

Introdução às linguagens de marcas

Marcello Peixoto Bax

bax@eb.ufmg.br

www.bax.com.br

Escola de Ciência da Informação

Universidade Federal de Minas Gerais

Resumo

*Apresenta-se o paradigma de gerenciamento da informação que surgiu com o padrão das linguagens ditas “de marcação” (ou **markup languages**). Faz-se uma rápida introdução à linguagem SGML, e analisam-se as características e os diferenciais que estão por trás do sucesso da linguagem XML, que promete uma revolução na Web. Mostram-se quais são as bases conceituais de uma nova geração de aplicações das áreas da Informação e da tecnologia da informação que democratizarão ainda mais o acesso à informação organizada na Internet. Aborda-se a evolução das pesquisas em direção à chamada Web Semântica, com o desenvolvimento de ontologias.*

Palavras-chave

Markup languages; Linguagens de marcas; Internet; HTML; XML; SGML.

Introduction to markup languages

Abstract

This paper presents the information management paradigm which came up with the so-called markup languages. A short introduction to SGML is presented and the characteristics and differentials behind the success of XML language – which promises a revolution on Web – are analyzed. There are also shown the conceptual bases of a new generation of applications of Information and Information Technologies which will democratize even more the access to organized information on the Internet. With the development of ontologies, the evolution of research towards the so-called Semantic Web is mentioned.

Keywords

Markup languages; Internet; HTML; XML; SGML.

INTRODUÇÃO

Procura-se discutir neste documento sobre o paradigma de gerenciamento – organização, recuperação e uso – da informação, que surgiu com o padrão das linguagens ditas “de marcação” ou “de marcas” (do termo inglês *markup languages*). Faz-se uma rápida introdução à linguagem SGML, e, em seguida, analisam-se os diferenciais que estão por trás do sucesso da linguagem XML, que fazem desta última responsável por uma verdadeira revolução na Web. Procura-se mostrar quais são as bases de uma nova geração de aplicações que serão lançadas na área da informação na rede Internet e em intranets.

No início da era dos computadores, há 40 ou 50 anos, estes eram usados, sobretudo, para processar dados e fazer cálculos. O nível de abstração na interação com as máquinas era baixo demais para a maioria das pessoas. Sua utilização era quase que restrita a laboratórios de tecnologia. Há hoje uma mudança radical deste cenário. Após a surpreendente evolução da microinformática nas últimas duas décadas, que elevou sobremaneira o nível de abstração da relação homem/máquina, constata-se hoje que a maior contribuição visível dos computadores está em sua utilização como ferramentas de armazenamento, organização, recuperação e intercâmbio de informações entre pessoas, empresas e até entre programas. Enfim, hoje o computador é visto pela sociedade, cada vez mais, como uma ferramenta de **comunicação**, e não propriamente de **cálculo**.

A informação e o computador são parceiros antigos, mas a intensificação e democratização do seu uso, aliadas à abstração sempre crescente do nível de interação e troca de informações, criaram terreno propício para a origem das chamadas linguagens de marcação. Este fenômeno foi marcante na primeira década dos anos 90, com o aparecimento da Web. Estas linguagens permitem a construção de padrões públicos e abertos que estão sendo criados para se tentarem maiores avanços no tratamento da informação; elas minimizam o problema de transferência de um formato de representação para outro e liberam a informação das tecnologias de informação proprietárias.

Estas linguagens identificam, de forma descritiva, cada “entidade informacional” digna de significado presente nos documentos, como, por exemplo, parágrafos, títulos,

tabelas ou gráficos. A partir destas descrições, os programas de computador podem melhor compreender e, em consequência, melhor tratar ou processar a informação contida em documentos eletrônicos.

Este artigo é organizado como segue: na primeira seção, explicam-se as vantagens no uso das linguagens de marcação como um novo paradigma e como elas são aplicadas com vistas a um melhor gerenciamento da informação. Em seguida, na seção 3, apresentam-se os princípios da linguagem *Standard Generalized Markup Language* (SGML), base desse paradigma, e uma tentativa de padronizarem-se os diversos formatos empregados para representar a informação. Na seção 4, apresenta-se uma recente descendente de SGML, XML, uma versão simplificada de SGML especialmente voltada para as necessidades da Web.

LIBERANDO O PODER DA INFORMAÇÃO

Marcação procedimental versus marcação descritiva

Todo sistema editor ou, mais genericamente, “processador” de textos tem de embutir, juntamente com o texto editado, marcas que fornecem indicações de como este texto deve ser apresentado ao usuário. As marcas podem estar escondidas do usuário, como geralmente é o caso nos editores do tipo *WYSIWYG*¹, ou devem ser explicitadas pelo usuário, que obterá seu documento no formato visual desejado somente após uma compilação do texto (Tex e Latex são exemplos conhecidos desse tipo de processador de textos, principalmente usado no mundo acadêmico).

