第 35 卷 第 9 期 计 算 机 工 程 2009 年 5 月
Vol.35 No.9 Computer Engineering May 2009
·开发研究与设计技术· 文章编号：1000—3428(2009)09—0258—03 文献标识码：A 中图分类号：TP311

# 嵌入式 HTML 文档解析器的设计与实现

李庆诚，彭 洁，宫晓利，刘嘉欣
(南开大学信息技术科学学院计算机科学与技术系，天津 300071)

**摘 要：**针对 HTML 文档在手持移动阅读设备上的阅读有效性问题，设计实现一种面向嵌入式应用的、平台无关的 HTML 文档解析器，对其关键技术进行阐述和分析。提出一种屏幕适配探测机制，实现对当前屏幕阅读无效内容的过滤。实验结果表明，该解析器降低了对嵌入式系统处理能力与内存配置的要求，能满足手持阅读设备的需要。

**关键词：**嵌入式；HTML 文档；解析器；屏幕适配探测；手持阅读设备

# Design and Implementation of Embedded HTML Document Parser

LI Qing-cheng, PENG Jie, GONG Xiao-li, LIU Jia-xin
(Dept. of Computer Science and Technology, College of Information Technology and Science, Nankai University, Tianjin 300071)

**【Abstract】**Concerning the effective reading of HTML document on handheld mobile devices, design and implementation of an embedded application-oriented, platform-independent HTML document parser, and its key technologies are described and analyzed. It presents Screen Matching Detection(SMD) mechanism to filter void content of current screen reading. Experimental results show that the parser reduces the requirements of the embedded system processing capabilities and memory configuration, and meets the needs of handheld reading device.

**【Key words】**embedded; HTML document; parser; Screen Matching Detection(SMD); handheld mobile device

## 1 概述

随着互联网的高速发展和各种数字技术的不断进步，信息数字化浪潮席卷全球。Web 文档作为应用最为广泛的互联网信息载体，大多数都采用 HTML(HyperText Markup Language)语言[1]书写而成。消费电子、计算机、通信一体化趋势日趋明显，嵌入式技术成为研究热点，其中手持阅读设备的快速发展，使得 HTML 格式的文档在这些设备上的阅读有效性问题日益显现。

在 HTML 文档解析领域，已经有大量的研究工作，其中最具普遍性的设计思想是采用对象树模型(Document Object Model, DOM)[2]来表示 HTML 文件的内部结构。在对象树模型中，网页中的标签按嵌套关系被整理成一棵树状结构，诸如字体大小、颜色等属性均被保存在每个节点中。DOM 对象树模型作为组织 Web 页面内部数据的标准形式，被用于当前流行的浏览器中。但在嵌入式应用中，采用 DOM 方法对整个 HTML 文件进行完整解析后再排版、显示，对系统动态内存空间的要求过高，有可能在解析环节由于内存不足造成系统崩溃，因此，该方法并不完全适用于嵌入式应用。

本文提出一种新的 HTML 文档解析器设计方法——屏幕适配探测(Screen Matching Detection, SMD)法。该方法打破了 DOM 模型中必须先进行完整解析后再排版显示的旧有思路，将排版环节确立为解析工作的控制中心，对目标显示区域外的内容进行过滤，实现了有针对性的解析。

## 2 HTML文档解析器设计概要

### 2.1 HTML解析的一般考虑

HTML 解析的过程是将 HTML 文档的流式数据结构化的过程[3]，它并不以发现和纠正 HTML 文档的语法错误为目的，而是要尽量忽略遇到的语法错误，最大程度地解析出 HTML 文档中合法有效的成分。针对面向屏幕显示的嵌入式应用，解析器的主要工作是将构成 HTML 文档的基本元素(称为显示对象，包括文字、直线、图片等)提取出来，并进行合理的组织排版，最终显示在屏幕上。因此，对于 HTML 文档中可能包含的一些服务器端脚本(如 ASP 和 PHP 脚本)，以及不同的用户代理对于标准 HTML 的扩展，解析器均予以忽略。

### 2.2 解析架构

本解析器由词法语法分析模块、排版控制模块、屏幕显示模块、多页面缓冲模块 4 部分组成，层次结构如图 1 所示。

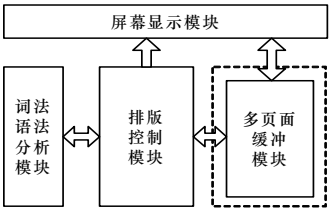


**图 1 HTML 解析器解析架构**

词法语法分析模块按照 HTML 语言规范识别标签以及标签之间的内容，每个标签都有对应的处理函数，由处理函数生成显示对象。词法语法分析模块是解析工作的基础，被排版控制模块反复调用。

