Exemelification of Parliamentary Debates

Tim Gielissen and Maarten Marx
ISLA, University of Amsterdam
Kruislaan 403 1098 SJ Amsterdam, The Netherlands
maartenmarx@uva.nl

ABSTRACT

Parliamentary debates 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 copy-right; citizens often have a legal right to inspect them; and several countries make great effort to digitize their entire historical collection and open that up to the general public. This provides many opportunities for the IR community.

In this paper we analyze the structure of the parliamentary proceedings and sketch a widely applicable DTD. We show how proceedings in PDF format can be transformed into deeply nested XML. We call this process "exemelification". 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; large savings on storage space and bandwidth for scanned documents.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: Miscellaneous; D.2 [**Software**]: Software Engineering

Keywords

XML, Semi Structured Data, Democracy, Information Retrieval, Information Extraction

1. INTRODUCTION

Parliamentary proceedings are an interesting set of data 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 to store the content of the meeting. They have 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 IR community:

- 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 [13];
- data integration issues and opportunities [8, 4, 9] 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 [6, 11];
- natural corpus for search tasks in which the answers do not consist of documents: *expert* or *people search* [1], video search[1] and *entry point retrieval* [14].

From this list, this paper treats the information extraction, data integration and entry-point retrieval aspects. The paper is organized as follows: Section 2 describes the structure of parliamentary meetings and formalizes it in a DTD. Section 3 describes the techniques used in the exemelification process. We discuss four benefits of exemelified data in Section 4 and conclude in Section 5.

A search engine containing all Dutch parliamentary data from 1984 till May 2008 is built and can be used at `http://www.polidocs.nl`. The corpus of over 80.000 XML files is avaliable for research on request.

DIR2009 Enschede

[1] As done in the TRECVID workshop: `http://www-nlpir.nist.gov/projects/trecvid/`

2. 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.

Up to our knowledge there is at the time of writing no DTD or markup language for meeting notes available[2].

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.[3] The close relation between politics and drama is an emerging theme in political science, see e.g., [5, 3].

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 digitized debates from 1803 till 2004 are available in XML[4] and basically have this structure.[5]

Within the Dutch proceedings however there is an intermediate structural element —the block— which distinguishes the theater drama from the political debate. In Dutch parliament, the debate on each topic is organized 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 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.

Note. For presentation purposes, the DTD presented here is the core of the model. The DTD actually used contains additional elements and attributes for storing all kinds of metadata. Up till now, DTD is expressive enough for the structure that we want to capture. But we need the possibility of XML Schema to constrain data-types like dates.

Figure 1 contains a visualization 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:

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.

2. 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 of the rule that the speaker at the lectern always has the last word. The second of the rule that a member of parlement can only have one block of interruptions of a member at the central lectern. See [15] for 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.

[2] The DTD of the XML versions of the British Hansard is effectively just a container to store the text, and not suitable as a genuine model of meeting notes.

[3] `http://metalab.unc.edu/bosak/xml/eg/shaks200.zip` One of the referees pointed out the well-documented DTD for drama which is part of the TEI guidelines for text markup (`http://www.tei-c.org/release/doc/tei-p5-doc/en/html/DR.html`). This DTD is a good starting point for modelling, but for our purposes both too general and too specific.

[4] `http://www.hansard-archive.parliament.uk/`

[5] The structure though is flat instead of nested as it is here, which makes retrieval quite cumbersome. For instance, to retrieve all text spoken by MP `X` we must collect all following siblings of the `member` element which contains the name `X` which come before the next `member` element. We note that this is an example of an until-like query which is not expressible in Core XPath 1.0 [10].

Formalizations of these written and unwritten rules in terms of regular expressions, and using these to find *violations* is an interesting open direction of research.[6]

This internal structure of blocks can be used to create high-level overviews of debates which show who attacks who and which can be used for navigation. We present an example in Section 4.3. The regular expression which best fits or describes a block can be obtained by the algorithm which induces DTD's from a set of example XML-files described in [2].

3. EXEMELIFICATION: FROM FLAT PDF TO DEEP XML

Figure 2 gives a good indication of the mappings created in the exemelification process. The following technique is used. First we extract the text from the PDF using the open source program `pdftohtml`[7] with the `-xml` option. This yields an XML file with for each line of text 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 on places where in the final file an XML element needs to be opened. For instance, the □ 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`. This command, new in XSLT 2.0, replaces the so-called Muenchian method which was needed in version 1.0 of XSLT [7].

