Processamento Estruturado de Documentos

António Soares
E-mail: a21697@correio.ci.uminho.pt

Nuno Martins
E-mail: a21707@correio.ci.uminho.pt

Paulo Correia
E-mail: a22707@correio.ci.uminho.pt

Dezembro 2000

Resumo

Informação é Poder! Esta afirmação tem cada vez mais sentido nos dias de hoje, em que toda a informação é armazenada, quer em bases de dados quer em ficheiros, no entanto, em sectores mais tradicionais, a informação ainda se encontra em papel, complicando a consulta e gestão de informação contida nesses documentos. Torna-se necessário converter este tipo de suporte menos eficaz num outro mais potente: o digital.

Neste projecto, iremos converter um documento do Arquivo Distrital de Braga, para formato electrónico, mas não nos limitaremos a fazer um simples *OCR*, para além deste passo, também serão tratados e gerados os diversos índices de forma automática, de modo a facilitar a pesquisa, indexação e gestão desta informação e outra do mesmo tipo.

Ao longo deste relatório de projecto irão ser indicados e explicados todos os passos tomados desde a versão em suporte físico do documento original, até à correspondente versão digital.

Um dos objectivos principais é que a conversão seja o mais automática e independente de intervenção humana possível, já que só deste modo é que toda a estrutura de conversão de documentos, poderá vir a ser realmente utilizada no mundo real.

Capítulo 1

Contexto e Análise do Problema

A consulta de informação de documentos em suporte físico (papel) é muito penosa e pouco versátil, uma vez que é pouco flexível, não é nada recomendável no que toca a pesquisas (porque a indexação é estática e é muito difícil de adaptar a novos requisitos). Inevitavelmente a conversão de documentos neste tipo de suporte para suporte digital teria de ser efectuada, uma vez que a digitalização da informação facilita imenso a manutenção, pesquisa e gestão da informação contida.

Para instituições em que o volume de informação é muito grande, caso do Arquivo Distrital de Braga (ADB), este tipo de conversão é muito útil e urgente, já que os seus documentos são sujeitos a muitas e diversos tipos de pesquisas, e se esta conversão for automatizada melhor ainda, já que a conversão de documentos é mais rápida e sujeita a menos erros, já que por arrasto, a intervenção humana é também menor.

também importante referir que, no caso do ADB, é importante garantir que o comum dos mortais não tem acesso a documentos antigos. , pois necessário oferecer meios de acesso, consulta e pesquisa desses documentos sem perigrar os documentos a serem consultados.

O documento que nos foi incumbido de converter é um excelente exemplo de documento em que a informação é muito difícil de pesquisar e consultar e, em muitos casos, se encontra repetida ou referenciada de modo errado, o que o torna um candidato excelente para a digitalização e anotação.

A simples digitalização de documentos não é significativa, nem traz grandes

vantagens, quer aos utilizadores, quer aos arquivos. necessário levar a tarefa mais longe e aproveitar para enriquecer o documento com informação que melhore a pesquisa e estrutura do mesmo. aqui que entra o XML, SGML e outras linguagens de anotação e ferramentas associadas, que permitem enriquecer os documentos com informação extra que irá facilitar a geração de índices, pesquisa, e outras *vistas* do mesmo documento, potenciando assim a consulta do mesmo.

O projecto foi realizado de uma forma faseada, cada fase sujeita a problemas e implicações diferentes, cada uma requerendo abordagens teóricas e práticas diversas.

As secções seguintes do relatório vão tentar reproduzir a sequência temporal da execução do projecto. em cada uma delas irá ser apresentado o problema a abordar, qual a base teórica e/ou prática, bem como o modo de o ultrapassar. No final teremos um documento em formato digital, que se que espera seja mais fácil de consultar, organizar e pesquisar do que o original.

Capítulo 2

Análise do documento e construção do DTD

A primeira fase do projecto foi constituída pela análise atenta e cuidada do documento original e pela construção de um DTD para o mesmo.

