From documents to data: linked data at the Dutch Parliament

Marx, M.; Aders, N.

Publication date
2010

Document Version
Final published version

Published in
Online Information 2010: Proceedings. Discover new ways of working in the linked and social web

Link to publication

Citation for published version (APA):
Marx, M., & Aders, N. (2010). From documents to data: linked data at the Dutch Parliament. In *Online Information 2010: Proceedings. Discover new ways of working in the linked and social web* (pp. 17-22). Incisive Media. http://www.online-information.co.uk/online2010/conference/conference_presentation_2010.html?presentation_id=1181

Maarten Marx

Informatics Institute, University of Amsterdam, The Netherlands

Nelleke Aders

House of Representatives of the States General, Department of Information Services, The Hague,
The Netherlands

From documents to data: linked data at The Dutch Parliament

Abstract

Parliamentary debates are important for the general public and for scientific research in numerous fields, such as political science, historical science, linguistics and communication; they are an interesting domain to apply state-of-the-art information retrieval technology.

Parliamentary debates are highly structured transcripts of meetings of politicians in parliament. These debates are an important part of the cultural heritage of countries; they are often free of copyright; citizens often have a legal right to inspect them; and several countries make great effort to digitise their entire historical collection and open that up to the general public.

In this paper, we analyse the structure of the parliamentary proceedings, show how proceedings in PDF format can be transformed into XML and describe the use of permanent identifiers for entities in parliamentary texts. Having the proceedings in XML makes a wide range of applications possible. We elaborate on four of these: entry point retrieval, advanced content and structure search; automatic creation of tables of contents and hyperlinked navigation menus; and large savings on storage space and bandwidth for scanned documents. We also describe the benefits of this approach for the so-called transparency of the parliamentary process for citizens and stakeholders.

1.0 Introduction

Parliamentary proceedings are an interesting set of data with which to apply state-of-the-art information retrieval technology. Parliamentary proceedings are written records of parliamentary activities containing a wide range of document types. In this paper we only discuss notes of meetings of parliament. As with all meeting notes, these records have the purpose of storing the content of the meeting. They come in varying degrees of detail. Currently in most Western democracies it is common to transcribe everything that is being said, keeping the content, but making it grammatically correct and pleasant to read.

We list a number of characteristics which make these documents of special interest to the information retrieval (IR) industry:

- large historical corpora; for example, in Holland all data from 1814 will be available in 2010, at the time of writing it is available since 1974; for the Flemish Parliament all data since 1971 is available in PDF; the British Hansard archives have all parliamentary minutes since 1803 available in XML.
- documents contain a lot of consistently applied structure which is rather easy to extract and make explicit;
- transcripts of meetings might be accompanied by audio and video recordings, creating interconnected multimedia data (Seaton 2005);
- data integration issues and opportunities (Lenzerini, 2002; Halevy et al, 2006; Levy 1996) both within one country (collections from different periods, in different formats, styles, language, ...), and across countries (cross-lingual IR);
- natural corpus for content and structure queries, combining keyword search with XPath navigation and selection (Kamps 2006; O'Keefe & Trotman 2004);
- and, natural corpus for search tasks in which the answers do not consist of documents: "*expert*", or "*people search*" (Balog 2008), video search and "*entry point retrieval*"(Sigurbjörnsson 2003).

From this list, this paper treats the information extraction, data integration and entry-point retrieval aspects. The paper is organised as follows: Section 2 describes the structure of parliamentary meetings. Section 3 describes the techniques used in the transformation process to XML. In section 4, we will discuss the use of permanent identifiers for known entities in parliamentary documents. We discuss four benefits of linked data in Section 5 and conclude in Section 6. Special attention will be given to the application of these results for improvement of the availability and accessibility (transparency) of parliamentary information.

