THE QUALITY OF THE XML WEB

Steven Grijzenhout
University College London
Department of Management Science and Innovation
Gower Street
London WC1E 6BT, United Kingdom
steven.grijzenhout.10@ucl.ac.uk

Maarten Marx
ISLA, Informatics Institute
University of Amsterdam
Science Park 904
1098 XH Amsterdam, The Netherlands
maartenmarx@uva.nl

ABSTRACT

We collect evidence to answer the following question: Is the quality of the XML documents found on the web sufficient to apply XML technology like XQuery, XPath and XSLT? XML collections from the web have been previously studied statistically, but no detailed information about the quality of the XML documents on the web is available to date. We address this shortcoming in this study. We gathered 180K XML documents from the web. Their quality is surprisingly good; 85.4% is well-formed and 99.5% of all specified encodings is correct. Validity needs serious attention. Only 25% of all files contain a reference to a DTD or XSD, of which just one third is actually valid. Errors are studied in detail. Automatic error repair seems promising. Our study is well documented and easily repeatable. This paves the way for a periodic quality assessment of the XML web.

Categories and Subject Descriptors

H.m [Information Systems]

General Terms

Measurement, Reliability, Standardization.

Keywords

XML, XML Web, Schemas, Data Quality.

1. INTRODUCTION

In this study we look at the prospects of using XML technology for information extraction and integration tasks on XML data found on the World Wide Web. Without becoming specific we refer to the multitude of these tasks as Extract-Transfer-Load (ETL) tasks [31]. Examples of ETL subtasks are data harvesting, text extraction, structure extraction, text mining [22], data de-duplication [13], data exchange (from one schema to another) [12] and data publishing (from one format (e.g. XML) to another (e.g., RDF or relational)).

Apart from the actual collecting of data from the web, all of these tasks can be expressed in the three XML query languages, XPath 2.0, XSLT 2.0 and XQuery 1.0. Not only can these tasks be expressed in these languages, when the input is XML it is desirable to do so for a number of reasons. XSLT and XQuery programs are largely declarative. The semantics of the languages is clear and well-defined. The languages are vendor and software independent, developed and maintained by a committed community and became W3C standards. The immense success of SQL shows the great software engineering benefits of working with such programming languages. Maintainability of code is crucial for ETL tasks as they are typically applied in a changing environment not under control of the developers of the ETL code.

Whether it is *feasible* to use XML technology for ETL tasks depends on many factors. This is out of the scope of this study. Here we only look whether it is *possible*. That is, is the quality of the XML documents found on the web sufficient to apply XML technology?

Another reason to study the XML web is the new XQIB[1], XQuery In the Browser, initiative. XQIB is an alternative to JavaScript. Obviously it needs XML of good quality.

Previous studies on HTML showed that the vast majority of HTML documents (around 95%) on the web did not comply with the standards set by the World Wide Web Consortium [10][28][29]. For XML, studies that measure basic quality indicators (like being syntactically correct) on arbitrary XML data from the web have not been performed yet. There are several empirical studies on XML but they either use data from repositories or have very small samples and always contain only well-formed XML (Cf. Section 2).

Unhappy with this omission and frustrated by our own efforts of using XML tools for a large data integration project we set ourselves the following research goal:

> Create a corpus of XML documents and accompanying schemas that is representative of the web, evaluate which part is ready to be processed with XML tools, and evaluate the prospects of automatic error correction for the other part.

The paper describes the created collection (Section 3), and the evaluation of its quality (Section 4). We also created a corpus of schemas in the three XML schema languages and evaluated their inter-translatability (Section 4.3). The remainder of this introduction consists of our operationalization of XML-quality and an overview of the main results. Related work is presented in Section 2.

1.1 Basic quality requirements

One can only apply XML tools to XML files if they satisfy a few basic but important properties. As XML is a self-describing format, these properties all state that files should not lie about themselves concerning some aspect X. We looked at three

[1] http://www.xqib.org/

aspects: a file should not lie about its encoding, it should not state that it is XML when it is not, and it should not lie about its validity with respect to a schema. More precisely,

1. The document should be encoded using a single encoding that is stated in the document.
2. The document should be well formed XML.
3. If the file references a schema, that should be useful and truthful. This means that
    a. the URI identifying the schema should be resolvable. Also all included schema files should be resolvable recursively;
    b. all these schemas are syntactically correct, and
    c. the file is valid with respect to the schema(s).

We collected almost 180K unique XML files from the web from almost 100K websites with a total size of 40GB. We now summarize the main results. Our first result states that encodings do not pose a real problem as 99.5% had a correctly specified encoding. The other results are neatly summarized in Figure 1.

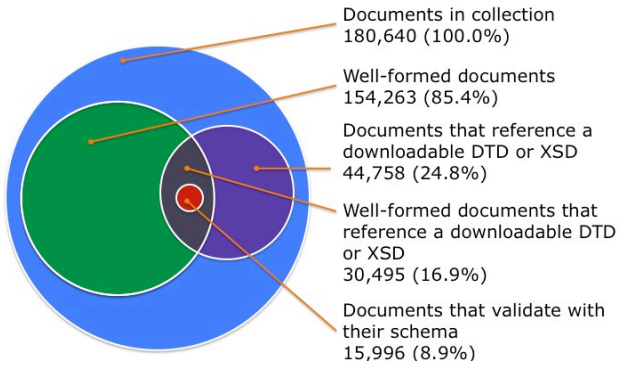

Figure 1: Summary of the Quality of the XML Web.