排版控制模块在解析器系统架构中处于中心的地位。它

**基金项目：**天津市科技发展计划基金资助项目(06YFGZGX04000)
**作者简介：**李庆诚(1964－)，男，教授、博士、博士生导师，主研方向：电子阅读平台，数字版权保护，嵌入式数据库，Java 中间件；彭 洁，硕士研究生；宫晓利，博士研究生；刘嘉欣，工程师、硕士
**收稿日期：**2008-09-29 **E-mail：**nkpengjie@163.com

负责调用词法语法分析模块生成显示对象，并对显示对象进行排版，将其中有效的显示对象送入屏幕显示模块。排版控制模块协调解析器的运行，这是实现 SMD 方法的关键。

屏幕显示模块实现了与具体系统相关的屏幕显示接口调用。屏幕显示模块与其他模块相互独立，针对不同的 GUI，只须相应改变 GUI 的 API 函数，而无须改变内部的解析调用机制，实现了解析过程与显示过程相分离，使解析器能够快速移植到不同的 GUI 系统下。

多页面缓冲模块是一个可裁减的功能模块[4]，负责管理页面位图缓冲区，缓冲区中存放着与当前屏幕请求页面最邻近的页面位图，当用户改变当前屏幕请求页面(如通过翻页操作)时，多页面缓冲模块将更新缓冲区中的页面位图，总保持对用户下一操作的高命中率。

### 2.3 解析器的工作流程

解析器的工作流程如下:

(1)解析器将 HTML 文件读入内存(对于较大的文件采用分块读入的方法)，调用词法语法分析模块对文件字符流进行分析。分析的具体过程包括：从字符流中识别出标签并进入相应的标签处理函数，在处理函数中修改属性状态和生成显示对象。词法语法分析的过程就是一个边提取标签边分析整合，最终生成显示对象的过程。

(2)排版控制模块接收到显示对象后，根据当前屏幕游标的位置，计算出该显示对象在版面上的绝对坐标，从而判断出该显示对象是否落入目标显示区域内，所有落入目标显示区域的显示对象均为有效显示对象。

(3)对于有效显示对象，排版控制模块将调用屏幕显示模块，将显示对象的内容(如文字、直线、图片)填入目标显示区域，随后调用词法语法分析模块继续解析；而对于无效显示对象，排版控制模块将对目标显示区域进行探测，当发现目标显示区域已填满，则停止解析并转去屏幕显示。

## 3 关键问题及解决方案

### 3.1 表格的处理

表格承担着设计排版 Web 文档的重要功能，表格的处理一直是 HTML 解析工作的重点[5]。表格结构可视作一个封闭的块，由若干个小的单元块组成。解析器将表格视作一个独立对象，图 2 是系统中定义的表格对象的数据结构。

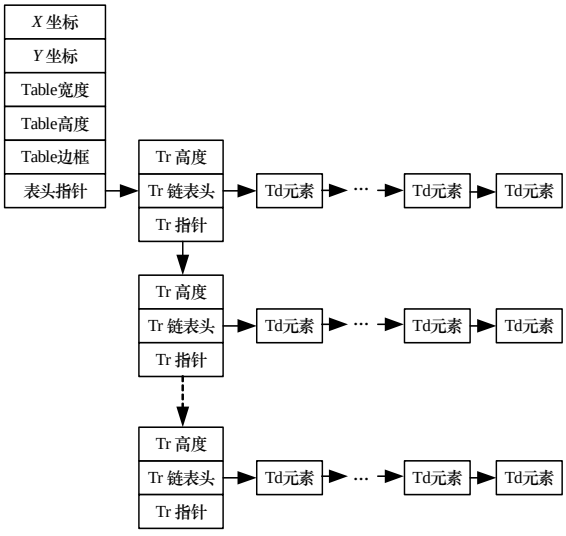


**图 2 表格对象的数据结构**

当遇到表格对象，词法语法分析模块将对表格起始标签<table>和表格结束标签</table>之间的内容进行 2 次扫描分析。第 1 次扫描利用 TD, TR 等标签的属性获取表格结构，并且记录每一个单元格的宽度、高度和单元格内包含的文字。第 2 次扫描对取得的结构进行分析和定位，利用 HTML 语言中关于内层标签默认继承外层标签属性的规定，对属性信息不完整的单元格进行信息补全。经过 2 次扫描，解析器得到了清晰完整的表格结构，并交由排版控制模块处理。

