Towards a Dynamic Linked Data Observatory

Tobias Käfer
Karlsruhe Institute of Technology, Germany

Jürgen Umbrich
Digital Enterprise Research Institute, National University of Ireland, Galway, Ireland

Aidan Hogan
Digital Enterprise Research Institute, National University of Ireland, Galway, Ireland

Axel Polleres
Siemens AG Österreich, Siemensstrasse 90, 1210, Vienna, Austria

ABSTRACT

We describe work-in-progress on the design and methodology of the *Dynamic Linked Data Observatory*: a framework to monitor Linked Data over an extended period of time. The core goal of our work is to collect frequent, continuous snapshots of a subset of the Web of Data that is interesting for further study and experimentation, with an aim to capture raw data about the dynamics of Linked Data. The resulting corpora will be made openly and continuously available to the Linked Data research community. Herein, we (1) motivate the importance of such a corpus; (2) outline some of the use-cases and requirements for the resulting snapshots; (3) discuss different "views" of the Web of Data that affect how we define a sample to monitor; (4) detail how we select the scope of the monitoring experiment through sampling, (5) discuss the final design of the monitoring framework that will gather regular snapshots of (subsets of) the Web of Data over the coming months and years.

1. INTRODUCTION

Linked Data enjoys continued momentum in terms of publishing, research and development; as a result, the Web of Data continues to expand in size, scope and diversity. However, we see a niche in terms of understanding how the Web of Data evolves and changes over time. Establishing a more fine-grained understanding of the nature of *Linked Data dynamics* is of core importance to publishing, research and development. With regards to publishing, a better understanding of Linked Data dynamics would, for example, inform the design of tools for keeping published data consistent with changes in external data (e.g., [26]). With regards to research, more granular data on Linked Data dynamics would open up new paths for exploration, e.g., the design of hybrid/live query approaches [30] that know when a query relates to dynamic data, and that retrieves fresh results directly from the source. With regards development, for current systems, the results of such studies would inform crawling strategies, local index refresh rates and update strategies, cache invalidation techniques and tuning, and so forth.

Towards a better understanding of Linked Data dynamics, the community currently lacks a high-quality, broad, granular corpus of raw data that provides frequent snapshots of Linked Data documents over a sustained period of time. Current results in the area rely on domain-specific datasets (e.g., Popitsch and Haslhofer [26] focus on DBpedia changes [2]), or infrequently updated snapshots of open domain data over short monitoring time-spans (e.g., our previous work looked at 20 weekly snapshots collected in 2008 [28]). Thus, we believe that a first, important step is to build from scratch a new monitoring framework to derive a bespoke, continuously updated collection of snapshots. This collection can then be freely used to study not only the high-level dynamics of entities, but also to distil the fundamental underlying patterns of changes in the Web of Data across different domains (e.g., studying if certain graph patterns are more dynamic than others [29]).

Clearly the design of such a framework, and gathering the requirements for the resulting collection, is non-trivial: many factors and actors have to be taking into consideration in the context of the open Web and the Linked Data community. Conceptually, we want the collection to be:

***general-purpose*:** suitable to study for a wide range of interested parties;

***broad*:** capturing a wide selection of Linked Data domains;

***substantial*:** the number of documents monitored should allow for deriving confident statistical measures;

***granular & frequent*:** offering detailed data on sources;

***contiguous*:** allowing comparison of sources over time; and

***adaptive*:** able to discover the arrival of new sources, and can monitor more dynamic sources more frequently.

However, some of these targets are antagonistic and demand a trade-off. Monitoring a ***substantial*** number of sources in a ***granular & frequent*** fashion requires a practical compromise to be made. Similarly, ***contiguous*** data and ***adaptive*** monitoring are conflicting aims. Furthermore, the aims to be ***general-purpose*** and ***broad*** need more concrete consideration in terms of what sources are monitored.

For implementing the monitoring framework, other practical considerations include *politeness* such that remote servers are not unnecessarily overburdened, *stability* such that the monitoring experiment can function even in the case of hardware failure, and *resource overhead* such that the computation can be run on a single, dedicated commodity machine.

Taking these design criteria into account, herein we motivate and initially propose a framework for taking frequent, continuous snapshots of a subset of Linked Data which we call *DyLDO*: the *Dynamic Linked Data Observatory.* We currently focus on defining the size and scope of the monitoring experiment, discussing the rationale behind our choice

Acknowledgements. This work has been funded in part by Science Foundation Ireland under Grant No. SFI/08/CE/I1380 (Líon-2).

LDOW2012 April 16, 2012 Lyon, France

of sources to observe, touching upon various issues relating to sampling the Web of Data. Later, we also sketch the framework itself, outlining the crawling to be performed for each snapshot, providing rationale for different parameters used in the crawl, as well as proposing an adaptive filtering scheme for monitoring more dynamic sources more frequently. Our primary goal here is to inform the community of our intentions, outline our rationale, collect use-cases and potential consumers of the snapshot collection, and to gather feedback and requirements prior to starting the monitoring experiments. In particular, we currently focus on defining the scope of the monitoring experiment.

To begin, we motivate our work in terms of envisaged use-cases and research questions our corpus could help with (§ 2), subsequently presenting some related work (§ 3). Next, in order to understand what we are monitoring/sampling – to ascertain its borders – we ask the question WHAT IS THE WEB OF DATA?, and compare two prominent "views" thereof: (1) the Billion Triple Challenge dataset view, and (2) the CKAN/LOD cloud view (§ 4). Thereafter, we describe the sampling methodology we have used to derive a "seed list" of URIs that form the core of the monitoring experiment (§ 5). Finally, we outline the proposed monitoring framework, detailing setup parameters, and adaptive extensions (§ 6). We conclude with future directions and a call for feedback and potential use-cases from the community (§ 7).

2. USE CASES AND OPEN QUESTIONS

We now discuss the potential benefit and impact of our proposed observatory for Linked Data based on (1) some envisaged use-cases, and (2) some open research questions that our data could help to empirically investigate.

In previous work [31], we gave an overview of Web and Linked Data dynamics, presenting four community use cases that require technical solutions to deal with dynamic data. We first extend these four example scenarios to motivate our ongoing work on the Dynamic Linked Data Observatory.

UC-1: Synchronisation.

Synchronisation addresses the problem of keeping an offline sample/replica of the Web of Data up-to-date.

The most common scenarios is the maintenance of locally cached LOD indexes. To the best of our knowledge, none of the popular semantic web caches (such as hosted by Sindice and OpenLink) investigated index-update strategies based on the current knowledge of the dynamicity of Linked Data, but rather rely on publisher-reported information about update frequencies where available (e.g., sitemaps[1]).

UC-2: Smart Caching.

This use-case tries to find efficient methods to optimise systems operating live over Web data by minimising network traffic wasted on unnecessary HTTP lookups.