If we pick a random XML document there is a 14.6% chance that it is not well-formed. This is much better than could be expected from earlier studies on HTML. Interesting is the position of the subset of documents that reference a DTD: 66.4% is not well-formed, almost five times more than on average. It seems that the addition of a DOCTYPE declaration is often added to hide their own poor quality.

Validity is rare on the web. Just over 10% of the well-formed documents are also valid. If we zoom in on validity we see very different patterns for DTD and XML Schema. We go through the three possible problems. The first problem is to reference a schema that cannot be retrieved. This happened in 12.5% of all references to a DTD. Things get subtle once you realize that DTDs can also include other DTDs, and these have to be retrieved as well. Of the 5410 include statements in DTDs in our corpus, 33% could not be downloaded. Includes in XML Schema behave much better: of 2110 includes only 23 could not be retrieved. XML documents which claim to be valid with respect to an XML Schema behave very well: they have 99% chance of being well-formed, which is much better than the average 85.6%. Figure 2 and Figure 3 show those files that reference a schema but could not be validated. The figures present the causes for non-validation.

The differences are remarkable. Files referencing a DTD are for 73% not valid just because they are not even well-formed. For XML Schemas this cause of error is negligible. Conversely, 31% of all XML Schema validity errors are due to a schema that is useless because it is not even syntactically correct. This happens in only 4% of the DTD-errors.

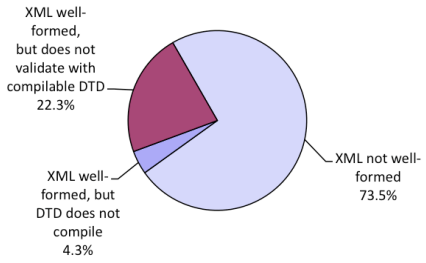

Figure 2: Distribution of causes for non-validation: DTD.

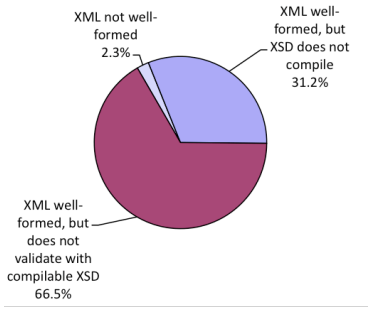

Figure 3: Distribution of causes for non-validation: XSD.

1.2 Main contributions

Our main contribution is an up-to-date and reliable estimate of the quality of the state of the XML web in 2010. Our second contribution consists of an extensive analysis of the type of errors that compromise the quality of the XML web. As they are mostly Pareto distributed we believe this can be used to guide research into automatically repairing errors.

Our third contribution is the collected data itself. All data is made publicly available in a uniform format at the url http://data.politicalmashup.nl/xmlweb. All referenced schema files are (recursively) locally available. Information about headers, encodings, and errors of each XML file is stored in a relational database that is also available. Also all scripts and settings for crawling and analyzing the collection are available and well-documented. This makes our study easily repeatable. We hope that this is a start of a longitudinal XML collection. Of course we also hope to see a steady improvement of the quality of the XML web.

Our last contribution consists of a corpus of schema files in the four main schema languages and an analysis of their inter translatability with James Clark's Trang.[2]

2. BACKGROUND

A large number of descriptive studies on XML have been conducted. There are three main themes identifiable in the literature, which will be discussed accordingly: studies on XML collections; studies on the quality of the HTML web; and studies on XML schema languages.

[2] http://www.thaiopensource.com/relaxng/trang.html

2.1 Studies on XML collections

Studies on XML document collections mainly differ in sample data [34]. A study has been done on 200,000 publicly available XML documents from the Xyleme repository [26]. This collection contains only well-formed XML documents. Another study used a number of XML collections, consisting of 16,534 documents and accounting for a total size of 20 Gigabytes [27]. The collections include well-known docbook samples, XML bibles, RDF samples and IMDb collections. In Section 3.2 we compare our collection to these two. Lastly, a third study used 601 XHTML web pages, 3 DocBook XML documents, and documents from the XML Data repository[3] project [19].

Macro-level analysis shows that XML documents are found in all geographic regions and across all major internet domains. 53% of all documents, accounting for 76% of the total file size, can be found at '.com' and '.net' internet domains [26].

Only 48% of the XML documents references a DTD, and 0,09% an XML Schema [26].

Most XML documents are small: around 4 Kilobytes. Also, the volume of markup in relation to the actual content of the documents is surprisingly high. Lastly, 99% of the documents had less than 8 levels of element nesting, and 15% appears to have recursive content. This all seems to indicate that most XML documents are not complex [19][26].

2.2 Studies on HTML web quality

Several surveys on the quality of HTML documents on the web exist [28][10][5][29]. Although XML's predecessor HTML differs greatly in applicability, these studies are relevant because of their approach.

The sample data across the studies differ between 226 web sites from environmental issues [29], 13,312 websites under the 'co.uk' domain [5], samples that combined websites from search engines and Alexa.com's top web sites [9], and homepages of the Alexa.com's top 100,000 web sites [28].

The studies use different methods to assess the quality of HTML documents: WebXACT [29], NSGMLS parser [5] and the W3C HTML Validator [10][28].

The differences in sample collections and quality measures do not make a large difference in results. All indicate a poor quality of the HTML web: a mere 6.5% [5], 5% [9], 4% [29] and 3% [28] of the HTML documents complied with W3C's HTML standards.

2.3 Studies on XML schema languages

XML schema languages describe the structure of XML data and are almost inseparable from XML. They allow automization and optimization of search, integration and processing of XML data [8].