2.0 Structure of parliamentary proceedings

Notes of a formal meeting with an agenda (e.g., business meeting, council meeting, meeting of the members of a club, etc.) are full of implicit structure and contain many common elements. The notes of meetings with a large historical tradition, like parliamentary debates, are in a uniform format which fluctuates little in time. This makes these notes very well suited for text-mining. Transcripts of a meeting contain three main structural elements:

- the topics – discussed in the meeting (the agenda);
- the speeches – made at the meeting, every word that is being said is recorded together with 1) the name of the speaker, 2) her affiliation and 3) in which role or function the person was speaking;
- non verbal content or actions – these can be:
- list of present and absent members;
- description of actions like “applause by members of the Green Party”;
- description of the outcome of a vote;
- the attribution of reference numbers to actions or topics;
- and much more.

The analogy with the structural elements in theatrical drama is striking: scenes, speeches and stage-directions are the theatrical counterparts of the three elements just listed. These are prominent elements in the XML version of Shakespeare’s work (http://metalab.unc.edu/bosak/xml/eg/shaks200.zip). The close relation between politics and drama is an emerging theme in political science (Hariman 1995; Hajer 2005).

These elements are structured as follows:

meeting	==>	(topic)+
topic	==>	(speech | stage-direction)+
speech	==>	(p | stage-direction)+
p	==>	(#PCDATA | stage-direction)*
stage-direction	==>	(#PCDATA).

All elements contain metadata stored in attributes. The British digitised debates from 1803 till 2004 are available in XML (http://www.hansard-archive.parliament.uk/) and basically have this structure.

Within the Dutch proceedings however there is an intermediate structural element – the block – which distinguishes the theatre drama from the political debate. In Dutch Parliament, the debate on each topic is organised as follows: each party may hold a speech by a member standing at the central lectern; other members may interrupt this speech; the chairman can always interrupt everyone. Most often, when all parties have had their say at the central lectern, a member of government answers all raised concerns while speaking from the government table and again he or she can be interrupted. In most cases this concludes a topic, but variations are possible and occur (e.g., several members of government speaking or a second round of the whole process).

The *block* is an important debate structural element because it indicates who is being attacked by the interrupters. Thus for the Dutch situation the DTD becomes:

topic	==>	(block)+
block	==>	(speech | stage-direction)+

If this block structure is not present in meeting notes, then each topic will have exactly one block child. Thus both types of meeting fit this DTD.

Figure 1 contains a visualisation of a one-topic debate which uses the block structure and which is created with an XSL stylesheet from the XML. Each row stands for one block and each vertically positioned mouth stands for one speech. The size of the mouth is proportional to the length of the speech measured in number of words. The speaker on the central lectern has the red mouth, the interrupters have a blue mouth. Interruptions by the chairman are not shown.

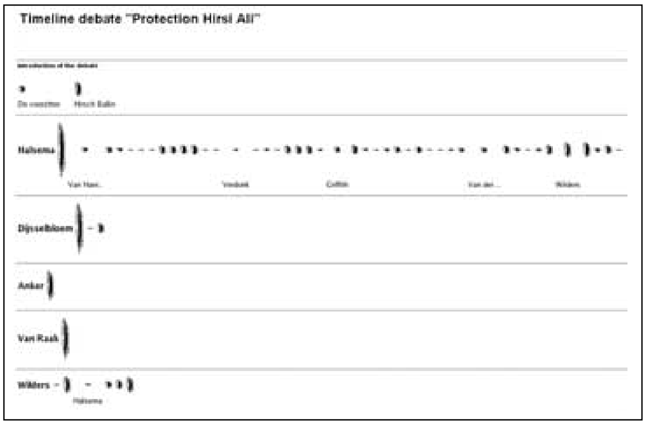
We end this section with two more observations on interesting structure in debates, also visible in Figure 1:

- Blocks consist either of one uninterrupted speech or they have the form (red,blue)+,red, that is a sequence of pairs of speeches by the central speaker and an interrupter ended by the central speaker.
- Zooming in on a block, if A is the speaker at the lectern and B,C,D are the ones interrupting A, then blocks very often look like (AB)+(AC)+(AD)+A, i.e., a sequence of small conversations with different members with A having the last word.