Versus synchronisation, this use-case targets systems that operate live and directly over the Web of Data. An exemplary use-case would be the integration of smart caching for live querying (e.g., [18]) or live browsing (e.g., [1]) over Linked Data, avoiding re-accessing a document or dereferencing a URI if it is unlikely to have changed according to knowledge of dynamicity patterns.

UC-3: Hybrid Architectures.

A large index of Linked Data can implement a hybrid architecture based on dynamicity statistics, where one processing pipeline is optimised for static knowledge, and another for dynamic knowledge.

In various Linked Data search and query engines, there is an inherent trade-off between running computation live during query-time or pre-computing answers offline. Abstractly, pre-computation suits static data and runtime computation suits dynamic data. Example trade-offs include dynamic insertions vs. batch loading, lightweight indexing vs. heavyweight indexing, runtime joins vs. live joins, backward-chaining reasoning vs. forward-chaining reasoning, window-based stream operators vs. global operators, etc. Knowledge of dynamicity can help decide which methods are appropriate for which data. Furthermore, smart hybrid queries may become possible: consider the query "GIVE ME (CURRENT) TEMPERATURES OF EUROPEAN CAPITALS", where knowledge of dynamicity would reveal that temperatures are dynamic and should be fetched live, whereas European capitals are static and can be run (efficiently) over the local index.

UC-4: External-Link Maintenance.

The link maintenance use-case addresses the challenge to preserve referential integrity and the correct type of links in the presence of dynamic external data.

Popitsch and Haslhofer [26] investigate this use case, which involves monitoring external Web datasets for changes to help ensure the integrity of links targeting them from local data. They propose DSNotify: a solution and an evaluation framework which is able to replay changes in a dataset; however, the authors only have knowledge of DBpedia dynamics to leverage. Such works help ensure the quality of links between datasets, and we hope that our corpora will help extend application to the broader Web of Data.

UC-5: Vocabulary Evolution and Versioning.

Knowledge of the dynamics of Linked Data vocabularies could lead to better versioning methods.

Changes in the semantics of terms in Linked Data vocabularies can have a dramatic influence on the interpretation of remote datasets using them. For example, the FOAF vocabulary is extremely widely used on the Web of Data [12, 20], but often resorts to informal versioning mechanisms: aside from the monotonic addition of terms, for example, FOAF recently removed disjointness constraints between popular classes like `foaf:Person` and `foaf:Document`, made `foaf:logo` inverse-functional, etc., that may change inferencing over (or even break) external data. Studying and understanding the dynamicity of vocabularies may motivate and suggest better methodologies for versioning.

We foresee that research and tools relating to these use-cases will benefit directly from having our data collection available for analysis. However, aside from these concrete use-cases and more towards a Web science viewpoint, we also see some rather more fundamental – and possibly related – empirical questions that our collection should help answer:

***Change frequency*:** Can we model change frequency of documents with mathematical models and thus predict future changes?

***Change patterns*:** Can we mine patterns that help to categories change behaviour?

[1] http://sitemaps.org/

***Degree of change*:** If a document changes, how much of its content is updated?

***Lifespan*:** What is the lifespan of Linked Data documents?

***Stability*:** How stable are Linked Data documents in terms of HTTP accessibility?

***Growth rate*:** How fast is the Web of Data evolving?

***Structural changes*:** Do we observe any changes in the structure of the network formed by links?

***Change triggers*:** Can we find graph patterns that trigger or propagate changes through the network?

***Domain-dependent changes*:** Do we observe a variation or clustering in dynamicity across different domains?

***Vocabulary-dependent changes*:** Do we observe different change patterns for data using certain vocabularies, classes or properties?

***Vocabulary changes*:** How do the semantics of vocabulary terms evolve over time?

3. BACKGROUND

Various papers have addressed similar research questions for the Web. Most works thus far have focused on the dynamicity of the traditional HTML Web, which (mostly) centres around dynamicity on the level of document changes. For the purposes of our use-cases, our notion of Linked Data dynamics goes deeper and looks to analyse dynamic patterns within the structured data itself: i.e., dynamicity should also be considered on the level of resources and of statements (as noted previously [29, 26]). Still, studies of the dynamicity of the HTML Web can yield interesting insights for our purposes. In fact, in previous works, we initially established a link between the frequency of change of Linked Data documents and HTML documents [28].

The study of the evolution of the Web (of Documents) and its implicit dynamics reaches back to the proposal of the first generation of autonomous World Wide Web spiders (aka. crawlers) around 1995. Bray [4] published one of the first studies about the characteristics of the Web and estimated its size in 1996. Around the same time, Web indexes such as AltaVista or Yahoo! began to offer one of the first concrete use-cases for understanding the change frequency of Web pages: the efficient maintenance of search engine indexes. In 1997, Coffman et al. [9] proposed a revisiting strategy for Web crawlers to improve the "freshness" of an index. This work was continuously improved over the subsequent years with additional experimental and theoretical results provided by Brewington [5, 6], Lim et al. [21], Cho et al. [8, 7], Fetterly et al. [14] and Ntoulas et al. [22], amongst others.

Based on large data collections, these papers presented theory and/or empirical analyses of the HTML Web that relate closely to the dynamicity questions we highlight. For example, various authors discovered that the ***change behaviour*** of Web pages corresponds closely with – and can be predicted using – a Poisson distribution [5, 6, 8, 7]; in previous work [28], we presented initial evidence that changes in Linked Data documents also follow a Poisson distribution, though our data was insufficient to be conclusive. Relating to high-level temporal ***change patterns***, Ntoulas et al. [22] analysed the different frequency of updates for individual weekdays and working hours. The same paper also empirically estimated the ***growth rate*** of the Web to be ∼8% new content every week, and regarding ***structural changes***, found that the link structure of the Web changes faster than the textual content by a factor of 3×. Various authors found that with respect to the ***degree of change***, the majority of changes in HTML documents are minor [21, 14, 22]. Loosely related to ***change triggers***, Fetterly et al. [14] found that certain parties simulate content changes to draw the attention of search engines. Regarding ***domain dependent changes***, various authors also showed that change frequencies vary widely across top-level domains [14, 8, 5, 6].

Relating to use-cases for studying the dynamics of Web documents, a variety have been introduced down through the years, including (i) the improvements of Web proxies or caches looked at by, e.g., Douglis et al [13], (ii) efficient handling of continuous queries over documents [24] or, returning to RDF, over SPARQL endpoints [25]. We refer interested readers to the excellent survey by Oita and Senellart [23], which provides a comprehensive overview of existing methodologies to detect Web page changes, and also surveys general studies about Web dynamics.

