Towards Dataset Dynamics: Change Frequency of Linked Open Data Sources

Jürgen Umbrich, Michael Hausenblas, Aidan Hogan, Axel Polleres, Stefan Decker
Digital Enterprise Research Institute (DERI)
National University of Ireland, Galway
IDA Business Park, Lower Dangan, Ireland

firstname.lastname@deri.org

ABSTRACT

Datasets in the LOD cloud are far from being static in their nature and how they are exposed. As resources are added and new links are set, applications consuming the data should be able to deal with these changes. In this paper we investigate how LOD datasets change and what sensible measures there are to accommodate dataset dynamics. We compare our findings with traditional, document-centric studies concerning the "freshness" of the document collections and propose metrics for LOD datasets.

1. INTRODUCTION

The Linked Open Data (LOD) movement has gained remarkable momentum over the past years. At the time of writing, well over one hundred datasets – including UK governmental data, the New York Times dataset, and LinkedGeoData – have been published, providing several billion RDF triples interlinked by hundreds of millions of RDF links. Some datasets, such as DBpedia, have been available from the very beginning of the LOD movement and regularly undergo changes on both the instance level and the schema level. New resources are added and old resources are removed; new links are set to other datasets, and old links are removed as the target has vanished. We should hence assume that datasets in the LOD cloud are dynamic in their very nature. *Dataset dynamics* is a term we recently coined [1], essentially addressing *content and interlinking changes in Linked Data sources.*

Our main contributions herein are: (i) define dataset dynamics characteristics and how to measure them, and (ii) compare the dataset dynamics of the LOD cloud to the traditional Web (Web of HTML Documents). The motivating use-case for our study of dataset dynamics is to gain insights into – and hopefully improve – concurrent work on an efficient system for performing live queries over the Linked Open Data Web [13]. However, aside from this use-case having knowledge about dataset dynamics is essential for a number of tasks:

- web crawling and caching [9];
- distributed query optimisation [13];
- maintaining link integrity [16];
- servicing of continuous queries [22];
- replication and synchronisation [24].

LDOW2010, April 27, 2010, Raleigh, USA.
.

We begin in Section 2 by reviewing existing work, and continue in Section 3 by discussing and contrasting document vs. entity centric perspectives concerning dynamics. Thereafter, in Section 4 we present the background of our analysis, in Section 5 we describe our methodology for analysing dataset dynamics, and in Section 6 we discuss the results of our analysis. Finally, in Section 7, we conclude and render future work.

2. RELATED WORK

As motivated above, the study of changes in documents and data sets is very relevant for a broad range of application domains. Earlier work discussed analysis of the dynamics of the Web circa. 2008, leveraging their findings for optimisation of re-indexing techniques [6]. The work of Cho et. al. provides a comprehensive study regarding the change frequency of Web documents: earlier work focussed on how to integrate the knowledge for an incremental crawler [8]; further work provided a detailed discussion for estimators of the frequency of changes given incomplete history [9]. Other research has focused on, for example, investigating the dynamics of Wikipedia articles [3] and the evolution of database schema over time [21].

With respect to the Semantic Web, some research regarding dynamics has been conducted with respect to analysing the evolution of ontologies in the life science community [15]. In [16] the authors reported on their work concerning *DSNotify*, a system for detecting and fixing broken links in LOD datasets.

However – and to the best of our knowledge – we are not aware of any published studies more generally regarding the change frequency of resources on the Linked Open Data Web, and thus deem the work herein to be novel.

3. DOCUMENTS VS. ENTITIES: DIFFERENT PERSPECTIVES ON LINKED DATA

There are various aspects of dataset dynamics which must be considered in order to achieve a comprehensive overview of how Linked Open Data changes and evolves on the Web. Firstly, the *change frequency* of data on the Web can vary significantly across datasets, from rather static sources – such as archives – to high-frequently changing sources – for example in the micro-blogging domain. Also, the *change volume* can range from small-scale updates – in our case, updates involving a low number of triples – to bulk updates, which potentially affect many resources. One must also pay attention to the *perspective* one takes on resources: that is, whether we are interested in local changes of particular datasets, or are interested in global changes with respect to what is said about a URI in all accessible linked datasets.

Before we continue, however, we must first provide some preliminaries. Firstly, our notion of a 'document' refers to an atomic Web 'container' in which Linked Data is typically exposed: these include RDF/XML, (X)HTML+RDFa documents, etc. Secondly, we often refer to an 'entity' by which we intuitively mean anything identified by a URI in Linked Data, including classes, properties, and the "real-world artefacts" described.[1] Following from both, we can now distinguish the following perspectives in dataset dynamics:

1. A *document-centric* perspective, which focuses on datasets and is motivated by the "traditional" Web as well as the REST community [12, 2]

2. An *entity-centric* perspective, which focuses on entities as described in the Linked Open Data Web [5] – we further separate the entity-centric perspective into:

    (a) An *entity-per-document* perspective which takes into account occurrences of an entity with respect to a specific document

    (b) A *global entity* perspective which takes into account all appearances of an entity across the Web

In particular, the entity-centric perspectives are more LOD-specific than the document-centric perspective prevalent in more traditional views on dataset dynamics. Many applications operating on the LOD cloud assume an entity-centric view where entities become the unit of knowledge and data on such entities are aggregated from multiple documents. Also, LOD documents may be dynamically served by an entity-centric index (e.g., a SPARQL endpoint), whereby a change in one entity may entail changes in many documents. Thus, we believe the distinction between the document- and entity-centric perspectives to be important for our purposes herein.