Debates in the Dutch Parliament are governed by a set of written regulations and a set of unwritten codes. Both observations above are instantiations of unwritten codes. The first rule being that the speaker at the lectern always has the last word. The second, that a member of parliament can only have one block of interruptions of a member at the central lectern. Baalen & Bos, 2008 explain these rules. Another rule is that someone may only interrupt another 3 times in a row. So according to these unwritten codes the second regular expression should be (AB){1,3}(AC){1,3}(AD){1,3}A and none of B,C,D should be equal.

Formalisation of these written and unwritten rules in terms of regular expressions, and using these to find violations, is an interesting direction for research.

Figure 1:
High-level visualisation of the first part of the debate on the protection of Hirsi-Ali[1].



The first speaker on the lectern is Halsema who is interrupted by Van Haersma Buma, Verdonk, Griffith, Van der Staai and Wilders, in that order. Onlythe first time a speaker interrupts, her name is shown.

[1] Original available at: http://www.geencommentaar.nl/parlando/index.php?action=doc&filename=HAN8183A16.

3.0 Transformation: from flat PDF to deep XML

Figure 2 (below) gives a good indication of the mappings created in the transformation from PDF to XML. The following technique is used. First we extract the text from the PDF using the open source program pdftohtml[2] with the -xml option. This yields an XML file for each line of text with four coordinates which indicate the bounding box of that text. Multiple columns are detected and preserved. Some font and layout information is preserved but not all. The XML structure is simple and flat:

root	==>	(page)*
page	==>	(text)*
text	==>	(#PCDATA,b,i)*

On these XML files we use patterns written as regular expressions to add special empty XML elements to places where in the final file an XML element needs to be opened. For instance, each little square is replaced by <blockstart/>. A phrase like:

Mevrouw **Swenker** (VVD):

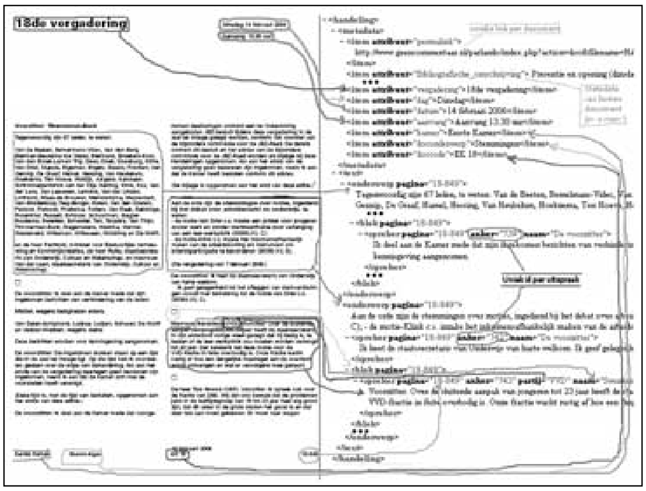
is replaced by

<speechstart speaker='Swenker' party='VVD' .../>,

with the ... containing additional information.

The result of this search and replace process is again a well formed XML file with a similar flat structure as before. In the last step we perform a cascade of groupings starting with the elements which need to be most deeply nested: the paragraphs p. XSLT 2.0 has a very useful command for this task: xsl:for-each-group.

Figure 2:
Example of the mapping from the description of a debate in PDF to the version in XML. Note how the start of a new block is indicated by a 2 (mapping indicated in yellow).



4.0 Permanent Identifiers

In the previous section, entities like people ('Swenker') and political parties ('VVD') are identified by their names only. It will be clear in this context, that this approach has important limitations.

The first two of the principles Tim Berners Lee outlined for Linked Data (2006), are "Use URIs as names for things", and "Use HTTP URIs so that people can look up those names". In short, permanent identifiers (Pids) for (digital) objects are names for objects which:

- are unique (that is, each identifier names exactly one object);
- will forever identify that object;
- are permanently resolvable (that is, there is some function/machinery which, given the name, returns the digital object).