4. APPLICATIONS OF THE XML STRUCTURE

We describe four applications of the XML structure. None of these is possible when working with the PDF data. They are entry point retrieval and the use of permalinks, complex content and structure queries, automatic creation of tables of contents and navigation menus and finally savings on bandwidth.

4.1 Entry point retrieval and permalinks

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`, on 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.

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 `http://parlando.sdu.nl/cgi/login/anonymous`. 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 at Parlando are divided into overlapping sets of pages;
- preliminary 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.

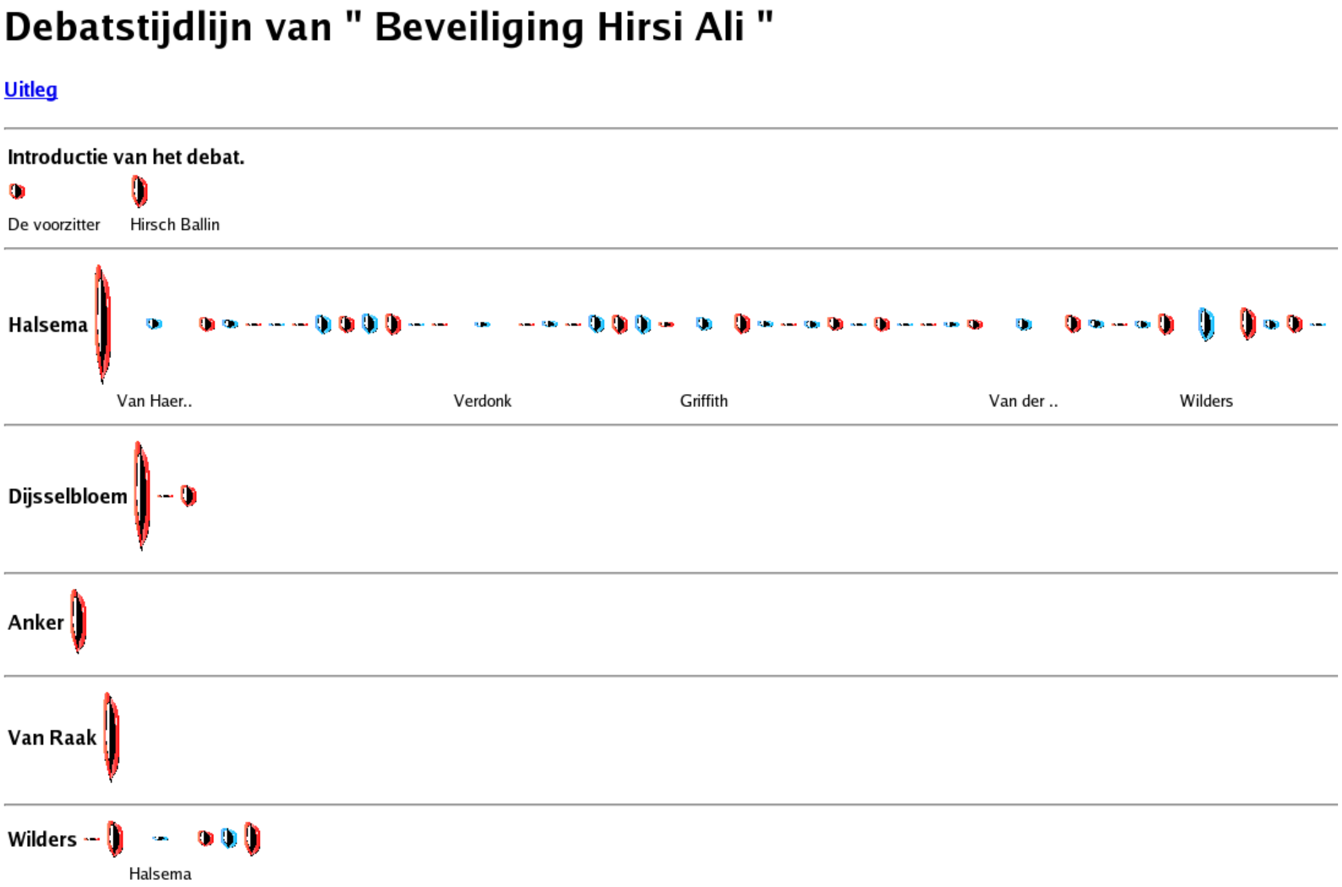
The permanent hyperlinks (permalinks) for each speech made in parliament have many applications besides making entry point retrieval possible. Examples are easy referencing in emails, weblogs and even scientific papers. Permalinks also stimulate third party development of websites (like mashups) based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