There are three main schema languages and one language for specifying dependencies in use. These are DTD[4], XML Schema[5] (abbreviated as XSD), Relax NG[6] and Schematron[7]. Schematron is the language used for expressing dependencies in the form of implications between tree patterns. As it is rarely used we will not discuss it further. The three other schema languages are all W3C recommendations. XSD and Relax NG documents are themselves written in XML. Relax NG also has a compact syntax. DTDs have their own syntax. No research yet exists on the actual use of Relax NG schemas in documents on the web.

XML schemes have been studied in a number of ways. Firstly, XML schemes are studied in relation to XML collections. As we have seen above, only a small percentage of documents reference a schema. In the Xyleme sample 48% of the documents references a DTD, and 0.09% an XML Schema [26]. In the semi-automatic collection by Mlynkova et al. [27], however, only 7.4% does not reference a schema; this might be due to the collection process. Furthermore, results show that the XML documents are simple and specific in comparison to their XML schema; the schema is usually too general. As is the case with HTML files, the syntax of most DTD files is incorrect [11][32]. This is generally also the case for XML Schema [7].

Secondly, the properties of XML schemas are studied. Most of this work has focused on DTDs. DTDs differ greatly in size and forms. However, DTDs are generally simple [11][18]. Many features of DTDs are not used or misused; this indicates that the features are not properly understood. Also, there are many ways to do things in DTDs, and people use hacks to cope with DTD shortcomings [32].

Thirdly, work has been done in developing metrics to measure the properties of DTDs [18] and XML Schemas [25]. These metrics might be interesting to use in future versions of quality analysis.

Lastly, research is done in comparing the use of the different XML schema languages. The three languages are incomparable in expressive power and their effect when validating. For instance, validating a document with a DTD changes the document: default values are added. DTDs have no means to restrict data values to data types like string or integer while this is possible with Relax and XSD. Theoretical work on the expressive power of schema languages abstracts many features of the concrete languages and compares their core logical part. Then we see that DTD is less expressive than XSD, which is less expressive than Relax NG. The latter is equally expressive (on XML trees) as Monadic Second Order Logic (MSO) [6][24]. While XSDs allow expressions that cannot be expressed in DTD syntax, these extras are rarely used in practice [7][23]. In our study we look at these differences in expressive power from a pragmatic point of view: how often can schemas be inter-translated using an existing schema translator (Cf. Section 4.3)?

3. DATA

We briefly describe the collections of XML and schema files, and how they were obtained. More detail can be found on the webpage where all data can be downloaded.

3.1 Desired Data

The population of the data in this study is the XML web. The definition for the XML web used here will be: the subset of the web made of XML documents only [3]. The population data consists of all kinds of XML documents. RSS, Atom, XSL stylesheets, XSD data and XHTML are all written in XML, and are therefore part of the population the XML web.

The actual amount of files in the XML web is unknown. Obtaining an estimate of its size is intrinsically difficult [1]. The size of the population is, however, irrelevant in calculating a representative sample size. Unfortunately, collecting XML documents from the web is often not a simple random sample.

---

[3] XML Data repository: http://www.cs.washington.edu/research/xmldatasets/

[4] http://www.w3.org/TR/REC-xml/#dt-doctype

[5] http://www.w3.org/XML/Schema.html

[6] http://www.relaxng.org/

[7] http://www.schematron.com/

Because of this, it is not possible to calculate a required sample size.

We decided to harvest as many XML documents from the web. as possible. The objective of our study is to assess the quality of the XML Web, and a large collection will maximize the probability that errors are included in the collection.

Our data collection process does not access the Hidden Web [30]. As a consequence, our collections will not contain any data from the Hidden Web.

3.2 Description of Data

We describe some general statistics about the collection and compare it to two earlier studies. We look at the origins of the data similar to [3]. We end by describing the collection of schema files.

The XML collection contains 180,640 XML files. Table 1 shows that this is 5.1% smaller than the collection used by Barbosa et al. [3], but 992.3% larger than the collection used by Mlynkova et al. [27]. The total file size of the collection is 40 Gigabytes. The largest file in the collection is 683.7 Megabytes, and the smallest is 1 byte. The average file size is approximately 223 Kilobytes. The number of documents that has a duplicate is 1296. This means there is chance of 0.007% that a document has a duplicate. The highest number of duplicates of one file is 119.

The URLs in the collection allow us to describe the distribution of XML documents on the web. The regions from which the XML Web is hosted and served can be explored and, up to a point, the underlying institutional goals of the XML can be described. For example, XML documents hosted on educational domains or commercial domains. The URL of a document contains the site from which it was retrieved. We define a site as the combination of a base domain and the top-level domain. Typically this looks like w3.org. There are files from 96,650 sites in our collection. To gather meaningful data, we have clustered the results of the Web sites by zones, consisting of generic Internet domains and geographical regions. We used the zones defined by Barbosa et al. [3], with the only difference that we define the European Union as of June 2010.

Figure 4 shows that 38,197 sites (39.5%) in the collection are in the '.com' domain. The EU follows with 25,870 Web sites (26.8%), and the Rest of the World category accounts for 18,753 Web sites (19.4%). These results are in line with results by Barbosa et al. with the exception that in geographical terms North America (composed of North America, .edu, .gov and .mil) is under represented: it accounts for only 3% in our collection, while it accounts for at least 16% in Barbosa's collection. This might be due to the fact that the harvesting process of our new sample was located in The Netherlands as opposed to North America.