Marcas inseridas no documento de forma implícita (pelo programa, em resposta a determinado comando do usuário) ou explícita (pelo usuário) indicam **como** o processador deve dispor o texto na página; qual fonte de caracteres usar e muitas outras características tipográficas. Estas marcas ou códigos são tipicamente específicos a um sistema de formatação proprietário. Cada *software* editor ou compilador de textos possui seu próprio conjunto de códigos com significado apenas para aquele sistema, que deverá rodar em um determinado sistema operacional ou em uma máquina específica. Diz-se destes sistemas que eles promovem uma **marcação procedimental do texto**, cada código indicando o procedimento a ser seguido para a apresentação do texto ao usuário [Marchal 2000, p. 14]. A figura 1 ilustra um exemplo de marcação procedimental.

FIGURA 1
Exemplo de Marcação Procedimental

```
(\f1\fs48 Three Kinds of Markup\par
\pard \nowidctlpar \widctlpar \adjustright
(\f1\par (\prtext\pard\plain\fs28\cgrid \loch\af1\loch\af0\loch\af1
1.\tab))\pard \fi-360\i360\nowidctlpar \widctlpar \clisttab\tx360
(\*pn \pr\nibody\ivl0\fs1 \pr\not\pndec\pnstart1 \pnindent360\pnhang(\prbda
.))\s1\adjustright (\f1\fs28 Presentational\par (\prtext\pard\plain\fs28
\cgrid \loch\af1\loch\af0\loch\af1 2.\tab))\pard \fi-360 \i360
\nowidctlpar \widctlpar \clisttab\tx360 (\*pn
\pr\nibody\ivl0\fs1 \pr\not\pndec\pnstart1 \pnindent360 \pnhang(\prbda
.))\s1\adjustright (\f1\fs28 Procedural\par (\prtext\pard\plain\fs28\cgrid
\loch\af1\loch\af0 \loch\af1 3.\tab))\pard \fi-360\i360\nowidctlpar \widctlpar
\clisttab\tx360(\*pn \pr\nibody\ivl0\fs1 \pr\not\pndec \pnstart1 \pnindent360
\pnhang(\prbda .))\s1\adjustright (\f1\fs28 Descriptive)\f1\fs28 \par )
```

FIGURA 2
Exemplo de Marcação Descritiva

```
<slide>
<title>Dois tipos de Marcação</title>
<point>Marcação Procedimental</point>
<point>Marcação Descritiva</point>
</slide>
```

Paralelamente à corrente de tratamento de informação descrita anteriormente, existe uma outra, a das chamadas linguagens baseadas em marcação descritiva. Estas linguagens usam marcas (ou *tags*) para qualificar cada objeto do texto, cumprindo um primeiro passo para transformá-los em informação tratável por computador. Uma marca ou *tag*² é tudo o que não for considerado conteúdo em um documento. Elas indicam a função (o propósito) da informação no documento, em vez de como ela deve ser apresentada, ou seja, sua aparência física. A idéia básica é a de que o conteúdo do documento deve estar separado do estilo usado em sua apresentação. Cabe, à aplicação que interpreta a linguagem de marcação, formatar o texto em tempo real e apresentá-lo aos usuários. No passado, a marcação procedimental era utilizada pelo usuário final (troff/TEX); hoje, nos programas de processamento de textos modernos, a marcação procedimental é escondida do usuário, porém ela ainda está lá, presente.

Separando conteúdo, estrutura e estilo

Do ponto de vista das linguagens de marcação, considera-se todo documento como constituído de três componentes, claramente distintos e separados: (a) conteúdo, (b) estrutura e (c) estilo (ou formatação). O conteúdo é a informação propriamente dita, a estrutura define como se dá a organização da informação, ou das idéias, no documento, e o estilo define o visual de apresentação das informações ao usuário.

Tal distinção ou separação promove, ou acaba se revertendo em uma simplificação, pois o autor não tem mais que se preocupar *a priori* com o “visual” da informação, podendo dedicar-se exclusivamente ao conteúdo e à estrutura de apresentação das idéias no documento. Dessa forma, o texto se manterá bem “mais limpo”, sem uma infinidade de códigos que não dizem respeito ao conteúdo da informação, podendo ser mais facilmente compreendido pelo homem.