4. WHAT IS THE WEB OF DATA?

Our data collection should allow researchers to study various aspects and characteristics of data dynamics across a broad selection of Linked Data domains. However, it is not clear which data providers should be considered as "in scope", which are of interest for the Linked Data community who we target, and how we should define our target *population. Linked Data* itself is a set of principles and an associated methodology for publishing structured data on the Web in accordance with Semantic Web standards and Web Architecture tenets [19]. Various data providers are compliant with Linked Data principles to varying degrees: there's no one yardstick by which a dataset can be unambiguously labelled as "Linked Data".

For the purposes of the seminal *Linking Open Data* (LOD) project, Cyganiak and Jentszch use a variety of minimal requirements a dataset should meet to be included in the LOD Cloud diagram [10], which serves as a overview of connections between Linked Data corpora. However, the LOD Cloud is biased towards large monolithic datasets published on one domain, and does not cover low-volume cross-domain publishing as common for vocabularies such as FOAF, SIOC, etc. For example, platforms like `identi.ca/status.net`, Drupal, Wordpress, etc., can export compliant, decentralised Linked Data – using vocabularies such as FOAF and SIOC – from the various domains where they are deployed, but their exports are not in the LOD Cloud. Furthermore, it gives no explicit mention to the vocabularies themselves, which are of high relevance to our requirements.

A broader notion to consider is the *Web of Data*, which would cover these latter exporters and vocabularies, but which is somewhat ambiguous and with ill-defined borders. For our purposes, we define the Web of Data as being comprised of interlinked RDF data published on the Web.[2] No clear road-map is available for the Web of Data per this definition; the LOD cloud only covers prominent subsets thereof. Perhaps the clearest picture of the Web of Data comes from crawlers that harvest RDF from the Web. A prominent ex-

[2] This definition is perhaps more restrictive than some interpretations where, e.g., Sindice incorporates Microformats into their Web of Data index [11].

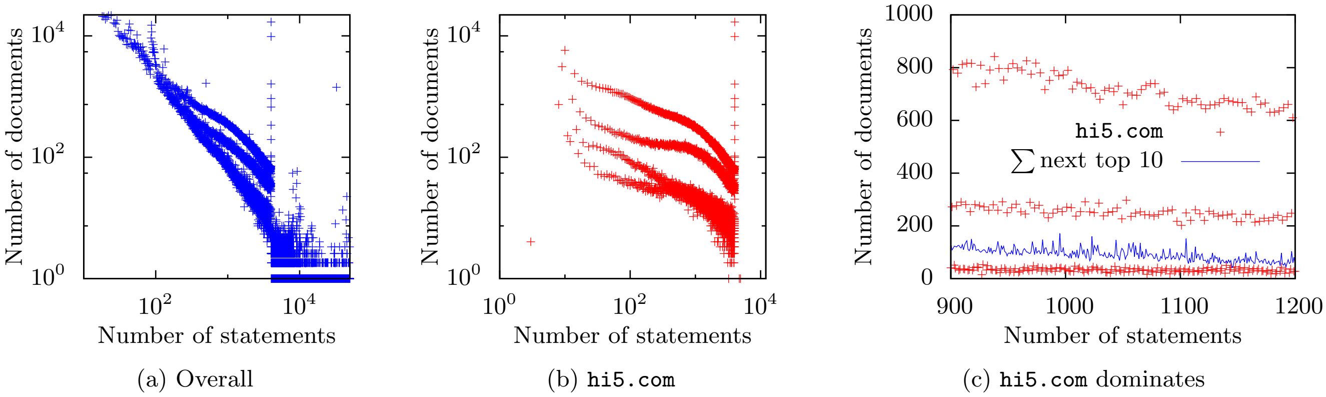

Figure 1: Distribution of the number of statements in documents for the BTC2011 dataset (1a) overall and (1b) for `hi5.com`; as well as (1c) the periodicity of distribution of statements-per-document for `hi5.com` that causes the split tail in (1a) & (1b).

ample is the Billion Triple Challenge Dataset, which is made available every year and comprises of data collected during a deep crawl of RDF/XML documents on the Web of Data. However, the precise composition of such datasets is unclear, and requires further study.

As such, the core question of what it is we want to monitor – i.e., what population of domains the Linked Data community is most interested in, and thus what population we should sample from – is non-trivial, and probably has no definitive answer. To get a better insight, in this section we contrast two such perspectives of the Web of Data, the:

Billion Triple Challenge 2011 dataset [**27**] which is collected from a Web crawl of over seven million RDF/XML documents; and the

Comprehensive Knowledge Archive Network (CKAN) [3] repository – specifically the `lodcloud` group therein – containing high-level metrics reported by Linked Data publishers, used in the creation of the LOD Cloud.

We want to see how these two road-maps of the Web of Data can be used as the basis of defining a population to sample.

4.1 The BTC 2011 dataset

The BTC dataset is crawled from the Web of Data using the MultiCrawler framework [17] for the annual Billion Triple Challenge [3] at the International Semantic Web Conference (ISWC). The dataset empirically captures a deep, broad sample of the Web of Data in situ.

However, the details of how the Billion Triple Challenge dataset is collected are somewhat opaque. The seed list is sampled from the previous year's dataset [27], where one of the initial seed-lists in past years was gathered from various semantic search engines. The crawl is for RDF/XML content, and follows URIs extracted from all triple positions. Scheduling (i.e., prioritising URIs to crawl) is random, where URIs are shuffled at the end of each round. As such, any RDF/XML document reachable through other RDF/XML documents from the seed list is within scope; otherwise, what content is (or is not) in the BTC – and how "representative" the dataset is of the Web of Data – is difficult to ascertain purely from the collection mechanisms.

As such, it is more pertinent to look at what the dataset actually contains. The most recent BTC dataset – BTC 2011 – was crawled in May/June 2011. The final dataset contains 2.145 billion quadruples, extracted from 7.411 million RDF/XML documents. The dataset contains RDF documents sourced from 791 *pay-level domains* (PLDs): a pay-level domain is a direct sub-domain of a top-level domain (TLD) or a second-level country domain (ccSLD), e.g., `dbpedia.org`, `bbc.co.uk`. We prefer the notion of a pay-level domain since fully qualified domain names (FQDNs) overexaggerate the diversity of the data: for example, sites such as `livejournal.com` assign different subdomains to individual users (e.g., `danbri.livejournal.com`), leading to millions of FQDNs on one site, all under the control of one publisher. The BTC 2011 dataset contained documents from 240,845 FQDNs, 233,553 of which were from the `livejournal.com` PLD. Henceforth, when we mention domain, we thus refer to a PLD (unless otherwise stated).

On the left-hand side of Table 1 we enumerate the top-25 PLDs in terms of quadruples contributed to the BTC 2011 dataset. Notably, a large chunk of the dataset (∼64%) is provided by the `hi5.com` domain: a social gaming site that exports a FOAF file for each user. As observed for similar corpora (cf. [20, Table A.1]) `hi5.com` has many documents, each with an average of over two thousand statements – an order of magnitude higher than most other domains – leading to it dominating the overall volume of BTC statements.