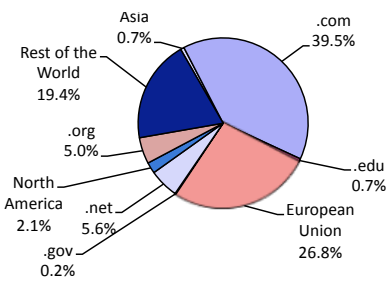

Figure 4. Distribution of web sites by Zone.

Table 1. Comparison of XML collections

	This paper	Barbosa et al [3]	Mlynkova et al [27]
Amount of files	180,640	190,417	16,538
Source	Selected from queries by Google and Yahoo	Randomly selected from Xyleme public repository (approx 500,000 files)	Semi-automatically from manually selected XML collections
Total size uncompressed	40 Gigabytes	843 Megabytes	20,756 Megabytes
Total size compressed (.tar.gz)	4.1 Gigabytes	151.4 Megabytes	--[a]
Amount of websites	96,650	19,254	133 Collections
Amount of duplicates	2430	26,989	--[a]
Type of duplicate detection	Hashing algorithm disregarding white spaces	Fingerprinting techniques	Simple hashing algorithm disregarding white spaces
Preprocessing	None	Collected from Xyleme database, which consists of well-formed XML files only.	Manually fixed most errors on XML, DTD and XSD files (including well-formedness, encoding & wrong usage of namespaces) Computer generated and random-content XML files were eliminated.

[a] The value is unknown.

With 180,640 documents and 96,650 sites in our collection, there is an average of 1.87 documents per site. The site 'gentoo.org' has most documents: 451, followed by 'thomann.de' with 207 documents. The distribution of documents per zone and of document size per zone largely mirrors the distribution of Web sites in Figure 4.

3.2.1 Schemas

The ability to reference a schema is one of the most important features of XML. We focused on DTDs and XSDs because they are used most often, and they are uniformly referenced in XML documents.

DTDs were downloaded by extracting the system identifier in the XML document header. All external referenced document includes have been downloaded recursively. Our collection contains 24,426 (13.5%) files with a reference to a DTD. 21,033 (86.1%) of all references use a public identifier, and 24,420 (99.9%) use a system identifier. We used the system identifier to download the DTDs. Of these, 3059 (12.5%) failed to download via the system identifier. The DTD schemas contained a total of 5410 includes of other DTDs or entity documents. These have been downloaded recursively, and the original schemas have been modified to include the locally downloaded included schemas.

1786 (33.0%) of them failed to download. The thus obtained collection contains 1375 DTDs.

XSD schemas have been extracted from references in the attribute labels "SchemaLocation" and "noNameSpaceSchema-Location". Includes have been downloaded recursively, and the original schemas have been modified to include the locally downloaded versions. The collection contains 24,087 files with a reference to an XSD (13.3%). There are files that contain multiple references to XSDs. The maximum amount of references in one file is 2399, and 90 documents have more than one reference to an XSD. Of the unique URLs with XSD schemas, 217 failed to download. A total of 2110 XSD includes were found. Of these, only 23 failed to download. The final collection consists of 437 XSDs. The most popular XSD[8] was referred in 82.5% of all files that reference an XSD. In contrast to DTD references, the list is not dominated by W3C schemas, but rather by sitemaps.org, indicating that XSDs are widely used for sitemaps.

Apart from collecting schema files that were referenced in an XML file, we also collected schema files directly using the same method of restricting searches at Google and Yahoo to specific file extensions. In this way we could also harvest Relax files. In total we have 3078 DTDs, 4141 XSDs, 338 Relax NGs in XML and 337 Relax NGs in the compact syntax. See http://data.politicalmashup.nl/xmlweb/trang.html.

3.3 Data Collection

The data were collected in the following 4 steps:

1. Crawl a list of URLs of XML documents from Yahoo and Google,
2. Download the content of each URL,
3. Organize the collection,
4. Determine duplicates.

For each URL, we store the URL, the HTTP header, the actual XML file, and recursively all schema files it references.

The list of URLs was created using a modified version of the crawler by Bex, Neven, & Bussche [7]. The crawler executes several keyword queries with the filetype restricted to XML. Only Yahoo and Google can limit search to XML files. The results of these two steps are described in Table 2.

Table 2. Statistics of URL List and Downloading

Filetype	Unique URLs in List	Files Downloaded	Loss Percentage	Last File Downloaded
XML	188,332	180,640	4.08%	2010-07-17

The resulting collection was organized in a MySQL database. The schema consists of nine main relations, which are described in the Appendix. For each file, the database stores its URL, its HTTP header, a list of its duplicates in the collection, information on the encoding, lists of all recursively referenced schemas, and all well-formedness and validity errors. The actual files are saved on disk with the appropriate id as its filename. Duplicates were not removed from the dataset, but rather a relation of duplicate content was inserted into the database.

4. QUALITY OF XML ON THE WEB

In three sections we look at the basic quality requirements outlined in Section 1.1: character encoding, well-formedness and validity. We are not only interested in the amount of errors but also whether a small amount of error-types is responsible for a large amount of errors. We also report correlations between errors and other variables.

4.1 Encoding

We checked whether documents lie about their encoding. Of every document in the collection, we checked whether the encodings as specified either in the HTTP header, in the encoding attribute in the XML declaration or in Content-Type meta tag (often used in XHTML documents) was compliant with the document. Reliable character encoding checking is difficult. To attain reliable results of this analysis, we have chosen to evaluate only those character encoding sets for which a reasonable reliable checking is available. Details are in the full paper. Our main result is that 99.47% of all specified encodings is correct. Figure 5 shows which encoding types give errors in our collection.