A utilização de padrões de marcação internacionais abertos (SGML, HTML, XHTML, XML, etc.) permite assim a criação de documentos portáteis, isto é, documentos que não são dependentes de um determinado *software*, *hardware*, ou sistema operacional. Documentos que contêm apenas texto ASCII (ao contrário de formatos binários) e que podem ser interpretados por aplicações presentes nos mais diversos ambientes computacionais, bastando que exista uma aplicação no ambiente que reconheça o padrão usado na criação do documento. Como são padrões abertos, a informação não fica aprisionada, pode-se desenvolver conversores de um padrão para outro. Geralmente os *softwares* de interpretação e conversão são de domínio público e gratuitos.

Dessa forma, as linguagens de marcação libertam a informação da “prisão” dos formatos proprietários. Além disso, permitem múltiplas apresentações do documento, de forma totalmente independente da mídia de veiculação, monitores, celulares, impressora, interpretador braile, televisão etc. A aplicação que deve tratar a informação é que se encarrega de interpretar as marcas e processá-las, para efeitos de estilo, ou outros processamentos.

Marcas ou tags descritivos

Nas linguagens de marcação, marcas descritivas definem o início e o fim do texto marcado como unidade ou elemento de informação. Por exemplo: <par>Isto é um parágrafo</par>. Pode-se também embutir elementos dentro de outros, por exemplo:

```
<topico>  
<par>Isto é um parágrafo</par>  
</topico>
```

Assim esse paradigma permite tratar cada unidade de informação como um objeto (ou entidade) ao qual se pode atribuir características específicas, o que possibilita **maior estruturação da informação**. De um monte de caracteres estáticos, dispostos em uma página à espera de uma interpretação humana (o computador está longe de entender texto livre), a informação passa a poder ser interpretada e tratada automaticamente por computador. Os dados se transformam em objetos qualificados com atributos. Tem-se então a possibilidade de reutilização automatizada da informação; pode-se mais facilmente compartilhá-la com outros usuários; organizá-la em bancos de dados e realizar pesquisas automáticas.

Imagina-se um sistema de pedidos de compras funcionando pela Internet. As informações constantes dos documentos transmitidos entre fornecedores e clientes precisam ser bem estruturadas. A entidade “cliente” poderia ter sua estrutura definida como no documento apresentado na figura 1.

Neste documento, o *tag* CLIENTE tem como atributos o nome do cliente e sua identificação. Ordens de compras (OCs) fazem parte da definição de cada cliente, assim o *tag* OC define com seu atributo id (identificação de cada OC) quais são as ordens de compras efetuadas pelo cliente. Por sua vez, as OCs são compostas de itens com o *tag* ITEMOC, definindo cada item da ordem de compra. A principal representante desta corrente de linguagens é SGML.

OS PRINCÍPIOS DE SGML

Standard Generalized Markup Language, ou simplesmente SGML, é uma (meta)linguagem criada há aproximadamente 30 anos como um esforço para se definir uma linguagem de marcas para a representação de informações em texto (Edwards, 1997). A linguagem foi reconhecida como um padrão ISO (8879) em 1986. SGML não é um conjunto predeterminado de marcas, e sim uma linguagem para se definirem quaisquer conjuntos de marcas, uma linguagem *autodescritiva*; cada documento SGML carrega consigo sua própria especificação formal, o *Data Type Document* (DTD), apresentado mais adiante.

O DTD é uma espécie de gramática formal criada a partir da notação *Extended Backus-Naur Form* (EBNF), que define como as marcas devem ser interpretadas, quais as regras que restringem o uso de cada marca nos diferentes contextos do documento e, até mesmo, quando relevante for, a ordem em que as marcas devem aparecer no documento. Resumindo, SGML é uma linguagem para definir outras linguagens, ou ainda uma linguagem para conceber DTDs, tipos de documentos.

A origem e evolução de HTML

No início dos anos 80, SGML passou a ser usada em várias organizações, entre as quais o CERN, Centro Europeu de Pesquisas Nucleares em Genebra, onde um pesquisador resolveu empregar a linguagem em seu programa de edição de hipertextos (Connolly *et al.*, 1997). Assim, Tim Berners-Lee acabou inventado o *World-Wide Web*, graças a uma idéia revolucionária na época: o *link* (ou ligação) entre documentos que poderiam estar situados em qualquer lugar na rede de computadores de seu laboratório ou do mundo, através da Internet e do conceito de *Universal Resource Locator* (URL).

HTML é um exemplo de linguagem originada de SGML. Ou seja, a definição formal (ou especificação, ou ainda o DTD) de HTML é construída em SGML. HTML possui um grupo de *tags* predefinidos, concebidos com a função de organizar a informação a ser transferida por meio de páginas Web.