4.2 Complex content and structure queries

The explicit XML structure allows one to formulate information needs using natural XPath, XQuery, XSLT or NEXI [6, 11] expressions. We illustrate this by some examples:

[6] We have found such violations with Dutch members of parliament who have a new debating style like Wilders and Verdonk.

[7] `http://pdftohtml.sourceforge.net/`

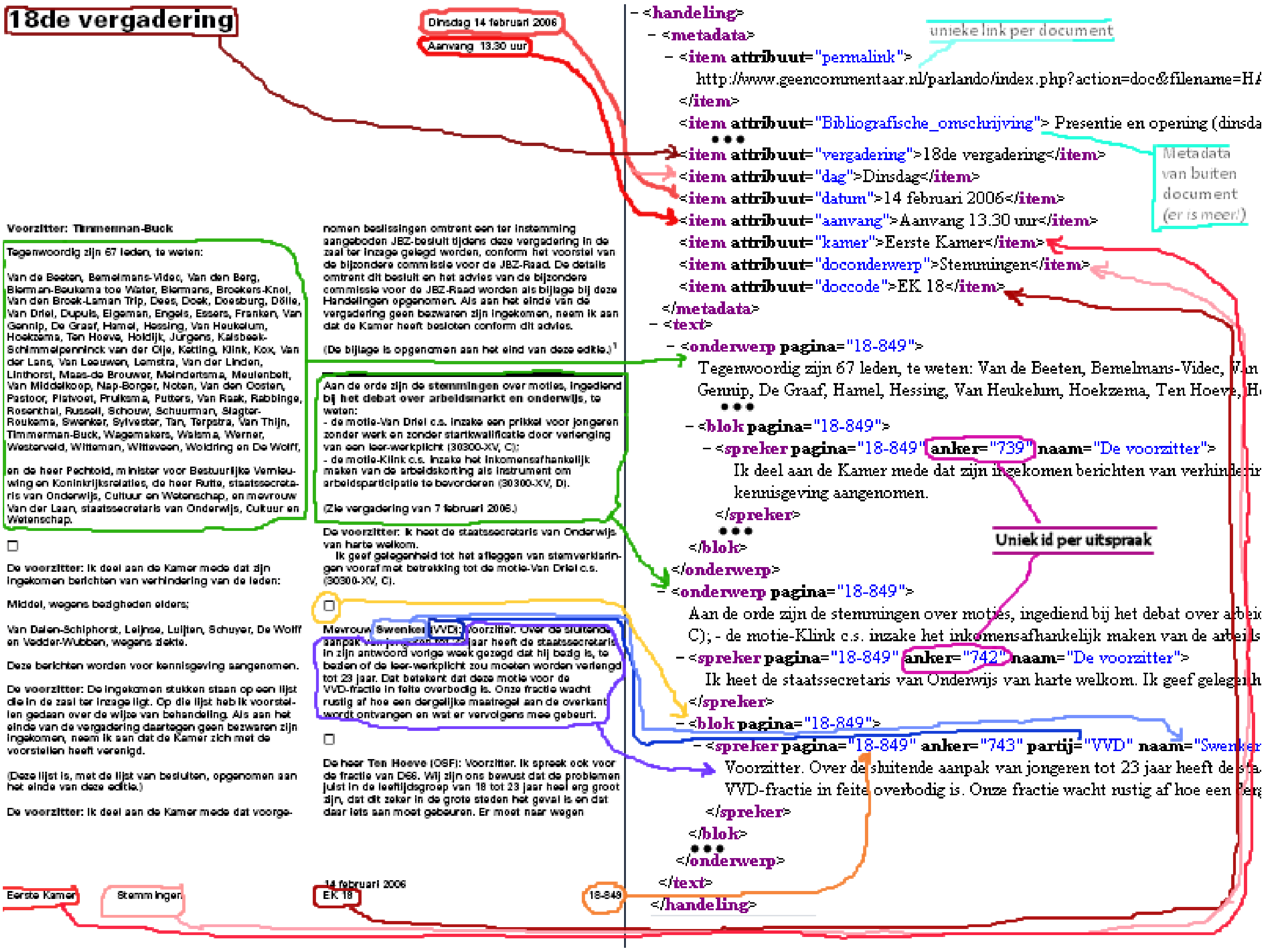


Debatstijdlijn van " Beveiliging Hirsi Ali "

Uitleg

Introductie van het debat.