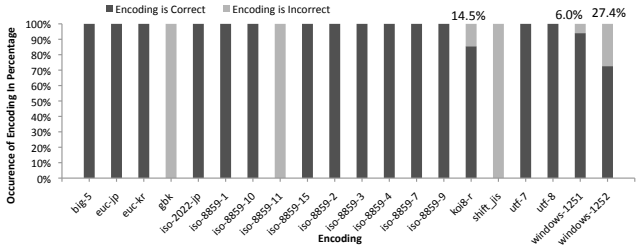

Figure 5. Percentage correct encoding per encoding type.

4.2 XML Well-formedness

We use the libxml2 parser to check whether a document is wellformed. We created a modified version of the XML parser libxml2. The main change was to make the error output uniform for all errors. The modified version of libxml2 distinguishes four different error levels: No error, Warning, Recoverable error and Fatal error. Files with only recoverable errors can still be parsed. The modified version of libxml2 also categorizes the error that occurred. We found 74 different error categories.

We note that libxml2 is not meant to collect statistics on the number of errors in a not well-formed XML file. It often happens that the parser outputs a large amount of errors while fixing just one makes the document well-formed. We however believe that the output is still useful for giving directions to research on automatic error repairmen.

26,377 different files (14.6% of the collection) had at least one fatal error. Figure 6 lists the 10 most common errors. Note that a document may have many errors, so the total sum is higher than the number of bad XML files. 'Opening and ending tag mismatch' is encountered in most documents (16,996 docs) followed by 'Premature end of data in tag' (14,250 docs). Third is an unknown encoding (11,615 docs). This last error does not necessarily have to be a fatal error, as libxml2 allows specifying the encoding of a document manually.

The distribution of errors across error categories follows a Pareto distribution. The Pareto Chart in Figure 7 shows the first nine error categories. Approximately 20% of the error categories (9 from a total of 74) account for 99% of the fatal errors.

[8] http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd

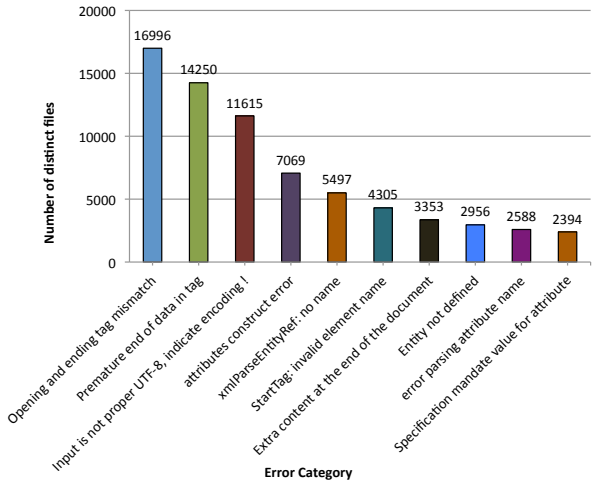

Figure 6. Top Ten Fatal Error Categories.

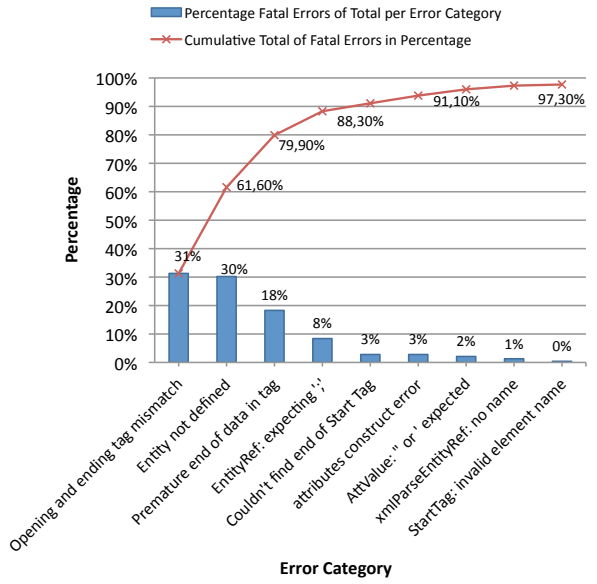

Figure 7. Pareto Chart of Fatal Errors per Error Category.

As indicated above, an error occurring need not mean that it is responsible for the file being not well-formed. Still, 5708 documents (21.6% of all documents with a fatal error), contain only one fatal error. Also here, 'Opening and ending tag mismatch' is most often (25.1%) responsible for not being well-formed.

We checked whether some internet zones were over- or under-represented in errors but found no significant deviations from Figure 4.

4.3 XML Validity

This section discusses the results about validity testing with respect to DTDs and XSDs.

4.3.1 Bad schemas

Figure 2 and Figure 3 give a breakdown of the reasons for non-validity of the files that reference a DTD or XSD. With both DTD and XSD being referred to in roughly 24K files there are large differences in their validity scores. Over half of the XML files with an XSD is actually valid, in contrast to less than 10% for files referencing a DTD. On the other hand, over half of the documents with a DTD is itself not even wellformed. With XSD, this occurs less than 1%. The most interesting case of invalidity occurs when all perquisites are satisfied. Here the two schema languages behave rather alike, with one sixth (DTD) and one fourth (XSD) of the files falling in this class.