In fact, the global entity perspective may be infeasible to monitor as arbitrary new sources can publish data about any entities. For this reason – and despite formally discussing 2b herein – note that in the present work we will focus on the analysis of 1 and 2a, and leave approximative techniques for analysis of 2b as part of our future research (discussed in Section 7).

Despite the two distinct perspectives, both are somehow related: there is naturally a relation between entities and their appearances in different containers. Along these lines, Figure 1 depicts a typical distribution of entities per document in the LOD cloud. As we have already shown elsewhere [17], this distribution follows a power law.

In order to formalise what we mean by these different perspectives, let $\mathcal{R} = \{r_1, ..., r_n\}$ be the set of all *resources* as of the Architecture of the World Wide Web [19]: that is, HTTP entities and documents. Further, we define $\mathcal{D} = \{d_1, ..., d_n\}, \mathcal{D} \subset \mathcal{R}$ as the set of all documents (i.e., dereferenceable entities that point to RDF data) and $\mathcal{E} = \{e_1, ..., e_n\}$, $\mathcal{E} \subset \mathcal{R}$ as the set of all entities. A document d_i can mention various entities; thus, we denote the set of entities mentioned in document d as $E(d) \subseteq \mathcal{E}$ and likewise the set of all documents mentioning e as $D(e) \subseteq D$. Further, let $ver(d,t)$ be the state of document d at time-point t – i.e., the RDF graph served by d at time t. It is clear, that different use cases require specific state functions $ver(d,t)$ and equality measures; e.g. a state function could be the hash value of the RDF graph, a set of RDF statements or the set of inferable new statements.

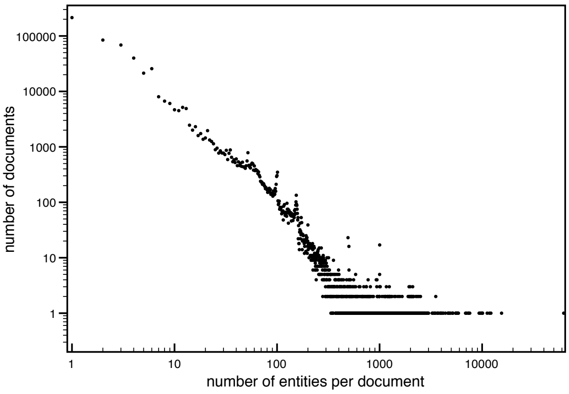

Figure 1: A typical distribution of entities in documents.

Then, the *document change function* of document d from time t to t' (where $t < t'$) is defined as follows:

Definition 1. ***Document Change Function***

$$C_d(t,t') = \begin{cases} 0 & \textit{if } ver(d,t) = ver(d,t') \\ 1 & \textit{otherwise} \end{cases}$$

Likewise, we define the *entity-per-document change function* as follows:

Definition 2. ***Entity-per-document Change Function***

$$C_d^e(t,t') = \begin{cases} 0 & \textit{if } ver(d,t) \cap e = ver(d,t') \cap e \\ 1 & \textit{otherwise} \end{cases}$$

where by $G \cap u$ we denote all triples in graph G mentioning u. Finally, the entity change function can be defined as follows:

Definition 3. ***Entity Change Function***

$$C^e(t,t') = max_{\forall d \in D(e)}(C_d(t,t'))$$

Please note that we pursue a purely 'syntactic' notion of change, and do not consider more advanced notions relating to 'semantic' change: for example, we would consider a change in a datatype literal if the syntax of that literal changes even though the semantic interpretation does not – this change would then propagate to the respective entity/document despite no real change on the semantic level. Further, we do not consider any forms of reasoning in the changes – e.g., we do not propagate changes in a class definition as changes to it's member entities. We leave further discussion and related analysis of 'semantic vs. syntactic change' for future work.