Permanent identifiers ensure that data can be used by others in a robust and sustainable manner. As Marx and Schuth (2010) point out, one of the main difficulties with political data is the lack of permanent identifiers. The need for a persistent, location independent resource identification mechanism for parliamentary data is also emphasised by Fabio Vitali (Griffith et al, 2007).

In the parliamentary context, a Pid is needed for (at least) the following objects:

- published parliamentary documents;
- sub-units in parliamentary documents;
- all named entities in parliamentary proceedings (persons, parties, other organisations, numbers of dossiers);
- controlled vocabulary terms.

For a permanent identifier, three things are needed: a resolver, a namespace, and a good internal practice of giving unique names to objects. The Dutch Parliament already has a unique name giving practice for parliamentary documents and for dossiers, which most probably can form the basis of a system of permanent identifiers. For political entities, like persons, parties and functions, an identification system is available (PDC, Parliamentary Documentation Centre of Leiden University). A Dutch parliamentary namespace does not exist yet, but can be created trivially, and the Royal Dutch Library (KB), together with Data Archiving and Networked Services (DANS) have set up a resolving system.

For a Linked Data approach to Dutch parliamentary data, every named entity that occurs in every parliamentary document has to be made linkable by adding its permanent identifier to the name of the entity inline in the XML text, whereupon linked data are published as RDF using the permanent identifiers in the Linked Open Data Cloud.

Permanent identifiers for parliamentary objects and permanent hyperlinks (permalinks) for each speech made in parliament have many applications. The first is making entry point retrieval possible. Other examples are easy referencing in emails, weblogs and even scientific papers. Permalinks also stimulate third party development of websites (like mashups) based on this data.

[2] http://pdftohtml,sourceforge.net/

5.0 Applications of the XML structure

We describe four applications of the XML structure. None are possible when working with the PDF data. They are: entry point retrieval; complex content and structure queries; automatic creation of tables of contents and navigation menus; and finally savings on bandwidth. We end with a subsection on the benefits for the transparency of the parliamentary process.

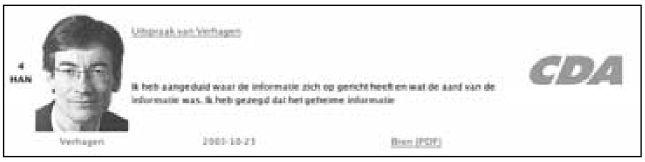
5.1 Entry point retrieval

The most natural answer unit in a retrieval system for parliamentary debates is the speech. The result page after a keyword query then will be a ranked list of items consisting of:

- the name of the speaker,
- her party,
- a photo of the speaker,
- the date of the speech
- a relevant text snippet of the speech,
- a hyperlink which points to the anchor attached to the speech within a debate, and
- a hyperlink to the original PDF source.

This is how it works in the UK on the site http://www.theyworkforyou.com, at the site of the European Parliament, and also in the retrieval engine that we built for the Dutch data http://www.polidocs.nl, see Figure 3.

Figure 3:
Answer snippet from result list: photograph of the speaker linking to his bio, logo of his party, a link to the official PDF source, the first 100 characters of his speech and a link to the speech.



Uitspraak van Verhagen

4
HAN

Ik heb aangeduid waar de informatie zich op gericht heeft en wat de aard van de informatie was. Ik heb gezegd dat het geheime informatie

Verhagen 2001-10-23 Bron (PDF)

Though natural, this notion of answer is by no means standard for parliamentary retrieval systems. The search systems of the German and Flemish parliaments return the proceedings of one day. These can be PDF files with two columns of up to a 100 pages. In the Netherlands, the situation is even more complex:

- proceedings before 1995 are available at http://www.statengeneraaldigitaal.nl/. The answer unit is the proceedings of a complete meeting;
- proceedings after 1995 are available at https://zoek.officielebekendmakingen.nl/zoeken/parlementaire_documenten. The answer unit roughly corresponds to one topic. It is indeed roughly as topics almost never start at the top of a page nor finish at the bottom of a page, and the PDF documents are divided into overlapping sets of pages;
- preliminary (draft) proceedings are available at http://www.tweedekamer.nl/. Search is not really possible on this site. Preliminary proceedings are available in HTML which is shown together with a navigation menu which contains the same topic-block-speech hierarchy as described in Section 2.