4.3.2 Geographic distribution

We looked whether validity errors were over or under represented in certain domains and found one significant deviation. The .edu and .gov domains behave well compared to the rest: 2, respectively .4% of all files come from these domains, but of all files that refer to a well-formed DTD they contribute 11 and .9%, respectively.

4.3.3 Most common errors

First we look at errors in DTDs. A total of 28 different errors have been found, of which the top ten is shown in Figure 8.

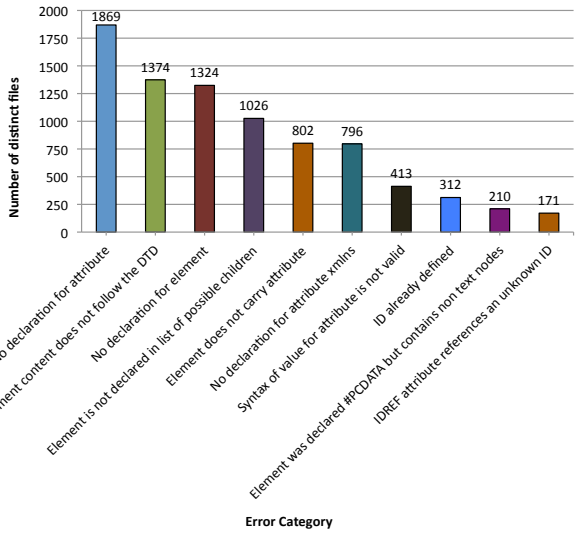

Figure 8. Top Ten Recoverable Error Categories in DTD validation based on occurrence in files.

The first two error-types are an indication that the data is in fact richer than the schema describes. If an application is built on the schema it can thus simply ignore the extra information. The third error type is problematic for parsers, and potentially difficult to repair automatically. It contains a lot of errors concerning CDATA. A text node is encountered where only element nodes are allowed. Some schemas have an obvious default element in which to wrap forbidden text nodes. E.g., in XHTML, text is forbidden under the body-element, and could be wrapped in a p-element.

This leads us to ask which DTDs have errors across the largest number of different files. Table 2 in the Appendix shows the top five DTDs leading to most invalid XML files. The list is dominated by W3C DTDs. The last column contains the part of all files with a DTD which refer to that specific DTD. The mobile and math DTDs are overrepresented, indicating that they might be hard to comply with (or people just don't care).

For XSDs we see very much the same picture. 93% of all validation errors are of type 'this element is not expected'.

Because XSDs can, in contrast to DTDs, also restrict the type of values, one could expect many type-errors. In fact these are quite uncommon: less than 5% of all errors and occurring in around 10% of all invalid files.

4.3.4 The good guys and the bad guys

Though not of direct practical value, it is fun to see which background variables correlate strongly with (in)validity. We looked at file size, domain extension, encoding and the type of webserver used for the site. Full details are in [14].

The Content-Length HTTP header is an interval variable, but not normally distributed[9]. We use Spearman's rho to determine if there is a relationship between file size and validation of the document. There indeed is a statistically significant but weak relationship, $r(131{,}831) = .214$, $p < .01$.

Regarding base domain, we did a binary logistic regression analysis. The produced model does indicate that domain name extension is statistically significant and explains variations in validity of the documents (χ^2=6087.791, df=334, $p < 0.01$). We found 33 domain name extensions with a statistically significant ($p < 0.05$) effect. The 5 bad guys are '.jp', '.org.au', '.cat', '.gov.uk', '.gov.br'. They are 3.2, 5.1, 3.6, 3.6 and 9.4 times more likely to be invalid than to be valid. The rest are good guys, ranging from 2.226 (.gov) to 24.750 (.im) more likely to be valid than invalid.

All seven statistically significant domain name extensions in the educational and academic domains (containing .edu or .ac) are more likely to be valid than invalid. In contrast, documents from governmental domains in the USA are more likely to be valid (.gov), while documents from two other governmental domains are less likely to be valid (.gov.br and .gov.uk). An other interesting fact is that documents from the .uk domain are generally almost 2.5 times more likely to be valid than invalid, while documents from the governmental domain in the uk (.gov.uk) are 3.6 times more likely to be invalid than valid. It might indicate that documents from the British government are of poorer quality than other documents originating in the UK

Does it matter for validity whether a site uses a commercial (Microsoft's ISS) or an open source (Apache) server? The effect is significant but extremely small: documents server by an Apache server are 1.07 times more likely to be valid than documents that are server by Microsoft IIS.

The encoding of a file has a minor effect on validity. The only statistically significant effects we found are for windows125(1|2) which is twice more likely to be invalid and iso88591 which is 2.5 times more likely to be valid.

4.3.5 Translations between schema languages

Because our collection does not contain any Relax NG schemas we also crawled schema files directly. Combined with the schema's already found we created a collection of 3087 DTDs, 4141 XSDs, 337 Relax NGs in compact syntax and 338 Relax NG's in XML. All schemas are syntactically correct. The data and the results can be found at http://data.politicalmashup.nl/xmlweb/trang.html.

While XSDs allow expressions that cannot be expressed in DTD syntax, these extras are rarely used in practice [6][7]. We wondered whether available schema translation software could support these findings on our dataset. We used James Clark's Trang as it can translate between all three schema languages except from XSD. The results are that Trang translates 88% of the DTDs to the other two languages and that 30% of the Relax schemas can be translated to DTD and 96% to XSD. The surprisingly low 88% seems due to Trang, not to the use of DTD features the other languages can not handle. The University of Dortmund is working on an improved translator, which can also translate from XSD.