A análise do documento original **Cartas Anuais das Missões da Etiópia** permitiu identificar a constituição do documento, todos os seus elementos, a sua constituição e relacionamento, enfim, o modo geral como foi escrito e que nos deu uma ideia do que era relevante ou superflúo no seu conteúdo e quais os elementos que deveríamos olhar com mais atenção.

O DTD é, no fundo, um *esquema* representativo da estrutura do documento. Nele são identificados os elementos constituintes do mesmo (por um nome), bem como a listagem de caracteríscas a ele associadas (ou atributos). São estes os elementos que irão dar uma estrutura ao documento, e que vão permitir o seu manuseamento e tratamento.

A escolha destes elementos *básicos* foi a tarefa fundamental desta fase, com estes elementos construímos o DTD, que é a peça central dos futuros documentos digitais, que serão anotados (todos os elementos claramente identificados com as tags e atributos respectivos) de acordo com o DTD definido nesta fase.

Capítulo 3

OCR do documento e geração de XML

Depois de estudada e identificada a estrutura do documento, passamos à digitalização do mesmo, para tal fizemos o *scanning* das páginas do documento original e o resultado foi PASSADO a uma aplicação de OCR e que gerou um documento no formato RTF, primeira versão digitalizada d'**As Cartas Anuais**, no entanto este documento era muito pouco útil já que a sua formatação era deficiente e não trazia nada de novo ao utilizador, pelo que a sua alteração foi o passo seguinte.

A primeira fase de anotação do documento foi conseguida com uma ferramenta denominada `rtf2xml` cuja função é transformar um ficheiro em formato *RTF* num documento *XML*. Após uma observação exaustiva do xml gerado, verificamos facilmente que a sua anotação era muito pobre e com elementos superflúos, que em nada contribuem para o enriquecimento estrutural do documento, mas foi a primeira etapa para a completa anotação do documento, que foi feita ao longo das próximas duas fases, relatadas nas secções 4 e 5.

Para o ficheiro gerado ser mais facilmente trabalhado, foi necessário converter a codificação dos caracteres acentuados, para o formato **ISO-8859-1**. Para tal utilizamos uma pequena script escrita em Perl.

Exemplo:

Script de conversão de formato - livro.pl

```
#!/usr/bin/perl
```

```
use XML::DT ;
my $filename = shift;

%handler=(
    '-outputenc' => 'ISO-8859-1',
    '-default'   => sub{toxml;},
);
print dt($filename,%handler);
```

Capítulo 4

Melhoramento XML

Como foi referido no capítulo anterior o ficheiro XML obtido era muito pobre e, consequentemente pouco útil, pelo que foi necessário alterá-lo, para que obedecesse *minimamente* ao DTD especificado, e para concretizar do que fazer uma stylesheet em XSL (`xml2xml.xsl`), já que é uma linguagem muito poderosa e ideal para conversão e tratamento de ficheiros XML. O funcionamento desta script é muito simples: o xml que foi obtido a partir do RTF, foi percorrido à procura dos elementos principais do documento, a introdução, conteúdo e bibliografia. Depois de localizados estes elementos, todos os parágrafos situados dentro deles, foram convertidos para os *verdadeiros* elementos constituintes de cada uma destas peças do livro, de acordo com o DTD, ou seja, documentos, bibitems, etc, e foi a partir deste ponto que o documento xml já começou a adquirir a *forma* do documento xml final esperado.

Como a primeira versão do xml era muito pobre, não foi possível reconhecer elementos mais específicos do que os principais, pelo que a anotação mais aprofundada teve que ser feita posteriormente. Neste ponto, a anotação resume-se à identificação dos elementos mais importantes (de topo), mas foi uma etapa essencial, porque poupou trabalho na anotação (em detrimento do trabalho que custou a escrever as scripts), e que facilitou a visualização geral da estrutura do documento, permitindo mais facilmente identificar os sub-elementos consituintes do documento.

Capítulo 5

Anotação do documento

O DTD é o esqueleto para um documento completo e estruturado. Se quisermos que o documento seja o mais completo possível, torna-se necessário anotar todos os elementos que seja possível identificar, mesmo os mais simples. Até aqui só se encontram anotados, de acordo com o DTD, os elementos mais importantes, pelo que é necessário anotar, **manualmente**, os elementos restantes.