It may also be interesting to consider more closely the relationship between documents and the entities they contain, examining separately the change function of entities which

[1] Note that in this paper, we currently overlook entities 'identified' by blank-nodes; concretely, blank-node entities do not have consistent naming which has adverse consequences on the analysis presented in Section 5.3

are considered ‘local’ with respect to the document they appear in. To this end, we introduce the term *local entity*, meaning an entity in a document whose pay-level domain (PLD) is the same as the document’s PLD: here, a PLD is defined as *any domain that requires payment at a [top-level-domain] (TLD) or country-code TLD registrar* [20]. Taking an example, let $PLD(uri)$ be the PLD extraction function; then:

$$PLD(http://www.deri.ie/) = deri.ie$$

We can now define a local entity as follows:

Definition 4. ***Local Entity***
We define the set of local entities $E_{local}(d)$ of document d as

$$E_{local}(d) = \{e \in E(d) \mid PLD(e) = PLD(d)\}$$

Definition 4 is closely related to a similar notion defined in [7], which defines locality based on the correspondence of hostnames. Note that, according to this definition, an entity may be local to several documents, which may not always be desirable. Alternatively, one could focus on the authoritative relationship between entities and documents whereby the document an entity redirects to is the authoritative document for that entity [18]. In this paper, we currently only consider the locality relationship between documents and entities and plan to investigate stronger notions such as authoritativeness in future work.

4. CHANGE DETECTION MECHANISM

So far, we have focused on identifying and formalising different notions of change – particularly change functions – as a foundational aspect of dataset dynamics. We now discuss how such changes can be detected; one can group *change detection mechanisms* as follows:

- *HTTP-metadata monitoring*: analysis of HTTP response headers – including datestamp and ETag [11] – to detect whether something has changed;
- *content monitoring*: fetching the entire content and determining locally what has changed;
- *notification*: active notification by a data source that something has changed (ideally *what* has changed) [16].

The Table 1 summarises aspects of the the aforementioned change detection mechanisms. The aspects – motivated by [10] – are as follows: (i) *availability*, meaning if the respective solution is available out-of-the-box in currently deployed systems on the Web; (ii) *reliability*, referring to the ability to correctly capture **all** changes; (iii) *costs*, referring to the resources needed for the approach (in terms of band-width, storage, etc.); and (iv) *scalability* with respect to the number of involved data publishers (in terms of infrastructure) and consumers (concerning, for example number of concurrent “subscribers” in a notification system). Further, we have included two Linked Data specific aspects in Table 1: (v) support for *document-centric* change detection, and (vi) support for *entity-centric* change detection.

	Content	**HTTP**	**Notification**
availability	+	± [10]	± [16]
reliability	+	± [10]	unknown
costs	high	low	unknown
scalability	high	high	unknown
documents	yes	yes	yes
entities	no	partially	yes

Table 1: Change detection mechanism’s aspect matrix.

Both content and HTTP metadata monitoring mechanisms are well studied and discussion about those is available elsewhere (cf. [10, 11]). The characteristics of Web-scale notification mechanisms – especially concerning reliability, costs, and scalability are subject to research at time of writing. However, there are some remarkable implementation and standardisation efforts ongoing, including but not limited to:

- online services;[2]
- earlier efforts for a lightweight notification standard: for instance the *Event Notification Protocol* (ESN) (see “Requirements for Event Notification Protocol” [23]);
- *pubsubhubbub*: a simple, open, server-to-server webhook-based pubsub (publish/subscribe) protocol as an extension to Atom and RSS.[3]

5. METHODOLOGY

To the best of our knowledge, this is the first study regarding the dynamics of documents and entities of the Linked Open Data Web. Hence, the methodologies used in our evaluation are inspired by legacy related work for Web documents. Specifically, we applied similar evaluation methods – and indeed try to answer similar questions – as presented in [8]. The experiments require a large data set which is constantly monitored over a long timespan to conclude significant findings: we are not aware of any significant, heterogeneous and publicly available data-set of Linked Open Data resources which includes a complete history of changes. Nevertheless, we have access to such a dataset collected for an extended period in early 2009; although the dataset was originally collected for a different purpose – and thus, as we will see is not as suitable for our analysis as a bespoke corpus might be – we can derive some illustrative statistics which give some early insights into the dynamic nature of Linked Data on the Web.[4] Next, we describe how this dataset was monitored and which methods we use for our evaluation.

5.1 Monitoring

To gain first insights about the dynamics of resources of the Linked Open Data Web we analyse 24 data dumps collected by weekly snapshots of the 7 hop neighborhood of Tim Berners-Lee’s FOAF file[5]. The weekly snapshots were collected using the MultiCrawler framework [14] with the following steps applied in each crawl cycle:

1. gathering the content of a list of URIs;
2. parsing of RDF/XML content;

[2] `http://www.changedetection.com/`

[3] `http://code.google.com/p/pubsubhubbub/`

[4] Notably, this dataset was already studied by Biessmann et. al. [4] w.r.t. to dependency dynamics between people described in the data set.

[5] `http://www.w3.org/People/Berners-Lee/card`

3. extracting of all URIs at the subject and object position of a triple;

4. shuffling list of extracted URIs;

5. applying a per-domain limit for the URIs (5000 URIs per PLD).