The dominance of `hi5.com` – and to a lesser extent similar sites like `livejournal.com` – shape the overall characteristics of the BTC 2011 dataset. To illustrate one prominent such example, Figure 1a gives the distribution of statements per document in the BTC dataset on log/log scale, where one can observe a rough power-law(-esque) characteristic. However, there is an evident three-way split in the tail emerging at about 120 statements, and ending in an outlier spike at around 4,000 statements. By isolating the distribution of statements-per-document for `hi5.com` in Figure 1b, we see that it contributes to the large discrepancies in that interval. The stripes are caused by periodic patterns in the data, due to its uniform creation: on the `hi5.com` domain, RDF documents with a statement count of $10 + 4f$ are heavily favoured, where ten triples form the base of a user's description and four triples are assigned to each of f

friends. Other lines are formed due to two optional fields (`foaf:surname`/`foaf:birthday`) in the user profile, giving a $9 + 4f$ and $8 + 4f$ periodicity line. An enforced ceiling of $f \leq 1,000$ friends explains the spike at (and around) 4,010.

The core message here is that although the BTC offers a broad view of the Web of Data, covering 791 domains, in absolute statement-count terms, the dataset is skewed by a few high-volume exporters of FOAF, and in particular `hi5.com`. When deriving global statistics and views from the BTC, the results say more about the code used to generate `hi5.com` profiles than the efforts of thousands of publishers.[4] This is also a naturally-occurring phenomenon in other corpora (e.g., [12, 20]) crawled from the Web of Data – not just isolated to the BTC dataset(s) – and is *not* easily fixed. One option to derive meaningful statistics about the Web of Data from such datasets is to apply (aggregated) statistics over individual domains, and never over the corpus as a whole.

4.2 CKAN/LOD cloud metadata

In contrast to the crawled view of the Web of Data, the CKAN repository indexes publisher-reported statistics about their dataset. These CKAN metadata are then used to decide eligibility for entry into the LOD cloud [10]: a highly prominent depiction of Linked Open Datasets and their interlinkage. A CKAN-reported dataset is listed in the LOD cloud iff it fulfils the following requirements: the dataset has to (1) be published according to core Linked Data principles, (2) contain at least one thousand statements and (3) provide at least 50 links to other LOD cloud datasets[5].

Given the shortcomings of the crawled perspective on the Web of Data, we explore these self-reported metadata to get an alternative view on what we should be sampling. On September 29, 2011, we downloaded the meta-information for the datasets listed in the `lodcloud` group on CKAN[6]. The data contain example URIs for the dataset and statistics such as the number of statements. We discovered meta-data for 297 datasets, spanning 206 FQDNs and 133 PLDs. On the right hand side of Table 1, we enumerate the top-25 largest reported datasets in the `lodcloud` group on CKAN. Note that where multiple datasets are defined on the same domain, the triple count is presented as the summation of said datasets. In this Table, we see a variety of domains claiming to host between 9.8 billion and 94 million triples.

Regarding the data formats present in the LOD cloud, most of the datasets claim to serve RDF/XML data (85 %), 4 % claim to serve RDFa (of which 50 % did not *also* offer RDF/XML). This shows the popularity of RDF/XML, but only supporting RDF/XML will still miss out on 15% of datasets. However, the syntax metadata are somewhat unreliable, where improper mime-types are often reported.

4.3 BTC vs. CKAN/LOD

Finally, we contrast the two different perspectives of the Web of Data. Between both, there are 854 PLDs mentioned, with BTC covering 791 domains (~92.6%), CKAN/LOD covering 133 domains (~15.6%), and the intersection of both covering 70 domains (~8.2% overall; ~8.8% of BTC; ~52.6% of CKAN/LOD). CKAN/LOD reports a total of 28.4 billion triples, whereas the BTC (an incomplete crawl) accounts for 2.1 billion quadruples (~7.4%). However, only 384.3 million quadruples in the BTC dataset (~17.9%) come from PLDs mentioned in the extracted CKAN/LOD metadata.

In Table 1, we present the BTC and CKAN/LOD statement counts side-by-side. We can observe that a large number of high-volume BTC domains are not mentioned on CKAN/LOD, where the datasets in question may not publish enough RDF data to be eligible by CKAN/LOD, or may not follow Linked Data principles or have enough external links, or may not have self-reported.

Perhaps more surprisingly however, we note major discrepancies in terms of the catchment of BTC statements versus CKAN/LOD metadata. Given that BTC can only *sample* larger domains, a lower statement count is to be expected in many cases: however, some of the largest CKAN/LOD domains do not appear at all. Reasons can be found through analysis of the BTC 2011's publicly available access log [27]. In Table 2, we present reasons for the top-10 highest-volume CKAN/LOD data providers not appearing in the BTC 2011 dataset (i.e., providers appearing with "—" on the right-hand side of Table 1). ROBOTS indicates that crawling was prohibited by `robots.txt` exclusions; HTTP-401 and HTTP-502 indicate the result of lookups for URIs on that domain; MIME indicates that the content on the domain did not return `application/rdf+xml` used as a heuristic in the BTC crawl to filter non-RDF/XML content; UNREACHABLE indicates that no lookups were attempted on URIs from that domain; OTHER refers solely to `europeana.eu`, which redirected all requests to their home page.

In summary, we see two quite divergent perspectives on the Web of Data, given by the BTC 2011 dataset and the CKAN/LOD metadata. Towards getting a better picture of what population we wish to sample for the monitoring experiments, and towards deciding on a sampling methodology, the pertinent question is then: WHICH PERSPECTIVE BEST SUITS THE NEEDS OF OUR MONITORING EXPERIMENT? As enumerated in Table 3, both perspectives have inherent strengths and weaknesses. As such, we believe that our sampling method should try to take the best of both perspectives, towards a hybrid view of the Web of Data.

5. SAMPLING METHODOLOGY

Due to the size of the Web and the need for frequent snapshots, sampling is necessary to create an appropriate collection of URIs that can be processed and monitored under the given time, hardware and bandwidth constraints. The goal of our sampling method is thus two-fold: to select a set of URIs that (i) capture a wide cross-section of domains and (ii) can be monitored in a reasonable time given our resources and in a polite fashion. Given the previous discussion, we wish to combine the BTC/crawled and CKAN/metadata perspectives when defining our seed-list.

Before we continue to describe the sampling methodology we choose, it is worthwhile to first remark upon sampling methods used in other dynamicity studies of the Web.