HTML é uma linguagem extremamente popular hoje. Segundo Benoît Marchal (Marchal, 1999), alguns estudos atestam a existência (no ano de 1999) de mais de 800 milhões de páginas na Web, todas baseadas em HTML. HTML é um padrão usado em milhares de aplicações, incluindo navegadores, editores, *softwares* de *e-mail*, servidores de base de dados e outros.

HTML e a “guerra dos *browsers*”

No início dos anos 90, nos seus primeiros anos de vida de 1992 a 1995, quando a Web literalmente “explodiu” no mundo todo, muitas organizações e empresas começaram a perceber que estavam bastante limitadas pela falta de flexibilidade de HTML no tocante às suas possibilidades em promover a troca mais efetiva de informações pela Web. HTML foi então estendida posteriormente a cada nova versão, de forma um tanto desorganizada, impulsionada pela conhecida guerra dos navegadores (ou *browsers*). E o que foi pior: estas extensões integraram principalmente elementos puramente de apresentação

FIGURA 3
Descrição de uma ordem de compra

```
<cliente nome="Paulo de Tal" id="1398">
<OC id="00150">
  <ITEMOC id="1234">
    <Descricao> Geladeira Frostfree Modelo 1234 </Descricao>
    <Preco>550,00</Preco>
  </ITEMOC>
  <ITEMOC id="1235">
    <Descricao> Fogão Especial 4 Bocas </Descricao>
    <Preco>450,00</Preco>
  </ITEMOC>
</OC>
</cliente>
```

(formatação, estilo), que controlam a aparência das informações nos navegadores. Como visto anteriormente, isso vai de encontro ao paradigma das linguagens de marcação descritiva, no sentido em que estas procuram separar a estrutura e a semântica da informação de sua apresentação física (estilo). A introdução da formatação de estilo em HTML começou a tornar os documentos de difícil leitura para o homem. Além disso, devido ao número de novos *tags* e de novos atributos de estilo nos *tags* que já existiam, a tarefa de formatação dos documentos HTML tornou-se extremamente entediante, exatamente como em processadores do tipo Word da Microsoft.

Tentando fazer o papel de árbitro nesta guerra, o *World-Wide Consortium* (W3C) definiu versões mínimas que deveriam ser interpretadas por todos os navegadores. O W3C é a organização que se encarrega do desenvolvimento e manutenção dos padrões da Web (para mais informações, visite www.w3c.org). Em uma de suas últimas publicação sobre HTML (a versão 4.0), o W3C incentiva a separação entre a estrutura e o visual dos documentos HTML, aspecto fundamental do paradigma, e desenvolveu as chamadas “folhas de estilo” ou CSS (*Cascading Style Sheet*), que definem *como* os elementos devem ser mostrados nos navegadores.

Os “Data Type Documents” ou DTDs

A estrutura de um documento em uma aplicação SGML é definida pelos chamados *Data Type Document* (DTD's). Cada DTD é uma espécie de gramática que dita as regras para a verificação da correção do documento. O DTD define os tipos dos elementos³ (capítulos, título de capítulo, cabeçalho de seção, parágrafo etc.) que constituem a estrutura do documento, assim como o relacionamento que existe entre estes elementos. Por exemplo, a marca que indica o título de cada capítulo precisa existir e, além disso, vir sempre antes da marca que define o capítulo.

Um DTD acompanha o documento para onde ele for e pode ser usado para validá-lo, verificando que o conteúdo está bem formado de acordo com as regras do DTD. Uma parte do DTD do documento apresentado na figura 1 seria o seguinte:

```
<!ELEMENT cliente - - (ITEMOC)+>  
<!ATTLIST cliente nome CDATA id CDATA>
```

Uma lista não numerada em HTML, por exemplo, definida pela marca UL do inglês “Unordered List”, é especificada como tendo um *tag* de início e um de fim (note os dois caracteres “-”), e contendo ao menos um item (definido pelo *tag* LI de “List Item”). Sua especificação em SGML seria:

```
<!ELEMENT UL - - (LI)+>
```

O sinal + após os parênteses indica, segundo a norma EBNF, que o seu conteúdo (LI) deve estar presente pelo menos uma vez no interior do *tag* UL.

Contrariamente à SGML, que é um padrão complexo e difícil de implementar, a grande vantagem de HTML é sua relativa facilidade em ser entendida pelo usuário da Web e de ser processada, mesmo em diferentes navegadores. Este aspecto foi o principal responsável pela explosão da Web. Paradoxalmente, a falta de flexibilidade acabou se revelando uma força da linguagem e seu fator popularizador.