Please note that steps 4) and 5) were done for politeness reasons to prevent too many parallel HTTP requests to one server: these steps introduce a non-deterministic element into our crawl and thus, we did not monitor a fixed list of URIs every week. Indeed, this passive monitoring makes change frequency analysis more challenging [9]. We have to deal with an incomplete history of sources, wherein it is very likely that many sources appear only once in the snapshot – thus, we sometimes present statistics which use only a small subset of the total dataset: the subset derived from sources that were available in more than 20 of the 24 snapshots.

5.2 Data Corpus

The data collection was performed over 24 weeks starting from the 2nd of November 2008 and contains 550K RDF/XML documents with a total of 3.3M unique subjects (∼6 entities appearing in the subject position per source) with 2.8M locally defined entities per our definition 4.

5.3 Change detection function

The change detection of a document $C_d(t, t‘)$ or entity $C^e_d(t, t‘)$ between two snapshots $t, t‘$ is a trivial task as long as the statements of the resource do not contain blank nodes [24]. For our preliminary evaluation, we used a simple change detection algorithm – based on a merge-sort scan over the weekly snapshots – as follows:

1. skolemise blank nodes within a document;

2. sort all relevant statements for the change detection of an document or entity by their syntactic natural order (subject-predicate-object-[context]);

3. perform pairwise comparison of the statements by scanning two snapshots in linear time;

4. trigger a detection of change (either w.r.t. a document or entity) as soon as the order of the statements differs between two snapshots (e.g. new statements were added or removed).

5.4 Evaluation

In this subsection, we describe in detail the evaluation we performed on the data set.

Document-centric evaluation. Firstly – and as a baseline – we performed a document-centric evaluation which allows us to compare our results with earlier studies about HTML documents. For this study, we compute the changes of a document as defined in Definition 1.

Entity-centric evaluation. Secondly, we studied the change frequency of entities from an entity-per-document perspective as defined in Definition 2. In fact, more accurately we analysed the change frequency from a local-entity-per-document perspective – a notion which follows intuitively from Definitions 2 and 4: to detect a change in an entity $C^{e_{local}}_d(t, t‘)$, we compare only the statements which 1) are contained in documents whose URIs matches on the PLD level with the entity URI and 2) in which the entity URI appears in the statement. Thus, we consider only the changes from documents in the locality of the entity as defined in Definition 4.

5.5 Change Process - A Poisson Process

Finally, for the purposes of comparison, we use an established model for changes of Web documents. Previously published studies [8] report that changes in Web documents can be modeled as a Poisson process (Equation 1). Poisson processes are used – for example – to model arrival times of customers, the times of radioactive emissions or the number of sharks appearing on a beach in a given year. The model allows to calculate the probability of a number of events occurring in a fixed period of time given that (i) the events are independent of the time elapsed since the last event and (ii) the events occur with a known average frequency rate λ. The parameter λ is the expected ‘events’ or ‘arrivals’ that occur per the required unit-of-time (in our case, a week). Further, let $\mathtt{N}(t + \tau) - \mathtt{N}(t)$ be the number of changes in an interval $(t, t + \tau]$ with τ given as the number of weeks – to take an example in our scenario, if an entity e changed five times in a window of the last 10 weeks, then $\mathtt{N}_e(14+10)-\mathtt{N}_e(14) = 5$. Finally, let k be the number of occurrences of a document or entity in the total monitoring time (24 weeks in our case). Then, according to the Poisson process, the probability of an event occurring within a given interval $(t, t + \tau]$ is given as:

$$P[\mathtt{N}(t+\tau) - \mathtt{N}(t) = k] = \frac{(\lambda\tau)^k}{k!}\exp(-\lambda\tau) \text{ for } k = 1, 2... \quad (1)$$

6. FINDINGS

In this section we present several early findings about the change frequency of resources on the Linked Data Web.

Firstly, we examine the usage of `Etag` and `Last-Modified` HTTP header fields, followed by an analysis of the various dynamic aspects which are aligned to the studies of the traditional Web in [8].

6.1 Usage of `Etag` and `Last-Modified`

One way to detect changes is to use the information contained in HTTP response headers as discussed in Table 1. The HTTP protocol offers two header fields to indicate a change of a document, viz: the `Etag` and `Last-Modified` fields. Using such methods of change detection is more economical in that it avoids the need for content sniffing.

We verified the usage (or lack thereof) of these two fields for all the documents in our corpus; Table 2 summarises the findings:

Header field	Fraction
only `Etag`	7.12%
only `Last-Modified`	8.18%
Both	16.75%
None	67.95%

Table 2: Usage of `Etag` and `Last-Modified` HTTP header fields.

Similarly to studies about the usage of these two fields for HTML documents [10], we found that 67.95% of the 550K documents did not report either of these two fields. Both fields were available by 16.75% of all the documents. Thus, we have to rely on actively monitoring of documents to detect their changes.