During the transformation from PDF to XML we add a unique anchor ID to every speech. This anchor together with the number of the document given by the parliament constitutes a unique permanent reference to each speech.

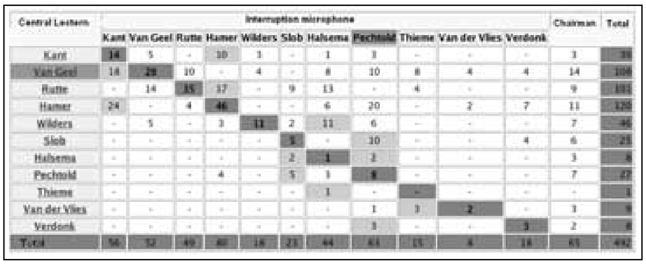
5.2 Complex content and structure queries

The explicit XML structure allows one to formulate information needs using natural XPath, XQuery, XSLT or NEXI (Kamps et al 2006; O'Keefe & Trotman 2004) expressions. We illustrate this by some examples:

- "give speeches about Islam from debates about immigration" can be formulated as the NEXI query //topic[about(.,immigration')]//speech[about(.,'islam')].
- "give all speakers who interrupted Geert Wilders during the Islam debate" can be formulated in XPath 1.0 as //topic[@title='islam']//block[@speaker='Wilders']//@speech[speaker != 'Wilders']/@speaker.
- "give a list of these speakers together with their number of interruptions ordered by that number" is expressed in XQuery or XSLT using the above XPath expression and the fn:count() function.
- "Create a cross table of speakers at the lectern and their interrupters and list the number of interruptions in each data cell" is a typical task for XSLT. The result for the "Algemene Beschouwingen" on September 17 2008, containing 624 speeches in one debate, is reproduced in Figure 4.

Based on experience with Bachelor of Information Science students we claim that it is easier to formulate such complex queries in XSLT directly on the original XML files than to state them in SQL on a relational representation of a debate.

Figure 4:
Who attacks who in the debate Algemene Beschouwingen on September 17 2008.



Central Lectern	Interruption microphone											Chairman	Total
	Kant	Van Geel	Rutte	Hamer	Wilders	Slob	Halsema	Pechtold	Thieme	Van der Vlies	Verdonk		
Kant	14	5	-	10	1	-	1	1	-	-	-	1	33
Van Geel	18	29	10	-	4	-	8	10	8	4	4	14	109
Rutte	-	14	35	17	-	9	13	-	4	-	-	9	101
Hamer	24	-	4	46	-	-	6	20	-	2	7	11	120
Wilders	-	5	-	3	11	2	11	6	-	-	-	7	46
Slob	-	-	-	-	-	5	-	10	-	-	4	6	25
Halsema	-	-	-	-	-	2	1	2	-	-	-	3	8
Pechtold	-	-	-	4	-	5	3	8	-	-	-	7	27
Thieme	-	-	-	-	-	-	1	-	-	-	-	-	1
Van der Vlies	-	-	-	-	-	-	-	1	3	2	-	3	9
Verdonk	-	-	-	-	-	-	-	3	-	-	3	2	8
Total	56	53	49	80	16	23	44	61	15	8	18	63	492

