

基于 HTML5 的电子白板的设计与实现

朱正扬, 双锴

(北京邮电大学, 网络与交换技术国家重点实验室, 北京 100876)

摘要: 电子白板是一种灵活的图像展示工具, 在课堂教学和视频会议等场景有广泛的应用。相对于传统的电子白板设备, 本文设计了一种运行在浏览器上的电子白板。它主要基于 HTML5 技术中的 WebSocket 和 Canvas, 实现了协同编辑和画板的功能。此外还加入了文档转换的功能, 使用户可以在电子白板中编辑多种格式的文档。本系统不需要用户安装额外插件, 并且支持移动端, 可扩展性强。

关键词: 计算机应用; 电子白板; WebSocket; Canvas;

中图分类号: TP393

Design and implementation of Electronic Whiteboard Based on HTML5

ZHU Zhengyang, SHUANG Kai

(Beijing University of Posts and Telecommunications, State Key Laboratory of Networking and Switching Technology, Beijing 100876)

Abstract: Electronic whiteboard is a flexible image display tool, in the classroom teaching and video conferencing and other scenes have been widely used. Compared with the traditional electronic whiteboard system, this paper designs an electronic whiteboard which runs on the browser. It is based on the WebSocket and Canvas of HTML5, and realizes the collaborative editing and drawing function. In addition, document conversion is a part of this system. Users can edit multiple format file in this electronic whiteboard. This system does not require users to install additional plug-ins, and supports mobile terminal. The servers and clients can be expanded easily.

Key words: Computer Application; Electronic Whiteboard; WebSocket; Canvas;

0 引言

随着社会发展和科技进步, 汇报讲演时往往需要借助图片、视频等数字信息进行展示。传统的黑板依赖手写文字和手绘图形, 形式单调, 输入较慢, 不能完全满足人们的需求。电子白板能够展示、编辑数字化的图片, 同时能够自由绘画^[1]。相对于 PPT, 电子白板修改更为便捷, 不需要安装专门的软件。由于电子白板交流具有灵活、方便、及时的应用特点, 因此被广泛地集成在远程教学、视频会议、自动化办公、ERP 软件中, 具有很高的实用价值^[2]。

目前, 从事电子白板研发工作的有松下、巨龙和汉王等公司^[3], 它们主要针对硬件外设进行开发, 例如电子笔。然而, 随着通信行业的发展, 支持触屏的移动终端变得非常容易获取^[4]。能够运行在这些移动终端的电子白板, 是有可能代替需要购买辅助外设的传统电子白板的。近年来, HTML5 技术的快速发展, 使得我们能够实现只需要浏览器支持的电子白板。

本文设计并实现了基于 HTML5 的电子白板。通过文档转换, 使用户的本地文档能够在电子白板中进行编辑。完成了画板的功能, 可以选择不同的颜色和粗细的笔进行绘画。采用 B/S (Browser/Server) 模式, 多个用户能够仅依靠浏览器对同一个白板进行编辑。

作者简介: 朱正扬(1991-), 男, 研究生, 主要研究方向: 下一代网络技术

通信联系人: 双锴(1977-), 男, 副教授, 主要研究方向: 下一代网络技术. E-mail: Shuangk@bupt.edu.cn

1 系统总体设计

系统架构如图 1 所示, 由浏览器客户端, 文档转换服务器和同步服务器三部分组成, 用户通过浏览器与整个系统进行交互。浏览器前端提供一个画布, 负责展现白板 and 实现编辑操作。文档转换服务器负责将用户本地上传的文档转换成图片, 将结果传回前端浏览器的画布中。同步服务器负责将用户的操作实时展现在同一会话中的其它用户的画布中。这三部分可以部署在不同的机器节点之上, 具有分布式, 简单可依赖的特点。