[9] Kolmogorov-Smirnov test. *Sig value* < 0.05. N = 131,831

5. CONCLUSIONS

Our results show that it is possible to do ETL tasks on 'XML' files solely using XML query and transformation languages. Only 14.6% is not truly XML and the distribution of errors is promising for (semi-)automatic repair. Of course ETL development would become much easier and far more robust when restricted to valid XML. Here the quality of the XML web needs drastic improvement as less than 10% is valid. Although it is hard to compare our data with previous studies, the growth of referenced XSDs and the fact that files with an XSD tend to be twice as often valid as those with a DTD seems a positive development. We have set up our study in such a way that it can easily be replicated in the future. Hopefully we can measure an upward trend in validity.

The distribution of XML syntax errors follows an 80-20 law which make them amenable to automatic error repairing techniques. Validation errors occur because schemas do not compile or because the XML is not valid. This shows that work on (semi-)automatic learning DTDs or XML Schemas from XML documents is useful [8]. Most validation errors occur because there is an element or attribute used that is not defined in the schema. This could mean that either the schema is not correct or a wrong name is used in the XML. Schema learning techniques may be expanded to schema repairing techniques. Techniques used in data-deduplication and learning schema mappings seem useful to repair XML documents in this case.

6. REFERENCES

[1] Abiteboul, S., & Vianu, V. (1997). Queries and Computation on the Web. ICDT '97: *Proceedings of the 6th International Conference on Database Theory* (pp. 262-275). London, UK: Springer-Verlag.

[2] Azze-Eddine, M., Samia, K.-B., & Douniazed, A. H. (2004). XML-DFG : A Dynamic Forms Generator for XML Valid DTD Document. *RIST* , 14 (2), 15-26.

[3] Barbosa, D., Mignet, L., & Veltri, P. (2005). Studying the XML Web: Gathering Statistics from an XML Sample. *World Wide Web* , 8 (4), pp. 413-438.

[4] Beatty, P., Dick, S., & Miller, J. (2008, Mar/Apr). Is HTML in a Race to the Bottom? A Large-Scale Survey and Analysis of Conformance to W3C Standards. *IEEE Internet Computing* , 12 (2), pp. 76-80.

[5] Beckett, D. (1997). 30% accessible - a survey of the UK Wide Web. *Computer Networks and ISDN Systems*, 29 (Nos 8-13), pp. 1367-75.

[6] Bex, G. J., Martens, W., Neven, F., & Schwentick, T. (2005). Expressiveness of XSDs: from practice to theory, there and back again. *WWW '05: Proceedings of the 14th international conference on World Wide Web* (pp. 712-721). New York, NY, USA: ACM.

[7] Bex, G. J., Neven, F., & Bussche, J. V. (2004). DTDs versus XML Schema: A Practical Study. *WebDB '04, Proceedings*

of the 7th International Workshop on the Web and Databases (pp. 79-84). New York, NY, USA: ACM Press.

[8] Bex, G. J., Neven, F., Schwentick, T., & Tuyls, K. (2006). Inference of concise DTDs from XML data. *VLDB '06: Proceedings of the 32nd international conference on Very large data bases* (pp. 115-126). Seoul, Korea: VLDB Endowment.

[9] Chen, B., & Shen, V. Y. (2006). Transforming Web Pages to Become Standard-Compliant through Reverse Engineering. *W4A '06: Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A): Building the mobile web: rediscovering accessibility?* , pp. 14-22.

[10] Chen, S., Hong, D., & Shen, V. (2005). An experimental study on validation problems with existing HTML web pages. *ICOMP'05: Proceedings of the International Conference on Internet Computing*, (pp. 373-379).

[11] Choi, B. (2002). What are real DTDs like? *WebDB '02, Proceedings of the 5th International Workshop on the Web and Databases* (pp. 43-48). Madison, Wisconsin, USA: ACM Press.

[12] Doan, A., & Halevey, A. (2005). *AI Magazine*, Vol. 26, pp. 83-94.

[13] Elmagarmid, A., Ipeirotis, P., & Verykios, V. (2007). Knowledge and Data Engineering, *IEEE Transactions on*, 19 (1), pp. 1-16.

[14] Grijzenhout, S. (2010). *Quality of the XML Web.* Master thesis, University of Amsterdam, Amsterdam, The Netherlands. http://data.politicalmashup.nl/xmlweb/

[15] Guerrini, G., Mesiti, M., & Rossi, D. (2005). Impact of XML schema evolution on valid documents. *WIDM '05: Proceedings of the 7th annual ACM international workshop on Web information and data management* (pp. 39-44). New York, NY, USA: ACM.

[16] Hackett, S., Parmanto, B., & Zeng, X. (2004). Accessibility of Internet websites through time. *SIGACCESS Access. Comput.* , 32-39.

[17] Harold, E. R. (2001). *XML Bible*. New York , NY, USA: John Wiley & Sons, Inc.

[18] Klettke, M., Schneider, L., & Heuer, A. (2002). Metrics for XML Document Collections. *XMLDM Workshop*, (pp. 162-176). Prague, Czech Republic.

[19] Kosek, J., Kratky, M., & Snasel, V. (2003). Struktura realnych XML dokumentu a metody indexovani. *ITAT 2003: Workshop on Information Technologies Applications and Theory*. High Tatras, Slovakia.

[20] Lawrence, S., & Giles, C. L. (2000, Spring). Accessibility of information on the Web. *intelligence* , 11 (1), pp. 32-39.

