stephen Crouch, Neil Chue Hong, Simon Hettrick, Mike Jackson, Aleksandra Pawlik, Shoaib Sufi, Les Carr, David De Roure, Carole A. Goble, and Mark Parsons. 2013. The Software Sustainability Institute: Changing Research Software Attitudes and Practices. *Computing in Science and Engineering* 15 (2013).

[8] Simon Hettrick, Mario Antonioletti, Les Carr, Neil Chue Hong, Stephen Crouch, David De Roure, Iain Emsley, Carole Goble, Alexander Hay, Devasena Inupakutika, Mike Jackson, Aleksandra Nenadic, Tim Parkinson, Mark I Parsons, Aleksandra Pawlik, Giacomo Peru, Arno Proeme, John Robinson, and Shoaib Sufi. 2014. UK Research Software Survey 2014. (Dec. 2014).

[9] Helge Holzmann, Vinay Goel, and Avishek Anand. 2016. ArchiveSpark: Efficient Web Archive Access, Extraction and Derivation. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries (JCDL '16)*. ACM, New York, NY, USA, 83–92. DOI: https://doi.org/10.1145/2910896.2910902

[10] Helge Holzmann, Wolfgang Nejdl, and Avishek Anand. 2016. The Dawn of Today's Popular Domains: A Study of the Archived German Web over 18 Years. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries, JCDL 2016, Newark, NJ, USA, June 19 - 23, 2016*. 73–82. DOI: https://doi.org/10.1145/2910896.2910901

[11] Helge Holzmann, Wolfgang Nejdl, and Avishek Anand. 2017. Exploring Web Archives Through Temporal Anchor Texts. In *Proceedings of the 2017 ACM on Web Science Conference - WebSci '17*. ACM Press. DOI: https://doi.org/10.1145/3091478.3091500

[12] Helge Holzmann, Wolfram Sperber, and Mila Runnwerth. 2016. Archiving Software Surrogates on the Web for Future Reference. In *TPDL*.

[13] ISO. 2012. 26324: 2012 Information and Documentation-Digital Object Identifier System. (2012).

[14] Nattiya Kanhabua, Roi Blanco, Kjetil Nørvåg, and others. 2015. Temporal information retrieval. *Foundations and Trends® in Information Retrieval* 9, 2 (2015), 91–208.

[15] Nikos Kasioumis, Vangelis Banos, and Hendrik Kalb. 2014. Towards building a blog preservation platform. *World Wide Web* 17, 4 (2014), 799–825. DOI: https://doi.org/10.1007/s11280-013-0234-4

[16] Martin Klein, Herbert Van de Sompel, Robert Sanderson, Harihar Shankar, Lyudmila Balakireva, Ke Zhou, and Richard Tobin. 2014. Scholarly Context Not Found: One in Five Articles Suffers from Reference Rot. *PLOS ONE* 9, 12 (12 2014), 1–39. DOI: https://doi.org/10.1371/journal.pone.0115253

[17] Angelina Kraft, Jan Potthoff, and Matthias Razum. 2016. Establishing a generic Research Data Repository. In *iPRES*.

[18] Angelina Kraft, Matthias Razum, Jan Potthoff, Andrea Porzel, Thomas Engel, Frank Lange, Karina van den Broek, and Filipe Furtado. 2016. The RADAR Project - A Service for Research Data Archival and Publication. *ISPRS Int. J. Geo-Information* 5, 3 (2016), 28. DOI: https://doi.org/10.3390/ijgi5030028

[19] Siân E. Lindley, Catherine C. Marshall, Richard Banks, Abigail Sellen, and Tim Regan. 2013. Rethinking the Web As a Personal Archive. In *WWW*.

[20] Catherine C. Marshall and Frank M. Shipman. 2012. On the Institutional Archiving of Social Media. In *JCDL*.

[21] Heather A. Piwowar, Roger S. Day, and Douglas B. Fridsma. 2007. Sharing Detailed Research Data Is Associated with Increased Citation Rate. *PLOS ONE* 2 (2007).

[22] Andrew Treloar. 2014. The Research Data Alliance: globally co-ordinated action against barriers to data publishing and sharing. *Learned Publishing* 27 (2014).

[23] Greg Wilson, D. A. Aruliah, C. Titus Brown, Neil P. Chue Hong, Matt Davis, Richard T. Guy, Steven H. D. Haddock, Kathryn D. Huff, Ian M. Mitchell, Mark D. Plumbley, Ben Waugh, Ethan P. White, and Paul Wilson. 2014. Best Practices for Scientific Computing. *PLoS Biology* 12 (2014).

[24] Ke Zhou, Claire Grover, Martin Klein, and Richard Tobin. 2015. No More 404s: Predicting Referenced Link Rot in Scholarly Articles for Pro-Active Archiving. In *JCDL*.