### 3.2 属性状态栈的使用

HTML 语言中规定，系统的显示属性(如字体大小、颜色、对齐方式等)具有可继承性和可还原性[6]，可继承性是指如果当前层标签没有显式地注明属性值，则它将默认继承外层标签的属性；可还原性是指如果当前层标签修改了属性值，则修改后的属性值只在当前层标签起始的范围内有效，遇到匹配的结束标签时要将系统属性值还原到修改之前的状态。

针对必须对属性值进行嵌套继承、还原这些特点，解析器使用栈这种数据结构来记录属性信息，并为栈元素定制了如图 3 所示的数据结构。

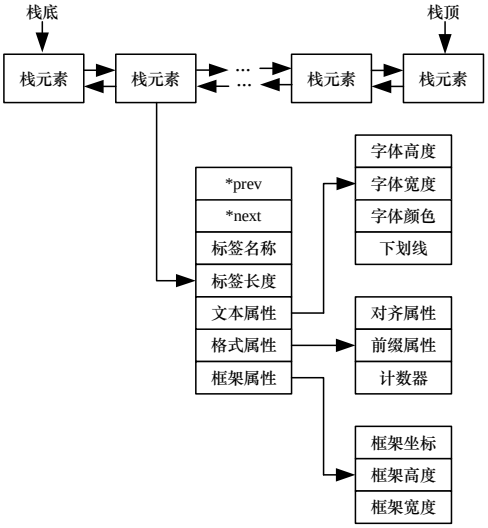


**图 3 属性状态栈的数据结构**

栈顶元素中总保存着当前系统正在处理的显示对象的属性值。当解析遇到标签引发系统属性值发生改变时(如<font>标签改变了当前字符的颜色)，解析器将对栈顶元素进行复制，修改相应属性值后压入栈中，成为新的栈顶元素，这样既完成了当前系统属性值的改变，又保存了修改前的属性值；当属性状态需要还原时(如遇到对应的</font>标签)，只须将新的栈顶元素出栈即可。

### 3.3 屏幕适配探测方法的实现

手持阅读设备的屏幕显示需要大量浮点运算，这是时间消耗的主要原因，因此，为了高效地显示目标页面，解析器需要对目标页面之外的显示对象进行过滤。定义 HTML 文档的全部页面为样本空间，其中目标页面所占的区域为屏幕空间。屏幕匹配探测方法通过计算显示对象在样本空间中的坐标，判断其是否落入屏幕空间，对落入屏幕空间之外的显示对象进行过滤，并在适当的时候对屏幕空间是否已填满进行探测。实现屏幕适配探测方法的关键是排版控制模块，屏幕适配探测的流程如图 4 所示。

在词法语法分析模块生成一个显示对象后，解析器并不立即调用显示，而是先将该显示对象送入排版控制模块，排版控制模块利用显示对象附带的属性信息(如对齐属性)，计

算出该显示对象在样本空间中的坐标，从而可以判断出该显示对象是否落入屏幕空间，决定是否将该显示对象送入屏幕显示模块。

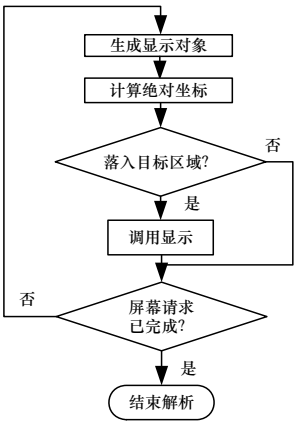


**图 4　屏幕适配探测的流程**

显示对象在样本空间中的位置与屏幕空间共存在 5 种相互关系，如图 5 所示。其中，对象 1 与对象 5 同属落入屏幕空间外的情况，为无效内容；对象 2、对象 3、对象 4 均部分或全部落入屏幕空间，为有效内容，所不同的是对象 2、对象 4 只需显示落入屏幕空间的部分即可。对象 5 完全落入屏幕空间之外，且纵坐标大于屏幕空间的下边界，因此，可以作为屏幕请求已完成的标志，当解析器遇到对象 5 时，即可结束对目标页面的解析，完成屏幕显示。