Published Sampling Techniques. There are several published works that present sampling techniques in order to

[4] Furthermore, `hi5.com` is not even a prominent domain on the Web of Data in terms of being linked, and was ranked 179/778 domains in a PageRank analysis of a similar corpus; http://aidanhogan.com/ldstudy/table21.html

[5] http://www.w3.org/wiki/TaskForces/CommunityProjects/LinkingOpenData/DataSets/CKANmetainformation

[6] http://thedatahub.org/group/lodcloud

№	Top-25 BTC			Top-25 CKAN/LOD		
	PLD	*BTC*	*LOD*	*PLD*	*LOD*	*BTC*
1	`hi5.com`	1,371,854,358	—	`rpi.edu`	9,803,140,000	900,464
2	`livejournal.com`	169,863,721	—	`linkedgeodata.org`	3,000,000,000	—
3	`tfri.gov.tw`	153,300,321	23,015,257	`legislation.gov.uk`	1,900,000,000	31,990,934
4	`scinets.org`	56,075,080	—	`wright.edu`	1,730,284,735	5
5	`ontologycentral.com`	55,124,003	122,000,000	`concordia.ca`	1,500,000,000	—
6	`rdfize.com`	36,154,381	—	`data.gov.uk`	1,336,594,576	13,302,277
7	`legislation.gov.uk`	31,990,934	1,900,000,000	`dbpedia.org`	1,204,000,000	25,776,027
8	`identi.ca`	30,429,795	—	`rdfabout.com`	1,017,648,918	—
9	`bibsonomy.org`	28,670,581	—	`dbtune.org`	888,089,845	1,634,891
10	`dbpedia.org`	25,776,027	1,204,000,000	`uniprot.org`	786,342,579	4,004,440
11	`freebase.com`	25,488,720	337,203,427	`unime.it`	586,000,000	—
12	`opera.com`	23,994,423	—	`uriburner.com`	486,089,121	—
13	`bio2rdf.org`	20,168,230	72,585,132	`openlibrary.org`	400,000,000	25,396
14	`archiplanet.org`	13,394,199	—	`sudoc.fr`	350,000,000	—
15	`data.gov.uk`	13,302,277	1,336,594,576	`freebase.com`	337,203,427	25,488,720
16	`loc.gov`	7,176,812	24,151,586	`fu-berlin.de`	247,527,498	5,658,444
17	`vu.nl`	6,106,366	14,948,788	`dataincubator.org`	205,880,247	3,695,950
18	`bbc.co.uk`	5,984,102	80,023,861	`viaf.org`	200,000,000	—
19	`rambler.ru`	5,773,293	—	`europeana.eu`	185,000,000	—
20	`fu-berlin.de`	5,658,444	247,527,498	`moreways.net`	160,000,000	—
21	`uniprot.org`	4,004,440	786,342,579	`rkbexplorer.com`	134,543,526	220
22	`dataincubator.org`	3,695,950	205,880,247	`ontologycentral.com`	122,000,000	55,124,003
23	`zitgist.com`	3,446,077	60,000,000	`opencorporates.com`	100,000,000	—
24	`daml.org`	3,135,225	—	`uberblic.org`	100,000,000	—
25	`mybloglog.com`	2,952,925	—	`geonames.org`	93,896,732	458,490

Table 1: Statement counts for top-25 PLDs in the BTC with corresponding reported triple count in CKAN (left), and top-25 PLDs in CKAN with BTC quad count (right)

PLD	Robots	Http-401	Http-502	Mime	Unreachable	Other
`linkedgeodata.org`			✗	✗		
`concordia.ca`				✗		
`rdfabout.com`				✗		
`unime.it`					✗	
`uriburner.com`	✗					
`sudoc.fr`					✗	
`viaf.org`				✗		
`europeana.eu`						✗
`moreways.net`					✗	
`uberblic.org`		✗				

Table 2: Reasons for largest ten PLDs in CKAN/LOD not appearing in BTC 2011

		BTC
Pros:	✓	covers more domains (791)
	✓	empirically validated
	✓	includes vocabularies
	✓	includes decentralised datasets
Cons:	✗	influence of high-volume domains
	✗	misses 47.4% of LOD/CKAN domains
		LOD/CKAN
Pros:	✓	domains pass “quality control”
	✓	community validated
Cons:	✗	covers fewer domains (133)
	✗	self-reported statistics
	✗	misses vocabularies
	✗	misses decentralised datasets

Table 3: Advantages and disadvantages for both perspectives of the Web of Data

create a corpus of Web documents that can be monitored over time. Having studied a variety of major works, we could not find common agreement on a suitable sampling technique for such purposes. The targeted use-cases and research questions directly affect the domain and number of URIs, as well as the monitoring frequency and time frame.

Grimes and O'Brien [16] studied the dynamics of very frequently changing pages and prepared their seed list accordingly: initially starting from a list of URIs provided by a Google crawl, they performed two crawls a day and selected the most dynamic URIs (in terms of content changes) that could also be successfully accessed 500 times in a row. As such, the authors focus on monitoring stable and dynamic Web documents. Fetterly et al. [14] randomly sampled URIs from a crawl seeded from the Yahoo! homepage to cover many different topics and providers, giving broad coverage for a general-purpose study; however, the surveyed documents would be sensitive to the underlying distribution of documents in the original Yahoo! corpus. Cho and Garcia-Molina [8] studied the change frequency of pages using a dataset which was sampled by selecting the root URIs of the 270 most popular domains from a 25 million web page crawl and then crawling three thousand pages for each of the domains; this method provides a good, broad balance of documents across different domains.

Our sampling method. We can conclude that existing sampling methods select URIs from crawled documents, either randomly, because of specific characteristics (e.g., dynamic or highly ranked), or to ensure an even spread across different hosts. Thus, we decide to use a combination of these three methods to generate our list of URIs.

Given the previous discussion of Section 4, we start with an initial seed list of URIs taken from: (1) the registered example URIs for the datasets in the LOD cloud and (2) the most popular URIs in the BTC dataset of 2011. The most popular URIs are selected based on a PageRank analysis of the documents in the BTC 2011 dataset, where we select the top-k ranked documents from this analysis (please see [15] for details); note that many of the top ranked documents refer to commonly instantiated vocabularies on the Web of Data, which are amongst the most linked/central Linked Data documents. At the time of access, we found 220 example URIs in the CKAN/LOD registry, and we complement them with the top-220 document URIs from the BTC 2011 to generate a list of 440 core URIs for monitoring. The core URIs contain 137 PLDs, 120 from the CKAN/LOD examples and 37 from the most popular BTC URIs. This selection guarantees us to cover all relevant domains (similar to [14]) and to also consider the most popular and interlinked URIs on the Web of Data (similar to [8]).