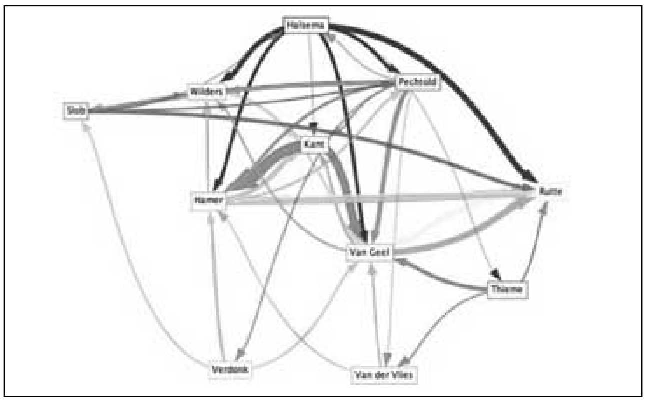
Speakers at the lectern are listed in the first column; their attackers on the top row. The numbers in the cell indicate how often the person on the x-axis interrupted the speech by the person on the y-axis. The numbers on the diagonal (in gray) are the number of answers to interruptions given by the speaker on the lectern[3].

Who is interrupting who can also be visualised in an interruption graph, as in Figure 5. The graph gives a high level summary of the structure of the debate. In the graph, speakers are depicted as nodes, and interruptions are depicted as arrows from the person who is interrupting to

[3] Source: http://staff.science.uva.nl/~marx/politicalmashup/AB2008/DebatstructuurAB2008.html

the person who is speaking. The size of the arrow is representative for the number of interruptions. The graph is constructed using a radial layout, where nodes are placed on concentric circles. The persons in the centre and the innermost circle are the persons at the centre of the debate. The persons on the outermost circles do not participate much in the debate (Kaptein 2010).

Figure 5:
Interruption graph for a topic consisting of 11 blocks



5.3 Automatic creation of tables of contents and navigation menus

The notes of a one day meeting of parliament tend to be quite long, typically between 50 and 100 pages of two columned PDF. Within the current search engine at www.statengeneraaldigitaal.nl these are the documents returned to users. Unfortunately these documents do not contain a table of contents listing the topics discussed in a meeting. But even if such tables were available in PDF they would be of little help when browsing these documents on a computer because they do not contain hyperlinks.

Since the topics are explicit elements in the XML version of the data it is straightforward to automatically generate a hyperlinked table of contents for each document. This can be done with XSLT.

Even one topic can be quite long. For instance, the meeting of September 18, 2008 took the whole day, consisted of 624 speeches with a total of 74,068 words, all within one topic. Fortunately the block structure can be used to break up this large chunk of text. In fact the debate timelines in Figure 1 are navigation menus: each mouth contains a hyperlink to exactly that part of the proceedings which record the speech represented by the mouth. Again this is possible due to the added anchors.

5.4 Savings on bandwidth

The Dutch parliamentary data from before 1995 was only available in printed form. Within the Staten Generaal Digitaal project of the Dutch Royal Library this data is scanned and OCR-ed, resulting in complex PDF documents consisting of facsimile images of every page, the OCR-ed text and a mapping from each word to its position on every page[4].

Such files can be enormous in size. For instance, the proceedings on http://resolver.kb.nl/resolve?urn=sgd:mpeg21:19851986:0000761 are 72 pages PDF. The size of this file is 24 Megabyte. The same proceedings in XML is less than 0.5Mb. We experimented with reducing the size with gzip: the PDF became 23Mb and the XML was reduced to 156Kb. This is 0.65% of the size of the original PDF. For further information on this procedure check out Gielissen & Marx's presentation for AND 2009.

5.5 Transparency of the parliamentary process

With the e-government movement the citizen appears at the centre of information services (Gunter 2006). "Transparency" is a keyword in this context. In the UK, the Government's work on transparency is being lead by the Public Sector Transparency Board (Berners Lee 2009). The website http://www.data.gov.uk/ seeks to be a single, easy to use, access point for public sector data in Britain.

With respect to transparency, parliamentary information is taking a key position. Access to parliamentary information is not only the key to creating interest in the democratic process (Gunter, 2006), it is also a prerequisite for participation. This means that in e-Parliament, the transparency of the parliamentary process is an important issue.