Figure 1: High-level visualization of the first part of the debate on the protection of Hirsi-Ali. Original available at `http://www.geencommentaar.nl/parlando/index.php?action=doc&filename=HAN8183A16`**. 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. Only the first time a speaker interrupts, her name is shown.**



18de vergadering

Dinsdag 14 februari 2006

Aanvang 13.30 uur

Voorzitter: Timmerman-Buck

Tegenwoordig zijn 67 leden, te weten:

Van de Beeten, Bemelmans-Videc, Van den Berg, Bierman-Beukema toe Water, Biermans, Broekers-Knol, Van den Broek-Laman Trip, Dees, Doek, Doesburg, Dölle, Van Driel, Dupuis, Eigeman, Engels, Essers, Franken, Van Gennip, De Graaf, Hamel, Hessing, Van Heukelum, Hoekzema, Ten Hoeve, Holdijk, Jurgens, Kalsbeek-Schimmelpenninck van der Oije, Ketting, Klink, Kox, Van der Lans, Van Leeuwen, Lemstra, Van der Linden, Linthorst, Maas-de Brouwer, Meindertsma, Meulenbelt, Van Middelkoop, Nap-Borger, Noten, Van den Oosten, Pastoor, Platvoet, Pruiksma, Putters, Van Raak, Rabbinge, Rosenthal, Russell, Schouw, Schuurman, Slagter-Roukema, Swenker, Sylvester, Tan, Terpstra, Van Thijn, Timmerman-Buck, Wagemakers, Walsma, Werner, Westerveld, Witteman, Witteveen, Woldring en De Wolff,

en de heer Pechtold, minister voor Bestuurlijke Vernieuwing en Koninkrijksrelaties, de heer Rutte, staatssecretaris van Onderwijs, Cultuur en Wetenschap, en mevrouw Van der Laan, staatssecretaris van Onderwijs, Cultuur en Wetenschap.

□

De voorzitter: Ik deel aan de Kamer mede dat zijn ingekomen berichten van verhindering van de leden:

Middel, wegens bezigheden elders;

Van Dalen-Schiphorst, Leijnse, Luijten, Schuyer, De Wolff en Vedder-Wubben, wegens ziekte.

Deze berichten worden voor kennisgeving aangenomen.

De voorzitter: De ingekomen stukken staan op een lijst die in de zaal ter inzage ligt. Op die lijst heb ik voorstellen gedaan over de wijze van behandeling. Als aan het einde van de vergadering daartegen geen bezwaren zijn ingekomen, neem ik aan dat de Kamer zich met de voorstellen heeft verenigd.

(Deze lijst is, met de lijst van besluiten, opgenomen aan het einde van deze editie.)

De voorzitter: Ik deel aan de Kamer mede dat voorgenomen beslissingen omtrent een ter instemming aangeboden JBZ-besluit tijdens deze vergadering in de zaal ter inzage gelegd worden, conform het voorstel van de bijzondere commissie voor de JBZ-Raad. De details omtrent dit besluit en het advies van de bijzondere commissie voor de JBZ-Raad worden als bijlage bij deze Handelingen opgenomen. Als aan het einde van de vergadering geen bezwaren zijn ingekomen, neem ik aan dat de Kamer heeft besloten conform dit advies.

(De bijlage is opgenomen aan het eind van deze editie.)[1]

Aan de orde zijn de stemmingen over moties, ingediend bij het debat over arbeidsmarkt en onderwijs, te weten:
- de motie-Van Driel c.s. inzake een prikkel voor jongeren zonder werk en zonder startkwalificatie door verlenging van een leer-werkplicht (30300-XV, C);
- de motie-Klink c.s. inzake het inkomensafhankelijk maken van de arbeidskorting als instrument om arbeidsparticipatie te bevorderen (30300-XV, D).