Obviously, 440 seed URIs are insufficient to resolve a meaningful corpus for observation over time. Thus, we decide to use a crawl and expand outwards from these core URIs to find other documents to monitor in their vicinity. Importantly, we wish to stay in the close locale of the 440 core URIs; if we go further, we will encounter the same problems as observed for the BTC 2011 dataset, where the data are skewed by a few high-volume exporters. To avoid being diluted by, e.g., `hi5.com` data and the likes, we thus stay within a 2-hop crawl radius from the core URIs. From the data thus extracted, we then sample a final set of extended seed URIs to monitor. The result is then our best-effort compromise to achieve representative snapshots of Linked Data that (i) take into account both views on Linked Data by including CKAN and BTC URIs in the core seed list, (ii) extend beyond the core seed list in a defined manner (2 hops), and (iii) do not exceed our crawling resources.

Crawling setup. The crawling setup will have a significant effect on the selection of URIs to monitor, and so we provide some detail thereupon. Our implementation is based on two open source Java projects: (1) LDSpider[7], a multi-threaded crawling framework for Linked Data, and (2) any23[8], a parsing framework for various RDF syntaxes. The experiments are intended to run on a dedicated, single-core 2.2GHz Opteron x86-64, 4GB RAM on a university network. Thereafter, we use the following configuration:

All RDF syntaxes: unlike BTC crawls, which only consider RDF/XML, we wish to consider all standard serialisation formats for RDF (including Turtle, which is soon to be standardised), as supported by any23. Further, we do not pre-filter content based on `Content-type` reporting or file extensions. RDFa is becoming a preferred format for many publishers: when parsing RDFa, we monitor the output statements and exclude the content of HTML documents for which we find *only* "accidental" triples as extracted from titles, stylesheets, icons, etc., and instead only consider documents that *intend* to publish non-trivial RDFa.

Threads: multithreading keeps the hardware busy while slow HTTP requests are being processed; in previous work [20], we found 64 threads to offer the best tradeoff between parallelism and CPU/disk contention.

Timeouts: we terminate unresponsive connections and sockets after 120 seconds. Timeouts are kept deliberately high to help ensure stable crawls.

Links: we consider all URIs contained in the RDF data as potential links to follow (and, e.g., not only the value of `rdfs:seeAlso` or such).

Breadth-first: we crawl documents in a round-based, URI scheduling approach, which should result in a broader set of diverse data (assuming a diverse seed-list) [20].

Redirects: 301, 302, 303 and 307 HTTP response codes are not treated as links, but are instead followed directly in the same round until we reach a non-direct response (or hit a cycle/path-limit).

Per-domain Queue: our crawler queue is divided into individual per-domain queues, which are polled round-robin: this helps ensure a maximal delay between accessing the same domain again.

Priority Queue: within each individual per-domain queue, we prioritise URIs for which we have found the most links thus far. (This only affects the crawl if an incomplete round is performed.)

Politeness Policy: we implement a politeness delay of two seconds, meaning that we do not access the same PLD twice within that interval; further, for each domain, we retrieve and respect standard `robots.txt` exclusions.

The minimum amount of time taken to complete a round becomes the maximum number of active URIs for a single domain, multiplied by the politeness delay. The combination of this per-PLD delay and the distribution of documents per domain on the Web [20] introduces the problem of *PLD starvation* [20]: the nature of the Web of Data means that after numerous low-volume domains have been crawled, the few remaining domains are not enough to keep the crawler resources busy between the politeness delay. In the worst case, when one PLD is left in the queue, only one URI can be crawled every two seconds. However, the *frontier* – the list of URIs for the next round – may contain a diverse set of domains that can overcome this problem, and allow for higher crawling throughput. Hence, we also add a termination condition for each round: once (1) the seed URIs have been processed, (2) all active redirects have been processed *and* (3) the number of active PLDs remaining in the per-domain queue drops under the number of threads (in our

[7] http://ldspider.googlecode.com/
[8] http://any23.org/

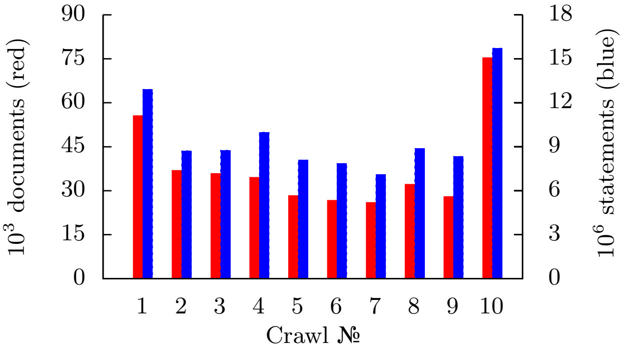

Figure 2: Number of stmts. and docs. per crawl experiment

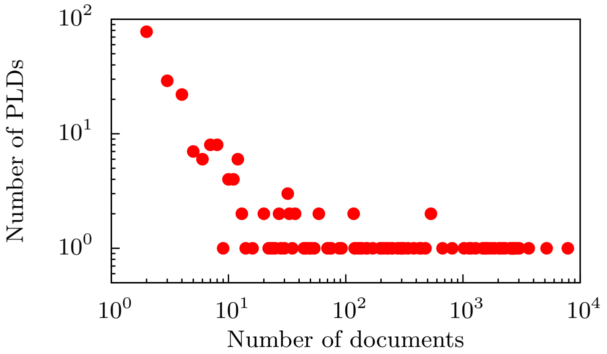

Figure 4: Distribution of the number of docs. per PLD

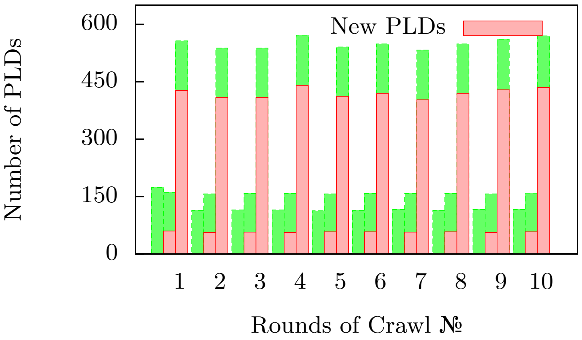

Figure 3: Number of PLDs per round per crawl experiment

№	PLD	URIs
1	gesis.org	7,850
2	chem2bio2rdf.org	5,180
3	dbpedia.org	3,643
4	freebase.com	3,026
5	fer.hr	2,902
6	loc.gov	2,784
7	concordia.ca	2,784
8	dbtune.org	2,767
9	fu-berlin.de	2,689
10	semantictweet.com	2,681

Table 4: Top 10 PLDs based on the number of URIs

setup 64), we end the current round and move to the next round (shifting remaining URIs to the frontier).

Seed list. Starting from our list of 440 core URIs, we wish to expand a 2-hop crawl using the outlined framework, from which we will extract the final seed list of URIs to monitor in our observation framework. However, due to the unpredictability/non-determinism of remote connections, we want to ensure a maximal coverage of the documents in this neighbourhood. Along these lines, we repeated ten complete 2-hop crawls from our core URI list.