As parliaments function through the medium of documents (Griffith et al 2007), for many years parliamentary information has been disseminated on a document level. The natural information needs of citizens, on the other hand, are concerned with named entities like politicians, parties and ministries, whose data is spread over many documents (Kaptein 2010). It will be clear that the XML- and linked data-based approach described above contributes to the transparency of the parliamentary process, by arriving at a data-centred as opposed to a document-centred dissemination of parliamentary information.

Working on this basis, some parliaments as well as other organisations have already made important steps along the way to more transparency. We will mention some examples: Since 2004, the MySociety website TheyWorkForYou (http://theyworkforyou.com) has provided a searchable, annotatable version of what is said in British Parliament. It aggregates content from the official Hansard record, and other publicly available data. The site aims to provide that information in a clear and concise way that is specific and relevant to the user. It also provides information on a range of different measures of activities by MPs, such as parliamentary appearances and voting patterns.

In the United States, the Sunlight foundation (http://sunlightfoundation.com/) develops and encourages new government policies to make government more open and transparent, by facilitating searchable and machine readable databases and building tools and websites to enable easy access to information. As for the parliamentary websites, Marx et al (2010) made an inventory of best practices of a number of parliaments in publishing parliamentary information. Good examples of parliaments on the way to transparency are in the first place the British Parliament (http://www.parliament.uk). On the website it is possible to browse parliamentary debates, statements and questions by MP, and a timeline for Bills is displayed. Also to be mentioned here are the European Parliament where the activity of MP's, such as statements and questions, is accessible on their homepage (http:// www.europarl.europa.eu/members/public/geoSearch.do?). In Austria (http://www.parlinkom.gv.at), members of parliament and government who are speaking are linked to their biographical page, numbers referring to laws or dossiers are linked to their pages and time-stamps are put in the proceedings at the start of each new speaker.

[4] See http://www.statengeneraaldigitaal.nl/backgrounds.html for extensive information on the digitisation process (in Dutch).

In a project at the Dutch Parliament, we applied the principles described above to develop new ways of presenting parliamentary information, such as dynamic homepages for MP's, including parliamentary activities, areas of interest and affiliations. Also, we created dynamic reports of parliamentary activity, consisting of subjects, questions and answers and most important players in a cabinet term or a period of time. Furthermore, it was shown that, on the basis of a fine-grained, data-centred approach to parliamentary information, there are numerous possibilities for analysis, such as social network analysis for politicians (Suermondt 2010) and the development of a search engine linking politicians to their subjects of interest (Marx & Nusselder (2010).

6.0 Conclusions

We have shown that text extraction from parliamentary proceedings based on regular expressions and XSLT is feasible, scalable, possible on both digital and scanned data, and leads to numerous benefits, including advantages for the transparency of the parliamentary process.

We stress that this extraction process is transparent, repeatable and independent of any software or hardware because we only use declarative programming languages with a well described semantics. This means that when the extraction scripts (which are themselves XML files, since it is XSLT) together with a copy of the XSLT reference (Kay 2002) are stored together with the original digitised data in a safe place, it is in principle always possible to recreate the XML versions we have described here.

Several parliaments are digitising their complete historical data. We are aware of efforts in the UK, Ireland, Australia, and the Flemish Parliament. This opens the possibility of creating a huge integrated multilingual XML repository of parliamentary proceedings. Such a repository will facilitate comparative parliamentary (historical) research.

Acknowledgments

This research was supported by the Netherlands Organisation for Scientific Research (NWO) under project number 380-52-005 (PoliticalMashup).

Bibliography

Baalen, C. van & Bos, A. (2008), In vergadering bijeen. Rituelen, symbolen, tradities en gebruiken in de Tweede Kamer. In *Jaarboek Parlementaire Geschiedenis 2008 : Het feest van de democratie : rituelen, symbolen en tradities*. Amsterdam:Boom.