Agora que a Web e tecnologias afins estão relativamente maduras, as empresas estão procurando formas de introduzir maior flexibilidade em seus documentos (como suas páginas Web), para aumentar o potencial de troca de informações, visando ao comércio eletrônico, por exemplo. Entra em cena um novo padrão, a linguagem XML.

O QUE É O PADRÃO XML?

A linguagem *Extensible Markup Language* (XML) é o resultado do trabalho de um grupo de especialistas estabelecido em 1996 pelo W3C, com o objetivo de propor uma simplificação de SGML que fosse voltada às necessidades específicas da Web (Bryan, 1998).

“XML ... often referred to as containing 20% of the complexity and 80% of the functionality of SGML.”
(Edwards, 1997)

XML é similar a HTML em vários aspectos, também é uma linguagem expressa em arquivos de texto puro (ASCII), concebida especialmente para armazenar e transmitir dados. Como uma representante do paradigma das linguagens de marcação, trata-se de texto com marcas embutidas que qualificam cada unidade de informação (também referidas como entidades, elementos, ou objetos) contida no texto.

Assim, um arquivo XML é constituído de elementos. Como sempre, cada elemento possui uma marca inicial (como <titulo> ou <title>), uma marca final (como </titulo> ou </title>) e a informação propriamente dita entre as duas marcas.

Porém, diferentemente de HTML, XML não propõe um número fixo de marcas. Um elemento XML pode ser marcado da forma que o autor do documento bem entender, ou seja, com o termo que melhor descreve a informação na sua opinião.

Por exemplo, um preço seria representado pelo *tag* <preco>, um número de pedido por <numpedido>, um nome por <nome> etc. Cabe ao autor do documento determinar que tipo de dado usar e quais marcas os representam melhor.

As diversas entidades de informação contidas em um documento XML (definidas pelas marcas) são interpretadas por aplicações (um navegador Web, por exemplo) e organizadas em um modelo de objetos onde permanecem acessíveis à aplicação. A aplicação pode assim ativar ações sobre as entidades de informação. A figura 2 apresenta um exemplo de documento em XML.

Também como já foi visto, ao invés de descrever como os dados devem ser *mostrados*, as marcas indicam o que cada dado *significa*. Qualquer agente (humano ou de *software*) que receba este documento pode decodificá-lo e usar os dados como lhe convier. Por exemplo, uma livraria poderia usar estes dados (figura 2) para preencher uma ordem de compra; um analista de mercado para descobrir quais livros são mais populares; um indivíduo poderia armazená-lo em um banco de dados como um registro de suas compras etc.

Como acontece em HTML, em XML as marcas podem ser embutidas umas dentro de outras. Geralmente usa-se isso para determinar uma informação com significado mais específico dentro do texto. Por exemplo:

```
<titulo>  
<compositor>Bach</compositor>'s First Piano  
Concerto  
</titulo>
```


NOTAS

¹ WYSIWYG significa “What You See Is What You Get”. O editor Word® da Microsoft é um exemplo desse tipo de processador de textos.

² No restante do texto, utiliza-se a palavra *tag* ou a palavra “marca” como sinônimos.

³ Um elemento é diferente de um *tag*. Quando nos referimos a elementos, estamos considerando os *tags* de abertura e finalização juntamente com o conteúdo de informação entre os *tags*.

⁴ No sistema de datas anglo-saxão, o mês vem antes do dia.

REFERÊNCIAS BIBLIOGRÁFICAS

BRYAN, M. *Guidelines for using XML for electronic data interchange*. [S. l.] : SGML Centre, 1998. Disponível em: <www.geocities.com/WallStreet/Floor/5815/guide.htm>

BRYAN, M. *An introduction to the Extensible Markup Language (XML)*. [S. l.]: SGML Centre, 1997. Disponível em: < www.sgml.u-net.com/xmlintro.htm>

Connolly, Dan; Khare, Rohit; Rifkin, Adam. The evolution of web documents: the ascent of XML. *World Wide Web Journal* v. 2, n.4, p. 119-128, 1997. Disponível em: <www.cs.caltech.edu/~adam/papers/xml/ascent-of-xml.html>

Edwards, M. XML: data the way you want it. [S. l.] : Microsoft, 1997.

Marchal, B. XML by example. *QUE*, 2000.

Powers, Shelley. XML expectations. [S. l. : s. ed.], 1998. Disponível em: <www.ne-dev.com/ned-01-1998/ned-01-xml.t.html>

XML: a technical perspective. [S. l.] : Microsoft, 1998.