With respect to the non-determinism, Figure 2 shows for each round the number of documents (left bars on y-axis) and the number of statements (right bars on y-axis). We can observe that two crawls (crawl number 1 and 10) have a significantly higher number of statements compared to the other crawls. One reason for this large discrepancy relates to the identi.ca domain, where a URI (referring to Tim Berners-Lee's account; a highly ranked document in the BTC dataset) in the seed round of crawls 1 and 10 offered a rich set of links within that domain, whereas the lookup failed in the other crawls, cutting off the crawler's path in that domain: for example, in the first crawl, identi.ca accounted for 1.5 million statements, whereas in crawl 2, the domain accounted for 17 thousand statements. Such examples illustrate the inherent non-determinism of crawling.

In Figure 3, we show for each crawl the number of visited PLDs per round together with the number of new PLDs per round with respect to the previous round. The left bar for each crawl represents Round 0, the middle bar Round 1, and the right bar Round 2. We can observe that the relative level of domains across the crawls is much more stable when compared with the number of documents (cf. Figure 2). Across rounds, the graph shows an average ∼1.3× increase of active PLDs between Rounds 0–1, and an increase of ∼3.4× between Rounds 1–2. Further, we observe that ∼30% of the PLDs in Round 1 are new compared to the previous round and roughly 70% of the PLDs in Round 2 are not visited by the crawler in the rounds before.

Given the non-deterministic coverage of documents, to ensure comprehensive coverage of URIs in the 2-hop neighbourhood, we take the union of URIs that dereferenced to RDF content, resulting in a total set of 95,737 URIs spanning 652 domains, giving an average of 146.8 dereferenceable URIs per domain. Figure 4 shows in log/log scale the distribution of the number of PLDs (y-axis) against the number of URIs in the union list (x-axis); we see that 379 PLDs (∼58.1%) have one URI in the list, 78 PLDs (∼12.0%) have two URIs, and so forth. In addition, Table 4 lists the number of URIs for the top-10 PLDs in the set (represented by the ten rightmost dots in Figure 4).

6. MONITORING CONFIGURATION

The next step in the setup of our observatory is to select the *monitoring techniques* and intervals we apply. Note that we have yet to start the monitoring experiments, where we now instead present some initial results and outline our proposed methodology for feedback from the community.

In general, there exists two fundamental monitoring techniques. The first technique is to periodically download the content of a fixed list of URIs, as widely used in the literature [5, 6, 7, 8]; this technique allows to study the evolution of sources over time in a ***contiguous*** fashion. The second technique is to periodically perform a crawl from a defined set of URIs [22]; this technique is more suitable if one wants to study the evolution of the neighbourhood network of the seed URIs in an ***adaptive*** fashion, but also can introduce a factor of randomness based on the crawling methods.

We decided to again apply a hybrid approach: primarily, we do not want to limit our observations to URIs online at the start of the experiment, although they will still be our focus. We thus take the set of 95,737 sampling URIs extracted in the previous section as a *kernel* of ***contiguous*** URIs accessed consistently in each snapshot. From the kernel, we propose to crawl as many URIs again using the crawler configuration outlined in the previous section, forming the ***adaptive*** segment of the snapshot. Roughly half of our snapshot would comprise of the contiguous kernel, reliably providing data about said URIs; and the other half of our snapshot would comprise of the adaptive crawl, reflecting changes in the neighbourhood of the kernel. We do not limit PLDs in the adaptive crawl so as to not exclude data providers that come online during the course of the experiment. This setup allows for studying (i) dynamics within the datasets (ii) dynamics between datasets (esp. links) (iii) and the growth of Linked Data and the arrival of new sources (although to a lesser extent).

Next, we must decide on the *monitoring intervals* for our platform: how frequently we wish to perform our crawl. In the literature, it is common to take the data snapshots in either a daily [22, 14] or weekly [5, 6, 8, 7] fashion. Again, in a practical sense, the intervals are highly dependent on the available resources, and the size of the seed list. Given our resources and the monitoring requirements, we decided to perform a full snapshot every week.

In addition, to get more granular data in a temporal sense, we propose to apply an adaptive scheduling policy that takes more frequent snapshots for more dynamic data. As such, we propose to set up different monitoring intervals within the full weekly snapshots, where we increase the monitoring interval for a kernel document if it changes within two consequential snapshots of the previous interval. Figure 5 depicts the core idea. Using this adaptive approach, we can avoid the local and remote computational overhead involved in regularly polling documents that are observed to be very static. At the moment, we consider fixing the maximum number of intervals per week to 16, which resolves to a time interval width of ∼10 hours. However, the intervals will take a minimum of a week to "warm-up", and will probably take longer to stabilise; thus, we can manually add further intervals on-the-fly at a later stage once the experiments are underway (if deemed useful from the results).

Initial experiment. We conducted an initial experiment, performing a crawl for the 95,737 URI kernel. Our framework downloaded 16 million statements from 80,000 documents, taking a total of 6 hours and 40 minutes. Figure 6 shows the number of documents processed per crawl hour. The average download rate increases from 20k to 70K documents per hour after 60 minutes, but tails off severely after hour 3 due to PLD starvation. We see that processing the kernel is thus feasible within a 10 hour interval.

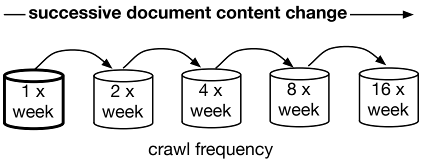

Figure 5: Adaptive scheduling based on change events.

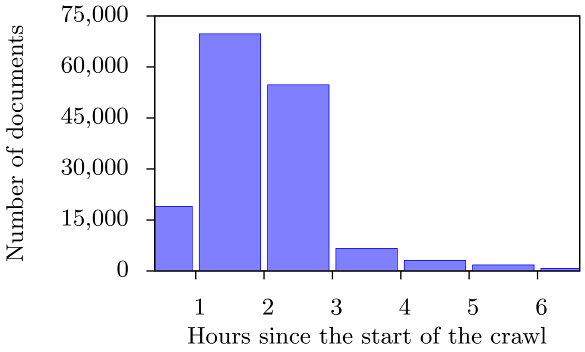

Figure 6: Number of downloaded documents over time.

7. CONCLUSION & OUTLOOK

In this paper, we have presented ongoing work towards building DyLDO: the Dynamic Linked Data Observatory. This observatory aims to continuously gather a high-quality collection of Linked Data snapshots for the community to gain a better insight into the underlying principles of dynamicity on the Web of Data. We motivated our proposals based on several concrete use-cases and research questions that could be tackled using such a collection, and presented related works that treat various aspects of dynamicity for HTML documents on the traditional Web. Next we looked at the non-trivial question of what view we should adopt for the Web of Data, introducing and comparing the BTC and CKAN/LOD perspectives, showing how and where they diverge, and weighing up their respective pros and cons. We proposed selecting a kernel of sources to monitor around a core set of URIs taken from BTC and CKAN/LOD datasets, which are then extended by a 2-hop crawl. We also presented the detailed crawl configuration we planned to use for our monitoring experiments. Finally, we proposed our methodology for performing the continuous observation of sources in and around the kernel, as well as using adaptive intervals to monitor more dynamic sources more frequently.