6.2 Access and lifespan distribution

We move now to analysis involving the content of data in our corpus. Firstly, we are interested in characterising the distribution of the number of accesses (i.e., appearances) and the lifespan (i.e., the time interval between the first and last appearance) of documents and entities respectively. This is a slightly different computation from [8], where, for example, the authors estimated the lifespan of a document by doubling the time the document was seen in the monitoring window if the document occurred at the beginning of the experiment but not at the end. Figure 2 contains the plots of the frequency and lifespan distribution for the documents (*crosses*) and entities (*circles*); we observe that the distributions follow approximately an “80-20” law.

From this figure, we can also conclude that only a fraction of the documents appeared frequently in the different snapshots – considering the importance of having as much information as possible to apply and verify our change frequency model (Section 5.5) – and thus to gain a good overview about their dynamics – going forward, we will give special consideration to the subset of our corpus derived from documents that appear in at least 20 weekly snapshots and ignore missing observations when considering changes. Again, this is necessitated by the non-deterministic factor in our incidentally crawled snapshots.

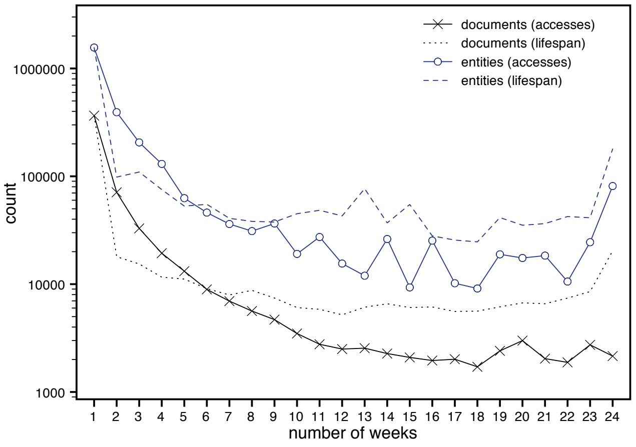

Figure 2: Access and lifespan distribution of entities and documents (y-axis logscale).

6.3 How often do the resources change?

Next, we will analyse the average change frequency of a resource. For the purposes of this analysis, we only consider the subset of the corpus which features resources that appear in more than 20 snapshots. Let us assume a document d changed 12 times during our monitoring interval of 24 weeks. In this case we can estimate the average change frequency of d to be 24 weeks/12 = 2 weeks. Following this example, the results for average change frequency of documents and entities are summarised in Figure 3. The left side of the diagram shows the percentage of all resources that were not observed to change (static resources). The right side of the diagram shows the percentage of non-static resources that were observed to have an average change frequency within the given interval. An interesting finding is that 62% of the total documents did not change at all, along with 68% of the entities. Further, we see that the fraction of documents is increasing with bigger change intervals, whereas for entities it is quite the opposite: by inspecting the data closer, we figured out that 51% of the entities with a change frequency of less than 1 week appear in more than one ‘local’ document. Thus, for example, one document may change the description of many local entities: along these lines, Figure 4 shows the distribution of the number of frequency of entities appearing in a given number of documents, where again, we can observe a power law distribution.

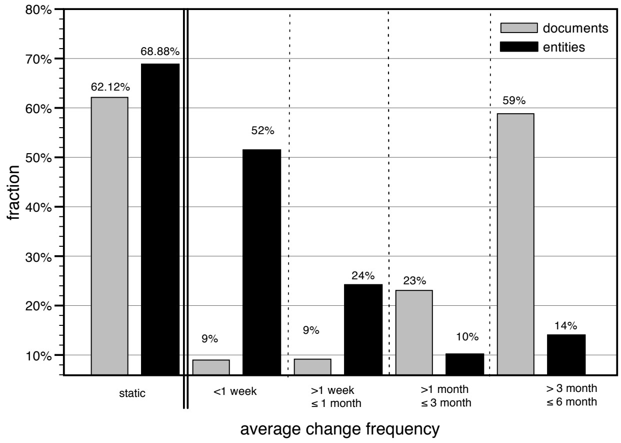

Figure 3: Fraction of documents with given average change frequency.

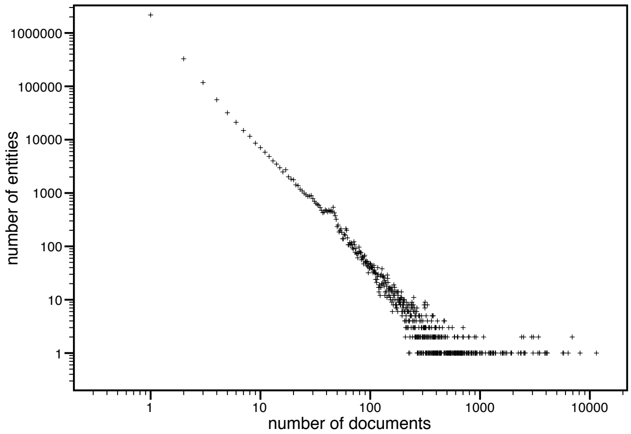

Figure 4: Distribution of reuse of entities among documents (log/log scale).