(Zie vergadering van 7 februari 2006.)

De voorzitter: Ik heet de staatssecretaris van Onderwijs van harte welkom.

Ik geef gelegenheid tot het afleggen van stemverklaringen vooraf met betrekking tot de motie-Van Driel c.s. (30300-XV, C).

□

Mevrouw Swenker (VVD): Voorzitter. Over de sluitende aanpak van jongeren tot 23 jaar heeft de staatssecretaris in zijn antwoord vorige week gezegd dat hij bezig is, te bezien of de leer-werkplicht zou moeten worden verlengd tot 23 jaar. Dat betekent dat deze motie voor de VVD-fractie in feite overbodig is. Onze fractie wacht rustig af hoe een dergelijke maatregel aan de overkant wordt ontvangen en wat er vervolgens mee gebeurt.

□

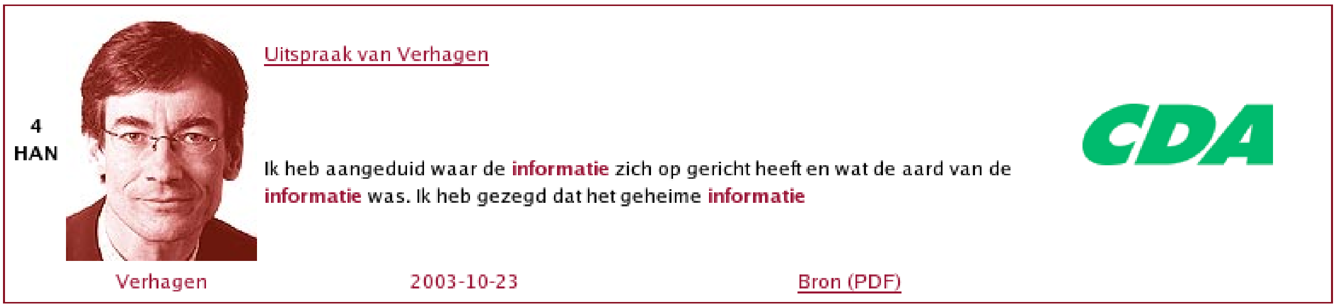
De heer Ten Hoeve (OSF): Voorzitter. Ik spreek ook voor de fractie van D66. Wij zijn ons bewust dat de problemen juist in de leeftijdsgroep van 18 tot 23 jaar heel erg groot zijn, dat dit zeker in de grote steden het geval is en dat daar iets aan moet gebeuren. Er moet naar wegen

Eerste Kamer Stemmingen

14 februari 2006
EK 18 18-849

```
- <handeling>
  - <metadata>
    - <item attribuut="permalink">
        http://www.geencommentaar.nl/parlando/index.php?action=doc&filename=HA
      </item>
      <item attribuut="Bibliografische_omschrijving"> Presentie en opening (dinsda
        •••
      <item attribuut="vergadering">18de vergadering</item>
      <item attribuut="dag">Dinsdag</item>
      <item attribuut="datum">14 februari 2006</item>
      <item attribuut="aanvang">Aanvang 13.30 uur</item>
      <item attribuut="kamer">Eerste Kamer</item>
      <item attribuut="doconderwerp">Stemmingen</item>
      <item attribuut="doccode">EK 18</item>
    </metadata>
  - <text>
    - <onderwerp pagina="18-849">
        Tegenwoordig zijn 67 leden, te weten: Van de Beeten, Bemelmans-Videc, Van
        Gennip, De Graaf, Hamel, Hessing, Van Heukelum, Hoekzema, Ten Hoeve, H
          •••
      - <blok pagina="18-849">
        - <spreker pagina="18-849" anker="739" naam="De voorzitter">
            Ik deel aan de Kamer mede dat zijn ingekomen berichten van verhinderir
            kennisgeving aangenomen.
          </spreker>
          •••
        </blok>
      </onderwerp>
    - <onderwerp pagina="18-849">
        Aan de orde zijn de stemmingen over moties, ingediend bij het debat over arbeic
        C); - de motie-Klink c.s. inzake het inkomensafhankelijk maken van de arbeids
      - <spreker pagina="18-849" anker="742" naam="De voorzitter">
          Ik heet de staatssecretaris van Onderwijs van harte welkom. Ik geef gelegenh
        </spreker>
      - <blok pagina="18-849">
        - <spreker pagina="18-849" anker="743" partij="VVD" naam="Swenker
            Voorzitter. Over de sluitende aanpak van jongeren tot 23 jaar heeft de sta
            VVD-fractie in feite overbodig is. Onze fractie wacht rustig af hoe een derg
          </spreker>
        </blok>
        •••
      </onderwerp>
    </text>
  </handeling>
```

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 □ (mapping indicated in yellow).