We plan to begin the monitoring experiments in the next few weeks, and to run them continuously and indefinitely. We still have some practical issues to tackle in terms of creating a backup and archiving system, a site offering access to the community[9], as well as remote fallback procedures in the event of network or hardware failure. Thus, we are at a crucial stage, and are keen to gather final feedback and requirements from the community: we are particularly anxious to hear from people who would have a specific interest or use-case for such data – be it in terms of research analysis, evaluation frameworks, etc. – what requirements

[9] Planned for http://swse.deri.org/DyLDO/

they have, and whether or not current proposals would be sufficient. Significant changes will invalidate the snapshots collected up to that point, so we want to gather comments and finalise and activate the framework as soon as possible. In the near future, the community can then begin to chart a new dimension for the Web of Data: time.

8. References

[1] T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets. "Tabulator: Exploring and Analyzing linked data on the Semantic Web". In: *SWUI*. 2006.

[2] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. "DBpedia - A crystallization point for the Web of Data". In: *J. Web Sem.* 7.3 (2009), pp. 154–165.

[3] C. Bizer and D. Maynard. "The Semantic Web Challenge, 2010". In: *J. Web Sem.* 9.3 (2011), p. 315.

[4] T. Bray. "Measuring the Web". In: *Comput. Netw. ISDN Syst.* 28 (7-11 1996), pp. 993–1005.

[5] B. Brewington and G. Cybenko. "How dynamic is the Web?" In: *Computer Networks* (2000).

[6] B. Brewington and G. Cybenko. "Keeping up with the changing web". In: *Computer* 33.5 (2000), pp. 52–58.

[7] J. Cho and H. Garcia-Molina. "Effective page refresh policies for Web crawlers". In: *ACM Transactions on Database Systems* 28.4 (Dec. 2003), pp. 390–426.

[8] J. Cho and H. Garcia-Molina. "Estimating frequency of change". In: *ACM Transactions on Internet Technology* 3.3 (Aug. 2003), pp. 256–290.

[9] E. G. Coffman Jr., Z. Liu, and R. R. Weber. "Optimal robot scheduling for web search engines". In: *Journal of scheduling* 1 (1997), pp. 0–21.

[10] R. Cyganiak and A. Jentzsch. *The Linking Open Data cloud diagram.* URL: http://richard.cyganiak.de/2007/10/lod/ (visited on 02/06/2012).

[11] R. Delbru, N. Toupikov, M. Catasta, and G. Tummarello. "A Node Indexing Scheme for Web Entity Retrieval". In: *ESWC (2)*. 2010, pp. 240–256.

[12] L. Ding and T. Finin. "Characterizing the Semantic Web on the Web". In: *ISWC*. 2006, pp. 242–257.

[13] F. Douglis, A. Feldmann, B. Krishnamurthy, and J. Mogul. "Rate of Change and other Metrics: a Live Study of the World Wide Web". In: *USENIX Symposium on Internetworking Technologies and Systems* December (1997).

[14] D. Fetterly, M. Manasse, M. Najork, and J. L. Wiener. "A large-scale study of the evolution of web pages". In: *WWW*. 2003, pp. 669–678.

[15] B. Glimm, A. Hogan, M. Krötzsch, and A. Polleres. *OWL: Yet to arrive on the Web of Data?* CoRR. URL: http://arxiv.org/pdf/1202.0984.pdf (visited on 02/06/2012).

[16] C. Grimes and S. O'Brien. "Microscale evolution of web pages". In: *WWW*. 2008, pp. 1149–1150.

[17] A. Harth, J. Umbrich, and S. Decker. "MultiCrawler: A Pipelined Architecture for Crawling and Indexing Semantic Web Data". In: *International Semantic Web Conference*. 2006, pp. 258–271.

[18] O. Hartig, C. Bizer, and J. C. Freytag. "Executing SPARQL Queries over the Web of Linked Data". In: *ISWC*. 2009, pp. 293–309.

[19] T. Heath and C. Bizer. *Linked Data: Evolving the Web into a Global Data Space*. Vol. 1. Morgan & Claypool, 2011, pp. 1–136.

[20] A. Hogan, A. Harth, J. Umbrich, S. Kinsella, A. Polleres, and S. Decker. "Searching and browsing Linked Data with SWSE: The Semantic Web Search Engine". In: *J. Web Sem.* 9.4 (2011), pp. 365–401.

[21] L. Lim, M. Wang, S. Padmanabhan, J. Vitter, and R. Agarwal. "Characterizing web document change". In: *Advances in Web-Age Information Management* (2001), pp. 133–144.

[22] A. Ntoulas, J. Cho, and C. Olston. "What's new on the web? The evolution of the web from a search engine perspective". In: *WWW*. 2004, pp. 1–12.

[23] M. Oita and P. Senellart. *Deriving Dynamics of Web Pages: A Survey*. INRIA TR.: inria-00588715. 2011.

[24] S. Pandey, K. Ramamritham, and S. Chakrabarti. "Monitoring the dynamic web to respond to continuous queries". In: *WWW*. 2003, pp. 659–668.

[25] A. Passant and P. Mendes. "sparqlPuSH: Proactive notification of data updates in RDF stores using PubSubHubbub". In: *Scripting and Development Workshop at ESWC*. 2010.

[26] N. Popitsch and B. Haslhofer. "DSNotify - A solution for event detection and link maintenance in dynamic datasets". In: *J. Web Sem.* 9.3 (2011), pp. 266–283.

[27] *The Billion Triple Challenge*. URL: http://challenge.semanticweb.org/ (visited on 02/06/2012).

[28] J. Umbrich, M. Hausenblas, A. Hogan, A. Polleres, and S. Decker. "Towards Dataset Dynamics: Change Frequency of Linked Open Data Sources". In: *Proc. of LDOW at WWW*. 2010.

[29] J. Umbrich, M. Karnstedt, and S. Land. "Towards Understanding the Changing Web: Mining the Dynamics of Linked-Data Sources and Entities". In: *Proc. of KDML at LWA*. 2010.

[30] J. Umbrich, M. Karnstedt, J. X. Parreira, A. Polleres, and M. Hauswirth. "Linked Data and Live Querying for Enabling Support Platforms for Web Dataspaces". In: *DESWEB at ICDE*. 2012.

[31] J. Umbrich, B. Villazon-Terrazas, and M. Hausenblas. "Dataset Dynamics Compendium: Where we are so far!" In: *Proc. of COLD at ISWC*. Shanghai, 2010.