6.4 What fraction of the Web changed?

Continuing, we now study how quickly and what fraction of the documents and entities changes over time. Along these lines, we count how many documents – and respectively entities – changed after a certain time period. Figure 5 presents the cumulative change function for documents (*circles*) and entities (*squares*). The graph cumulatively shows how many documents and entities had changed after X weeks. The plot contains the cumulative change function for all resources (appearing at least once), and for resources that appeared in at least 20 snapshots. Again, the plot correlates with Figure 3 in that for the subset of the corpus with more than 20 observations, we can also see a large amount of entities changing after the first week, with a more gradual increase in observed document changes. An interesting observation is that the entities with more than 20 observations show a higher propensity to change; one could assume that such entities are better linked (and thus appear more often in our crawl) and so are reused in more documents (cf. Figure 4).

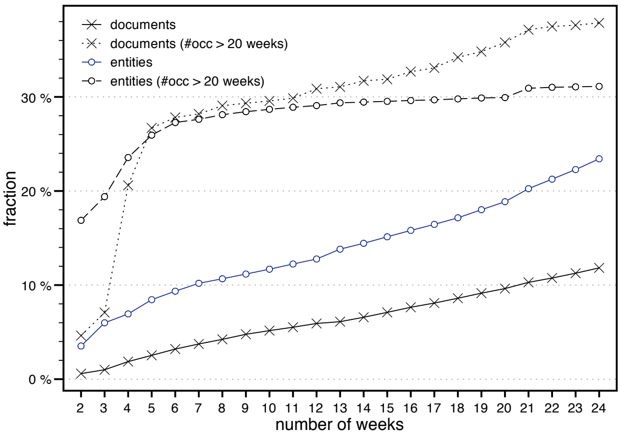

Figure 5: Cumulative change function.

6.5 Change process - A mathematical model?

Next, we analyse whether we can apply the Poisson model presented in Section 5.5 to the changes of documents and entities detected in our analysis. Therefore, we must compute the average change rate λ for each document d and entity e. We group the documents and entities with the same change rate and plot their distribution of successive change intervals; e.g., a document which changed in week 2 and 6 has a successive change interval of 4. If the changes can be modeled as a Poisson process, the resulting graph should be distributed exponentially.

For illustration, we selectively present the graph for documents with an average change frequency of 4 weeks (Figure 6) and the graph for entities with an average change frequency of 4 weeks (Figure 7). We performed a Poisson regression (log-linear regression) and use the maximum likelihood method to estimate the parameters. The predicated poisson process is plotted in the graphs as the line and describes the observed data quite well, despite some small variations. Similar effects are observed for around half of the other plots. However, we also spotted several graphs for documents and entities in which the Poisson model does not well describe the observed data points: The main reason for this observation is that there are not enough available sample points. As a conclusion of the findings: we currently cannot accept or reject the described change model with statistical significancy. Further studies with more data samples are required.

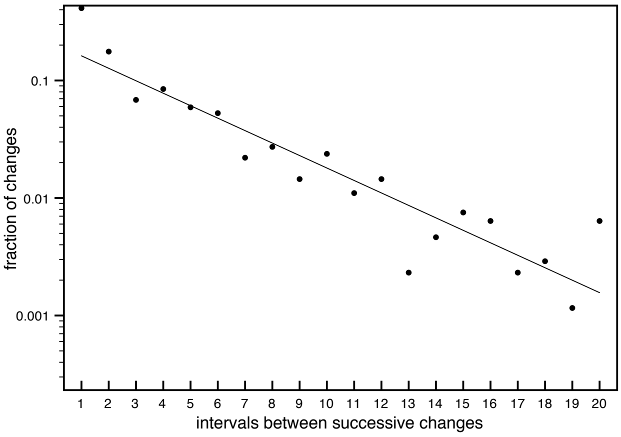

Figure 6: Documents with an average change frequency of 4 weeks (#occ >20 weeks, y-axis logscale).

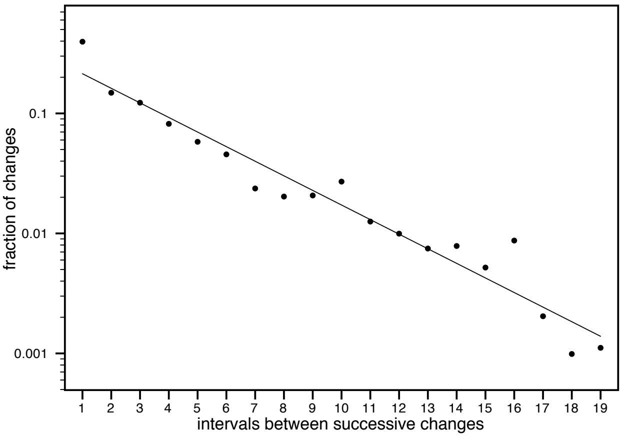

Figure 7: Entities with an average change frequency of 4 weeks (#occ >20 weeks, y-axis logscale).