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 2003-10-23 Bron (PDF)

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.

- *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 cel* 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 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.

4.3 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 two column 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 would be 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 74068 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.

4.4 Savings on bandwidth

The Dutch parliamentary data from before 1995 was only available in printed form. Within the StatenGeneraalDigitaal 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.[8]

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 .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.

Preliminary experiments show that using XSLT and LaTeX the original format of the proceedings can be produced with very good layout accuracy and very fast. The resulting PDF is again less than .5Mb. Producing this PDF from the gzipped XML can even be done at query time: on a standard Linux box this process took less than 1.5 seconds real time. For detailed information on this experiment see `http://ilps.science.uva.nl/PoliticalMashup/2008/10/trading-space-for-time`.

Thus large savings in bandwidth and storage space become possible. We must note that the XML version is based on OCR-ed data and contains quite a few OCR errors. Of course these come back in the PDF created from the XML source. Repairing such mistakes automatically has been done with promising accuracy by Martin Reyneart using his TICL technique [12].

We believe that the facsimiles need to be available as the ultimate source but that in a search and browse interaction process with the data the alternative, much smaller, version based on the XML is preferable. Users get results faster, they get clean hyperlinked files, and they use much less bandwidth. Once a user knows exactly which document she wants to consult, the large facsimile PDF can be downloaded.

5. 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.

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 [7] are stored together with the original digitized data in a safe place, it is in principle always possible to recreate the XML versions we have described here.

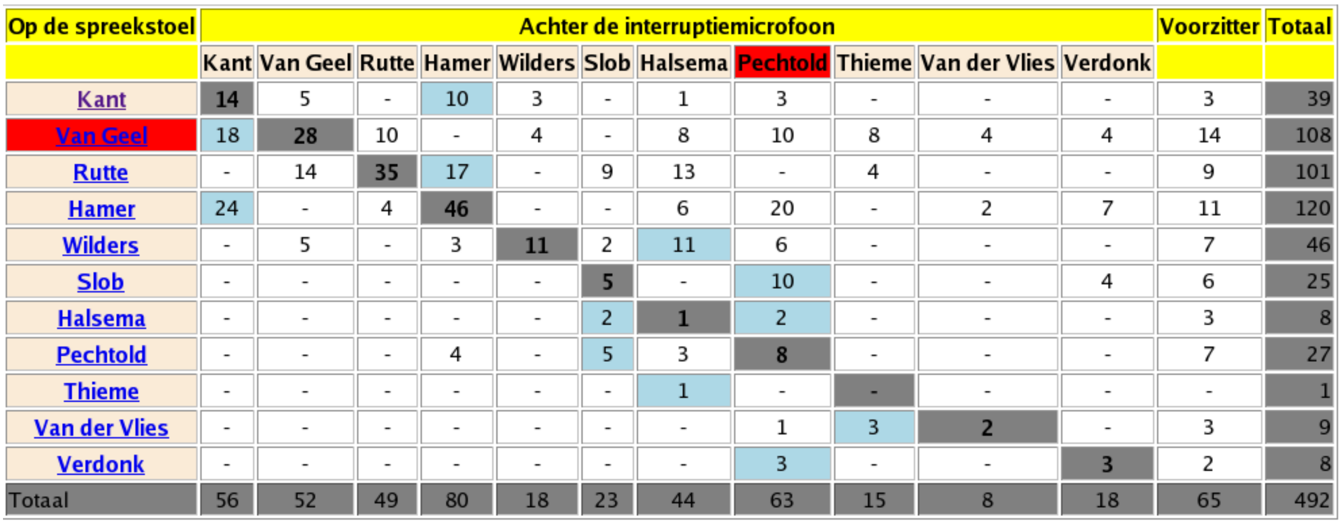
Several parliaments are digitizing their complete historical data. We are aware of efforts in the UK, Ireland, Australia, and the Flemish Parliament. Our DTD is general enough to fit all these proceedings. This opens the possibility of creating a huge integrated multi-lingual XML repository of parliamentary proceedings. Such a repository will facilitate comparative parliamentary (historical) research.