Balog, K. (2008), *People Search in the Enterprise*. PhD thesis, University of Amsterdam, June 2008.

Berners-Lee, T. (2006), Linked Data (*W3C Design Issues*). Available from: http://www.w3.org/DesignIssues/LinkedData.html

Berners-Lee, T. (2009), Putting government data online (*W3C Design Issues*). Available from: http://www.w3.org/DesignIssues/GovData.

Bex, G.J., Gelade, W., Neven, F., & Vansummeren, S. (2008), Learning deterministic regular expressions for the inference of schemas from XML data. In *Proceedings WWW '08*, pp 825-834.

Gielissen, T. & Marx, M. (2009), Digital Weight Watching: Recreation of scanned documents. In *Proceedings Third Workshop on Analytics for Noisy Unstructured Text Data (AND 2009)*, pp 25-31.

Griffith, J.C., et al. (2007), *ICT in Parliaments: Current Practices, Future Possibilities*: *A discussion paper prepared on the occasion of the World e-Parliament Conference*. 2007, Global Centre for Information and Communication Technologies in Parliament: Geneva, Switzerland.

Gunter, B. (2006), Advances in e-democracy: Overview, *Aslib Proceedings,* 58(5), pp 361-370.

Hajer, M. (2005), Setting the stage, a dramaturgy of policy deliberation. *Administration & Society*, 36 (6), pp 624-647.

Halevy, A., Rajaraman, A. & Ordille, J. (2006), Data integration: The teenage years, *Proceedings VLDB '06*, pp 9-16.

Hariman, R. (1995), *Political style. The artistry of power,* University of Chicago Press.

Kamps, J., Marx, M, Rijke, M. de & Sigurbjörnsson, B. (2006), Articulating information needs in XML query languages, *ACM Trans. Inf. Syst.*, 24 (4), pp 407-436.

Kaptein, R. & Marx, M. (2010), Focused retrieval and result aggregation with political data, *Information Retrieval*, 22. DOI: 10.1007/s10791-010-9130-z.

Kay, M. (2002), *XSLT 2.0 3rd edition Programmer's Reference*. Wrox, 2004.

Lenzerini, M. (2002), Data integration: A theoretical perspective. In *Proc. PODS*, pp 233-246.

Levy, A., Rajaraman, A. & Ordille, J.J. (1996), Querying heterogeneous information sources using source descriptions, *Proceedings VLDB '96*, pp 251-262.

Marx, M., Aders, N. & Schuth, A. (2010), Digital sustainable publication of legacy parliamentary proceedings, *11th Annual International Conference on Digital Government Research* 2010, ACM: Mexico. pp 99-104.

Marx, M. & Nusselder, A. (2010), What you say is who you are. How open government data facilitates profiling politicians, *Proceedings Open Knowledge Conference*. 2010. London.

Marx, M. & Rijke, M. de (2005), Semantic Characterizations of Navigational XPath. *ACM SIGMOD Record*, 34 (2), pp 41-46.

Marx, M. & Schuth, A. (2010), DutchParl: The parliamentary documents in Dutch, *CREC Seventh International Conference on Language Resources and Evaluation*, 2010.

O'Keefe, R.A. & Trotman, A. (2004), The Simplest Query Language That Could Possibly Work, *Proceedings of the 2nd INEX Workshop*, 2004.

Seaton, J. (2005), The Scottish Parliament and e-democracy. *Aslib Proceedings: New Information Perspectives*, 57 (4), pp 333-337.

Sigurbjörnsson, B. (2006), *Focused information access using XML element retrieval*, PhD thesis, University of Amsterdam.

Stockwell, S. (2001), Hacking democracy: the work of the global citizen, *The Sourthern Review - Online Journal,* 34 (3), pp 87-103.

Suermondt, A. (2010), *Social network analysis of political communities*. Available from: http://mashup0.science.uva.nl/PoliticalMashup/SocialNetworks/#copy

Contact

Nelleke Aders
n.aders@tweedekamer.nl