6.6 Discussion of the results

We found that in 90% of all documents less than 10% of the entities changed, as depicted in Figure 8, which shows the distribution of the average fraction of entities that changed per document. It is hence safe to assume that – in the context of Linked Data – the finer-grained entity-centric perspective for changes is superior, compared to the more traditional document-centric point of view.

Drawing towards a conclusion to our analysis, we now discuss the observed changes for the documents over time. Therefore, we defined the following three main change categories:

- Update (U) – that is, between two snapshots of a document, the entities described were the same but the information about the entities changed: new statements were added and/or removed;

- Add (A) – that is, between two snapshots of a document, new entities were added;

- Del (D) – that is, between two snapshots of a document, entities were deleted;

- Combination of the three categories mentioned above: UA, UD, AD; UAD.

Table 3 lists the fraction of documents which encountered such a change (or combination thereof) for each of the seven categories. We can see that 76% of the documents have only entity updates as changes, whereas in 9.46% of the documents new entities were added.

	U	A	D	(UA|UD|AD)	total
U	76.88%	9.46%	7.08%	3.87%	97.29%
A	9.46%	0.19%	2.29%	3.87%	15.81%
D	7.08%	2.29%	0.23%	3.87%	13.5%

Table 3: Results of document change categories.

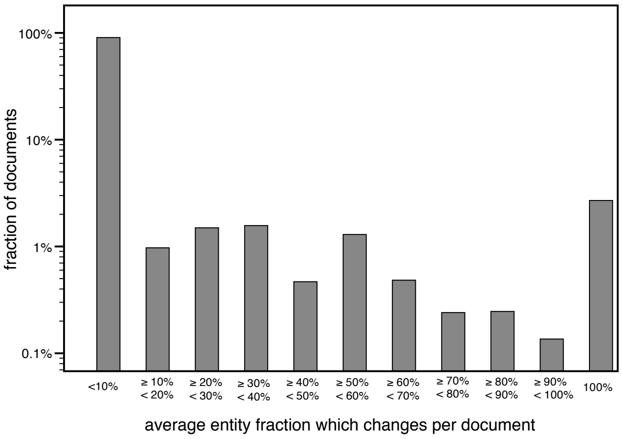

Figure 8: Average fraction of entity changes per document (y-axis logscale).

7. CONCLUSION

We motivated this work by highlighting the importance of a fundamental understanding of dataset dynamics with respect to Linked Open Data sources; we further claim that such knowledge can be leveraged to optimise existing systems and algorithms, such as making incremental index updates techniques more efficient. Further, we discussed in detail the differences between document-centric and entity-centric dynamics together with possible approaches for change detection: content-monitoring, HTTP header monitoring, and active notifications.

The findings we gained from weekly snapshots of the neighborhood graph of Tim Berners-Lee FOAF file are the following:

- less than 35% of the monitored documents contained `Etag` and `Last-Modified` HTTP header fields in the response;

- a surprisingly small amount (∼ 35%) of the monitored resources changed over the time interval of 24 weeks;

- half of the documents that changed had a change frequency of more than 3 months – in contrary, on a entity-centric level, half of the entities had a change frequency of less than a week applying our definition of local entities (based on PLD correspondences between document and entity);

- comparing our results to previous published studies we cannot verify that the change frequency of the documents and entities follow entirely the change model of a Poisson process.

We should perhaps look at these early findings with a critical eye in that we did not actively monitor a fixed set of sources. This work is very much an early attempt in this field, and needs further exploration and research to fully understand and exploit the change frequency of resources in the Linked Data Web.

7.1 Future Work

Large scale experiment To verify our early findings and derive statistical significant results, we plan to expend and run our evaluation for a larger dataset which is monitored over a longer time period. Further, we plan to study in more detail the dynamics on a entity-centric level; e.g. studying the dynamics of only authoritative entities as defined in [18] or the dynamics of the global entities as defined in Section 3.

Active monitoring. A major drawback of the current study is the monitoring method used for our data set. To overcome the problem of an incomplete change history, we will actively survey a selected set of documents over a long time period, thus creating a tailored corpus for our analysis. In addition to active monitoring, we plan to study how we can dynamically adapt the monitoring interval based on the estimated change frequency of a resource.

Fine-grained analysis of changes on a entity-centric level Finally, the findings of this work will be integrated into an existing system which aims to execute live queries over the LOD Web, which uses efficient data summary approaches [13]. Thus, using our analytics, we would hope to discern documents which are highly dynamic and those which are more static: highly dynamic documents would thus be better suited to direct-lookup approaches, whereas static data would be more suited to index summaries (or indeed, full-blown data warehousing approaches) for query-answering. Similarly, we could also investigate what kinds of statements for an entity in a document changes; e.g. a rdf:type statement should be rather very static, whereas a statement describing the values of sensor data is rather very dynamic.