Este processo de anotação teve de ser manual uma vez que até aqui não foi possível reconhecer os elementos mais básicos do DTD de uma forma automática, em que são exemplos os elementos `nome` e `local`. Esta tarefa foi uma das mais árduas, já que o documento foi percorrido manualmente, e anotados todos os elementos que até aqui ainda não o tinham sido. Após isto, todo o documento respeitava integralmente o DTD especificado, estando completamente anotado, ou seja todas as informações que se podem influir do documento estão claramente identificadas.

Para testar a integridade do documento, isto é, se toda a anotação respeita o DTD do documento, utilizamos uma script implementada em **OmniMark**, gentilmente cedida pelo Professor Ramalho, cujo conteúdo se lista de seguida:

Exemplo:

Script de verificação

```
;------------------------------------------------------------
; parser.xom por jcr em 06.11.2000
;------------------------------------------------------------

down-translate with xml
```

```
element #implied
  suppress
```

Esta script simplesmente vai percorrer o documento xml, e tendo em conta o DTD, adoptado para ele, vai verificar quer o encadeamento das tags, quer os seus atributos. Também verifica se a sintaxe é a correcta, ou seja, se todas as tags abertas têm a sua correspondente tag de fecho. No final da verificação indica que houve erros, se detectar alguma anomalia na constituição do documento, ou não refere nada se tudo estiver correcto.

Esta fase do projecto, foi muito importante, já que foi aqui que se deu toda a estrutura semântica ao documento, uma anotação bem feita é meio caminho andado para obter bons resultados com o documento em causa, pelo que foi necessária atenção redobrada ao detectar e identificar todos os elementos relevantes do coumento. Neste ponto era necessário anotar correctamente todas as tags de identificação e classificação de documentos, nomes, locais, etc., para que ao ser tratado, o documento XML produzissse os melhores resultados: não perder a informação que os historiadores identificaram como sendo importante, não colocar informação extra que seja superflúa e anotar tudo de modo a conseguir identificar ao máximo toda e qualquer informação que seja importante para qualquer utilizador e que lhe facilite a consulta e pesquisa deste e doutros documentos dentro do mesmo estilo.

Capítulo 6

Scripts de tratamento do XML e geração do HTML final

Depois de correctamente anotado, ficamos com um documento extremamente rico em informações extra que lhe prestaram outra semântica e muito mais sentido, mas um documento XML anotado, por si só, não é muito relevante nem útil. pois necessário trata-lo, extrair a informação importante e formata-la de modo atraente e de fácil manuseio para o utlizador final.

Mais uma vez, recorremos ao XSL, já que permite a conversão facilitada do documento XML, e criamos um conjunto de scripts, que percorrem o documento anotado e geram um conjunto de páginas HTML, em que a informação é disposta de forma amigável, e que permite uma fácil consulta e *navegação* de todo o documento, possibilidades que os documentos originais dificilmente, ou mesmo nunca, oferecem.

Para a obtenção de um bom resultado final, o trabalho foi dividido em várias partes, cada uma das quais implementada por uma script xsl. Para a conversão do xml gerado pela script rtf2xml para um xml melhor organizado, contendo apenas informação importante, foi implementada a script `xml2xml.xsl`.

Como já foi dito anteriormente, a script gerou um documento, que posteriormente foi anotado de acordo com o que nos pareceu necessário para uma boa identificação de todos os elementos contidos no documento. Esta anotação foi feita manualmente, pois através de uma script xsl, seria impossível anotar o documento de uma forma tão exaustiva.

Numa segunda fase, foi implementada uma outra script xsl, cujo principal objectivo era o de pegar no documento previamente anotado, e transformá-la num ficheiro HTML. Esta script foi denominada pelo grupo por `xml2html.xsl`.