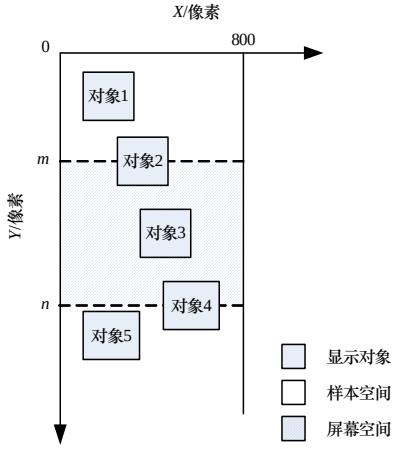


**图 5　显示对象在样本空间中的位置关系**

本解析器采用标准 C 语言编写，拥有独立的文字处理和图形图像渲染引擎，具有良好的跨平台性，目前在 Linux、WinCE 等系统平台上运行良好，且已被用于某商业化手持电子阅读器。在 200 MHz 主频、32 MB 内存的运行条件下，以随机选取的中文新闻类网页为样本，对使用 SMD 方法前后，显示同一页面所需的显示调用次数、解析时间以及内存占用情况进行了统计，如表 1 所示。

显示调用的过程通常会占用大量的动态内存空间，例如生成矢量字库中的字模、调用图片库解析 JPEG 图片等。SMD 方法从机制上避免了所有在目标显示区域外的显示调用，节省了大量系统资源，降低解析器对系统动态内存空间的要求，并加快了解析速度。图 6 是某样本在手持阅读设备上的解析实例。

**表 1　显示调用次数、解析时间及内存占用的统计结果**

| | 未使用 SMD 方法 | | | 使用 SMD 方法 | | |
|---|---|---|---|---|---|---|
| | 显示调用/次 | 解析时间/ms | 内存占用/KB | 显示调用/次 | 解析时间/ms | 内存占用/KB |
| 样本 1 | 2 082 | 1 227 | 8 436 | 766 | 506 | 3 968 |
| 样本 2 | 1 280 | 893 | 9 050 | 521 | 306 | 4 060 |
| 样本 3 | 1 427 | 601 | 8 260 | 656 | 251 | 3 996 |
| 样本 4 | 1 870 | 1 468 | 13 256 | 471 | 627 | 7 012 |

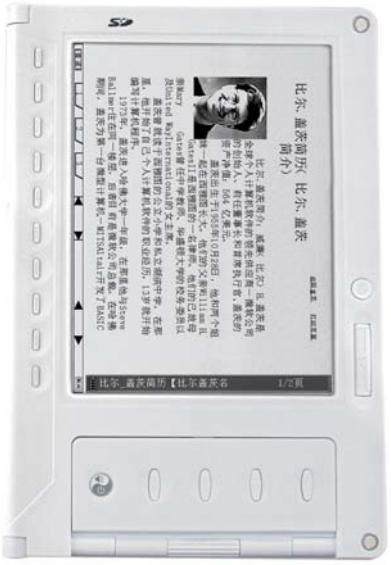

**图 6　某样本在手持阅读设备上的解析实例**

## 4　结束语

HTML 文档是互联网应用最广泛的信息载体，在嵌入式设备上实现对 HTML 文档的解析具有很重要的意义。基于一般嵌入式设备处理能力低、可用内存空间小的特点，本文提出一种新的 HTML 文档解析器设计方法，实现了有针对性的解析，降低了解析器对嵌入式系统动态内存空间的要求，可适用于手持阅读设备。

## 参考文献

[1] Raggett D, Hors A L, Jacobs I. HTML 4.01 Specification[EB/OL]. [2008-04-20]. http://www.w3.org/TR/html401/.

[2] 李效东, 顾毓清. 基于 DOM 的 Web 信息提取[J]. 计算机学报, 2002, 25(5): 526-533.

[3] 王　强, 王继成, 武港山, 等. Web 文档清洗系统中 HTML 解析器的开发[J]. 计算机应用研究, 2002, 19(2): 54-57.

[4] 李庆诚, 刘永超, 刘嘉欣. 平台无关的 PDF 嵌入式高性价比解析器设计与实现[J]. 计算机应用, 2007, 27(增刊): 278-280.

[5] 于满泉, 陈铁睿, 许洪波. 基于分块的网页信息解析器的研究与设计[J]. 计算机应用, 2005, 25(4): 974-976.

[6] 伍　星, 王　茜. 设计模式在 HTML 解析器中的应用[J]. 计算机工程, 2005, 31(2): 89-90.

编辑　顾姣健