8. ACKNOWLEDGEMENTS

Our work has partly been supported by the European Commission under Grant No. SFI/08/CE/I1380 (Lion-2), and under Grant No. 231335, FP7/ICT- 2007.4.4 iMP project. Further, this work has greatly benefited from the feedback of Bernhard Haslhofer and Nico Popitsch as well as from discussions with Andreas Harth.

9. REFERENCES

[1] Dataset dynamics (esw wiki). `http://esw.w3.org/topic/DatasetDynamics`.

[2] REST and RDF Granularity. http://dret.typepad.com/dretblog/2009/05/rest-and-rdf-granularity.html, May 2009.

[3] R. Almeida, B. Mozafari, and J. Cho. On the evolution of wikipedia. In *Int. Conf. on Weblogs and Social Media*, 2007.

[4] F. Biessmann and A. Harth. Analysing dependency dynamics in web data. In *Linked AI: AAAI Spring Symposium "Linked Data Meets Artificial Intelligence"*, 2010.

[5] C. Bizer, T. Heath, and T. Berners-Lee. Linked Data—The Story So Far. *Special Issue on Linked Data, International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(3):1–22, 2009.

[6] B. E. Brewington and G. Cybenko. How dynamic is the web? *Comput. Netw.*, 33(1-6):257–276, 2000.

[7] G. Cheng and Y. Qu. Term dependence on the semantic web. In *ISWC '08: Proceedings of the 7th International Conference on The Semantic Web*, pages 665–680, Berlin, Heidelberg, 2008. Springer-Verlag.

[8] J. Cho and H. Garcia-Molina. The evolution of the web and implications for an incremental crawler. In *VLDB*, pages 200–209, 2000.

[9] J. Cho and H. Garcia-Molina. Estimating frequency of change. *ACM Trans. Internet Techn.*, 3(3):256–290, 2003.

[10] L. R. Clausen. Concerning Etags and Datestamps. In *Proceedings of the 4th International Web Archiving Workshop*, 2004.

[11] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, Y. Lafon, M. Nottingham, and J. Reschke. HTTP/1.1, part 6: Caching. Internet Draft, Expires: April 29, 2010, IETF HTTPbis Working Group, 2009.

[12] R. Fielding and R. Taylor. Principled design of the modern Web architecture. *ACM Trans. Internet Technol.*, 2(2):115–150, 2002.

[13] A. Harth, K. Hose, M. Karnstedt, A. Polleres, K.-U. Sattler, and J. Umbrich. Data summaries for on-demand queries over linked data. In *Proceedings of the 19th World Wide Web Conference (WWW2010)*, Raleigh, NC, USA, Apr. 2010. ACM Press. accepted for publication.

[14] A. Harth, J. Umbrich, and S. Decker. Multicrawler: A pipelined architecture for crawling and indexing semantic web data. In *International Semantic Web Conference*, pages 258–271, 2006.

[15] M. Hartung, T. Kirsten, and E. Rahm. Analyzing the evolution of life science ontologies and mappings. In *DILS '08: Proceedings of the 5th international workshop on Data Integration in the Life Sciences*, pages 11–27, Berlin, Heidelberg, 2008. Springer-Verlag.

[16] B. Haslhofer and N. Popitsch. DSNnotify - detecting and fixing broken links in linked data sets. In *Proceedings of the 8th International Workshop on Web Semantics (WebS 09), co-located with DEXA 2009*, 2009.

[17] M. Hausenblas, W. Halb, Y. Raimond, and T. Heath. What is the Size of the Semantic Web? In *I-Semantics 2008: International Conference on Semantic Systems*, Graz, Austria, 2008.

[18] A. Hogan, A. Harth, and A. Polleres. Scalable Authoritative OWL Reasoning for the Web. *Int. J. Semantic Web Inf. Syst.*, 5(2), 2009.

[19] I. Jacobs and N. Walsh. Architecture of the World Wide Web, Volume One. W3C Recommendation 15 December 2004, W3C Technical Architecture Group (TAG), 2004.

[20] H.-T. Lee, D. Leonard, X. Wang, and D. Loguinov. Irlbot: Scaling to 6 billion pages and beyond. *ACM Trans. Web*, 3(3):1–34, 2009.

[21] B. S. Lerner and A. N. Habermann. Beyond schema evolution to database reorganization. In *OOPSLA/ECOOP '90: Proceedings of the European conference on object-oriented programming on Object-oriented programming systems, languages, and applications*, pages 67–76, New York, NY, USA, 1990. ACM.

[22] S. Pandey, K. Ramamritham, and S. Chakrabarti. Monitoring the dynamic web to respond to continuous queries. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 659–668, New York, NY, USA, 2003. ACM.

[23] S. Reddy and M. Fisher. Requirements for Event Notification Protocol. Internet Draft, May 1, 1998, IETF WEBDAV Working Group, 1998.

[24] G. Tummarello, C. Morbidoni, R. Bachmann-Gmür, and O. Erling. Rdfsync: Efficient remote synchronization of rdf models. In *ISWC/ASWC*, pages 537–551, 2007.