Nesta script foram transformadas algumas secções do documento, tais como, a bibliografia, a introdução e as cartas. Esta divisão no tratamento das várias seções em várias scripts deve-se únicamente ao cuidado que tivemos com a apresentação do resultado final. Para tal, os índices, a capa e o sumário foram tratadas por scripts diferentes.

Para fazer a transformação de cada uma destas secções, foram identificados os elementos que as constituem, de acordo com o dtd do documento, e construído um template diferente para cada um dos elementos. No template do elemento raíz foi indicado que as transformações deveriam ser aplicadas a todos os filhos, através da função `xsl:apply-templates`.

Os templates para cada um dos elementos das secções envolvem, principalmente, as funções xsl `xsl:for-each` e `xsl:if`, para aplicar a cada um dos elementos que entram na template.

As scripts que geraram os índices têm a particularidade de ordenar os seus elementos. Na nossa opinião deve ser dada uma breve explicação sobre a orgânica que esteve por detrás da implementação das scripts que deram origem aos índices toponímico e onomástico.

Para cada um destes índices, foi implementada uma primeira script que apenas ordena todos os elementos, gerando um documento XML intermédio. Este documento vai ser tratado, em seguida, por uma segunda script, cujo objectivo é o de retirar as entradas repetidas geradas pela primeira script.

Capítulo 7

O duo HTML e CSS

Chegados a esta fase, já só nos restava *embelezar* o HTML gerado pelas nossas *scripts*. Encontrámo-nos, então numa encruzilhada: ou o HTML gerado continha a formatação gráfica ou, esta ficaria independente do HTML gerado. Obviamente esta última opção foi a eleita, uma vez que é muito mais fléxivel e permite ao comum dos mortais *editar* as cores e todas as demais formatações do HTML, sem entrar nos meandros das nossas scripts.

Foi assim que decidimos criar três CSS's. Uma para a capa, onde estão definidas todas as características gráficas dos elementos que compõem a capa. Outra CSS definida foi para ser utilizada pelos índices, para que todos os índices tivessem a mesma aparência. Por último a CSS principal que é utilizada pelas cartas propriamente ditas. Nesta, estão definidas as opções que podem ser tomadas para a definição de praticamente todos os elementos que compõem as cartas, como sejam as datas, locais, instituições, números romanos, etc.

Exemplo:

CSS de formatação da capa

```
BODY{
background-color:#ffffdf;
}

.universidade{
    color:navy;
    text-align:center;
}
```

```
.adb{
    color:navy;
    text-align:center;
}

.titulo{
    color:#123456;
    text-align:center;
}

.localidade{
    color: #FEDCAB;
    text-align:center;
}

.data_capa{
    color:#6789AB;
    text-align:center;
}
```

Vemos assim que a utilização de CSS, só nos trouxe grandes vantagens, pois permite-nos gerar um HTML livre de formatação (e consequentemente mais *limpo*) e permite ao utilizador flexibilizar a escolha gráfica do site sem ter que percorrer centenas de linhas de código que desconhece.

Capítulo 8

Glossário

- **ADB**
  Arquivo Distrital de Braga.
- **CSS**
  Cascading Style Sheet
- **DTD**
  Document Type Definition
- **HTML**
  HyperText Markup Language
- **OCR**
  Optical Character Recognition
- **RTF**
  Rich Text Format
- **XML**
  eXtensible Markup Language
- **XSL**
  eXtensible Stylesheet Language

Apêndice A

Agradecimentos

Agradecemos a colaboração e apoio do Professor José Carlos Ramalho, que esteve sempre disponível para esclarecer as nossas dúvidas e dificuldades.

Também uma palavra de apreço e agradecimento às nossas namoradas, que toleraram, melhor ou pior, as nossas noitadas em frente ao computador, e a nossa má disposição quando as coisas não corriam como o esperado!

Bibliografia

[1] "Anotação Estrutural de Documentos e sua Semântica", José Carlos Ramalho, Departamento de Informática, Universidade do Minho, 2000.

[2] "Manuais on-line: XSLT" (http://www.zvon.org)

[3] "The XML Companion", Neil Bradley, Addison-Wesley, 1998

Conteúdo