[21] Lee, D., & Chu, W. W. (2000). Comparative analysis of six XML schema languages. *SIGMOD Rec.* , 29 (3), 76-87.

[22] Liu, B. (2007). *Web Data Mining*. Springer.

[23] Martens, W., Neven, F., & Schwentick, T. (2005). Which XML schemas admit 1-pass preorder typing? *Proceedings of the 10th International Conference on Database Theory* (pp. 68-82). Berlin, Germany: Springer.

[24] Martens, W., Neven, F., Schwentick, T., & Bex, G. J. (2006). Expressiveness and complexity of XML Schema. *ACM Trans. Database Syst.*, 31 (3), 770-813.

[25] McDowell, A., Schmidt, C., & Yue, K.-B. (2004). Analysis and Metrics of XML Schema. *SERP '04, Proceedings of the International Conference on Software Engineering Research and Practice* (pp. 538-544). CSREA Press.

[26] Mignet, L., Barbosa, D., & Veltri, P. (2003). The XML Web: a First Study. *WWW '03, Proceedings of the 12th international conference on World Wide Web*. 2, pp. 500-510. New York, NY, USA: ACM Press.

[27] Mlynkova, I., Toman, K., & Pokorny, J. (2006). *Statistical Analysis of Real XML Data Collections* (Technical Report). Charles University, Faculty of Mathematics and Physics, Department of Software Engineering, Prague, Czech Republic.

[28] Ofuonye, E., Beatty, P., Dick, S., & Miller, J. (2010). Prevalence and classification of web page defects. *Online Information Review* , 34 (1), 160-174.

[29] Pollach, I., Pinterits, A., & Treiblmaier, H. (2006). Environmental Web Sites: An Empirical Investigation of Functionality and Accessibility. *Proceedings of the 39th Hawaii International Conference on System Sciences*. IEEE.

[30] Raghavan, S., & Garcia-Molina, H. (2001). Crawling the Hidden Web. *VLDB '01: Proceedings of the 27th International Conference on Very Large Data Bases* (pp. 129-138). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

[31] Rahm, E., & Do, H.H. (2000). *Data Cleaning: Problems and Current Approaches*, 23 (4)

[32] Sahuguet, A. (2001). Everything You Ever Wanted to Know About DTDs, But Were Afraid to Ask (Extended Abstract). *Selected papers from the 3rd International Workshop WebDB 2000 on The World Wide Web and Databases* (pp. 171-183). London, U.K.: Springer-Verlag.

[33] Sundaresan, N., & Moussa, R. (2001). Algorithms and programming models for efficient representation of XML for Internet applications. *WWW '01: Proceedings of the 10th international conference on World Wide Web* (pp. 366-375). New York, NY, USA: ACM.

[34] Toman, K., & Mlynková, I. (2006). XML Data - The Current State of Affairs. *Proceedings of XML Prague '06 conference*, (pp. 87-102). Prague, Czech Republic.

[35] W3C. (2004, 02 10). *World Wide Web Consortium (W3C)*. Retrieved 04 05, 2010, from RDF - Semantic Web Standards: http://www.w3.org/RDF/

Appendix

Table 1: Structure of the database; the nine main relations with their attributes.

file	header	duplicate_def5
id (*int, key*) URL (*text*) basedomain (*text*) domainextension (*text*) calculatedfilesize (*int*) filenameextension (*text*)	id (*int, key*) author (*text*) cache-control (*text*)	fileid (*int, foreign*) duplicate (*int, foreign*)

encoding	schema_dtd	schema_xsd
id (*int, key*) headerencoding (*enum*) headerencodingchecked (*enum*) pseudoattrencoding (enum) pseudoattrencodingchecked (*enum*) metatagencoding (*enum*) metatagencodingchecked (*enum*)	id (*int, key*) schemaid (*int, foreign key*) fileid (*int, foreign key*) reconstructedurl (*text*) compiles (*enum*)	id (*int, key*) schemaid (*int, foreign key*) fileid (*int, foreign key*) reconstructedurl (*text*) compiles (*enum*)

xmllint_wellformedness_errors	xmllint_validity_errors_dtd	xmllint_validity_errors_xsd
id (*int, key*) fileid (*int, foreign*) linenumber (*int*) errordomain (*enum*) errorlevel (*enum*) errortype (*enum*) specificinformation (*text*) entityline (*text*) element (*text*)	id (*int, key*) schemaid (*int, foreign*) fileid (*int, foreign*) linenumber (*int*) errordomain (*enum*) errorlevel (*enum*) errortype (*enum*) specificinformation (*text*) entityline (*text*) element (*text*)	id (*int, key*) schemaid (*int, foreign*) fileid (*int, foreign*) linenumber (*int*) errordomain (*enum*) errorlevel (*enum*) errortype (*enum*) specificinformation (*text*) entityline (*text*) element (*text*)

Table 2. Top Five DTDs responsible for most errors and warnings in distinct number of documents.

DTD (Reconstructed URL)	Is referenced in distinct number of documents containing validation errors	%	DTD referenced in percentage of documents in total collection (popularity)
	1405	38.3%	41.7%
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd			
http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd	606	16.5%	11.1%
http://www.w3.org/Math/DTD/mathml2/xhtml-math11-f.dtd	181	4.9%	1.3%
http://www.w3.org/TR/MathML2/dtd/xhtml-math11-f.dtd	89	2.4%	< 0.6%
http://www.wapforum.org/DTD/xhtml-mobile10.dtd	64	1.7%	0.6%