Acknowledgments

This research was supported by the Netherlands Organization for Scientific Research (NWO) under project number 380-52-003.

Many people have helped with creating the `www.polidocs.nl` infrastructure, in particular Robert Jan de Groot, Marina Lacroix, Breyten Ernsting and Marc Makkes. We thank the people from the Dienst Informatievoorziening from the Tweede Kamer der Staten Generaal, in particular Ben van der Beek; those from the StatenGeneraal project at the

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



Op de spreekstoel	Achter de interruptiemicrofoon											Voorzitter	Totaal
	Kant	Van Geel	Rutte	Hamer	Wilders	Slob	Halsema	Pechtold	Thieme	Van der Vlies	Verdonk		
Kant	14	5	-	10	3	-	1	3	-	-	-	3	39
Van Geel	18	28	10	-	4	-	8	10	8	4	4	14	108
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
Totaal	56	52	49	80	18	23	44	63	15	8	18	65	492

Figure 4: Who attacks who in the debate *Algemene Beschouwingen* on September 17 2008. Speakers at the lectern are listed in the first column; their attackers on the top row. The numbers in the cel 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. Source: `http://staff.science.uva.nl/~marx/politicalmashup/AB2008/DebatstructuurAB2008.html`.

Koninklijke Bibliotheek, Huibert Crijns and Tineke Koster, and Hans Nielen from PDC.

6. REFERENCES

[1] K. Balog. *People Search in the Enterprise*. PhD thesis, University of Amsterdam, June 2008.

[2] G. J. Bex, W. Gelade, F. Neven, and S. Vansummeren. Learning deterministic regular expressions for the inference of schemas from xml data. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 825–834, New York, NY, USA, 2008. ACM.

[3] M. Hajer. Setting the stage, a dramaturgy of policy deliberation. *Administration & Society*, 36(6):624–647, 2005.

[4] A. Y. Halevy, A. Rajaraman, and J. J. Ordille. Data integration: The teenage years. In U. Dayal, K.-Y. Whang, D. B. Lomet, G. Alonso, G. M. Lohman, M. L. Kersten, S. K. Cha, and Y.-K. Kim, editors, *VLDB*, pages 9–16. ACM, 2006.

[5] R. Hariman. *Political style. The artistry of power.* University of Chicago Press, 1995.

[6] J. Kamps, M. Marx, M. de Rijke, and B. Sigurbjörnsson. Articulating information needs in XML query languages. *ACM Trans. Inf. Syst.*, 24(4):407–436, 2006.

[7] M. Kay. *XSLT 2.0 3rd edition Programmer's Reference*. Wrox, 2004.

[8] M. Lenzerini. Data integration: A theoretical perspective. In *Proc. PODS*, pages 233–246, 2002.

[9] A. Y. Levy, A. Rajaraman, and J. J. Ordille. Querying heterogeneous information sources using source descriptions. In T. M. Vijayaraman, A. P. Buchmann, C. Mohan, and N. L. Sarda, editors, *VLDB*, pages 251–262. Morgan Kaufmann, 1996.

[10] M. Marx and M. de Rijke. Semantic Characterizations of Navigational XPath. *ACM SIGMOD Record*, 34(2):41–46, 2005.

[11] R. A. O'Keefe and A. Trotman. The Simplest Query Language That Could Possibly Work. In *Proceedings of the 2nd INEX Workshop*, 2004.

[12] M. Reynaert. Non-interactive OCR post-correction for giga-scale digitization projects. In *Proceedings. CICLing (Computational Linguistics and Intelligent Text Processing, 9th International Conference)*, pages 617–630, 2008.

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

[14] B. Sigurbjörnsson. *Focused information access using XML element retrieval*. PhD thesis, University of Amsterdam, 2006.

[15] C. van Baalen and A. Bos. In vergadering bijeen. Rituelen, symbolen, tradties en gebruiken in de Tweede Kamer. In *Jaarboek Parlementaire Geschiedenis 2008*. Boom, 2008.