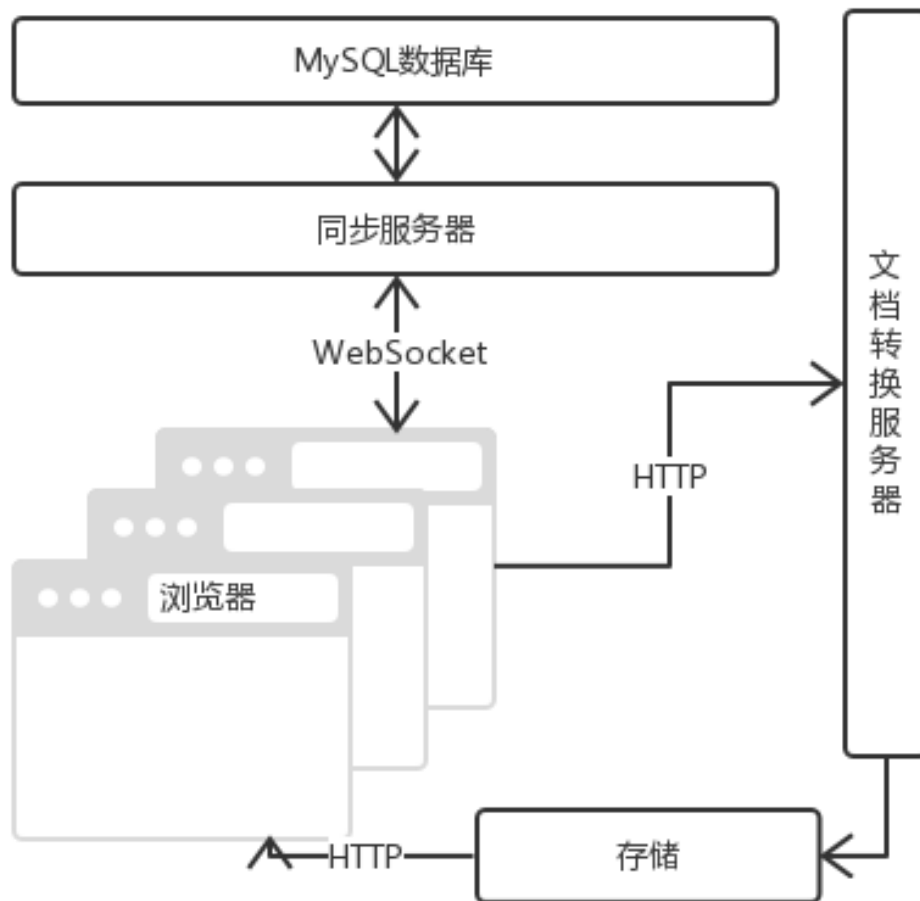


图 1 系统总体设计图

Fig. 1 Overall Design of the System

客户端使用 HTML5 中的 Canvas 标签来实现画布。Canvas 是 HTML5 规范中在浏览器中绘图的程序接口, 它为开发人员提供了一套二维绘图系统。在此之前, 开发人员若要在浏览器中绘图, 只能使用 Adobe 的 Flash 和 SVG(Scalable Vector Graphics, 可伸缩矢量图形), 然而使用 Flash 的解决方案需要借助 Flash Player 插件来实现^[5]。因此, 在本系统中用户可以通过个人电脑或者移动设备上支持 HTML5 的浏览器接入系统, 无需安装额外的硬件设备或软件插件。

此外, 在客户端与同步服务器的连接上, 采用了 HTML5 中的 WebSocket^[6]协议。WebSocket 是 HTML5 新标准中的通信机制, 分为服务器和客户端。服务器放在后台, 保持与客户端的长连接, 完成双方通信的任务, 能够实现稳定全双工实时通信, 具有简便高效的

特点，为开发者提供了一种新的实现 Server Push 的方式^[7]。客户端也是实现在支持 HTML5 浏览器中，通过提供 JavaScript API 使用网页可以建立 WebSocket 连接。因此，本系统需要 HTML5 的支持，是基于 HTML5 进行开发的。

2 电子白板客户端的详细设计

本系统中电子白板的客户端实质上是一个由 HTML，JavaScript 和 CSS 组成的网页。使用 HTML5 的 Canvas 标签构建一块画布，通过 form 标签的 action 属性调用文档转换服务器提供的文档转换服务，页面加载完成后使用 WebSocket 与同步服务器建立长连接。

2.1 电子白板的打开方式

用户通过在浏览器中输入 URL 进入本客户端。URL 中使用 whiteboard 字段修饰一个由同步服务器生成的会话 ID，形如 `www.example.com/index.html?whiteboard=id`。首先，客户端判断 whiteboard 参数是否存在，如果不存在则进入一个空白的电子白板，如果存在则取得这个 ID，传入同步服务器进行校验。校验成功则进入对应的白板，校验失败则进入空白的白板。客户端开启的流程图如图 2 所示。

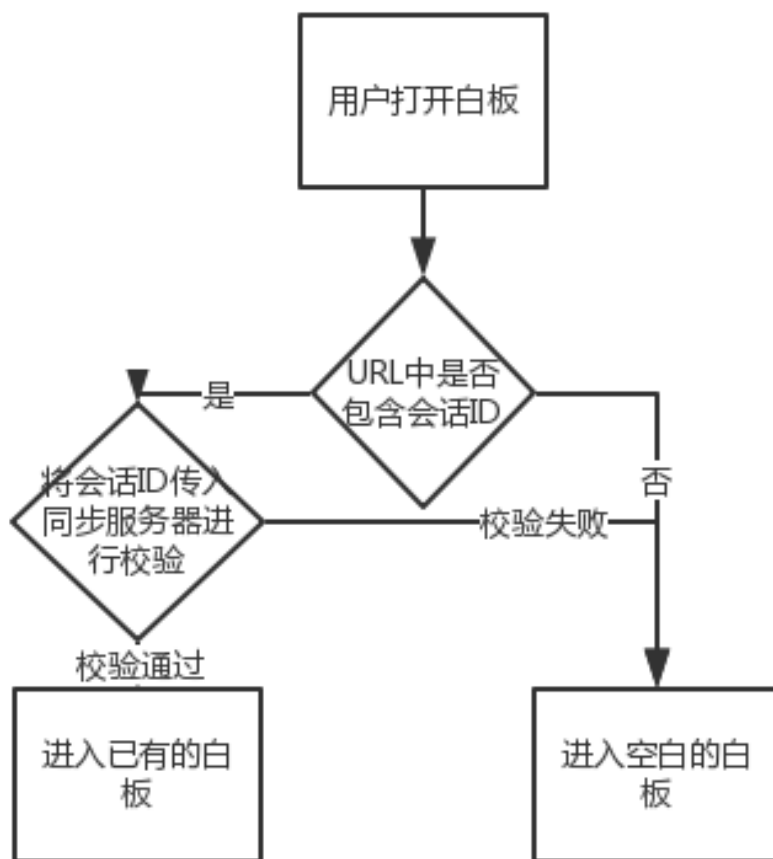


图 2 打开电子白板

Fig. 2 Open the Electronic Whiteboard

用户在进入电子白板之后，可以进行两类操作。一类是将本地资源上传到电子白板中打开，另一类是选择不同颜色和粗细的画笔对电子白板上的图形进行编辑。对于第一类操作，

根据不同的资源类型，又分为两种打开方式。第一种是图片类型的资源。对于图片，使用 HTML 中的 input 标签来接收本地上传的资源，type 设为 file，accept 设为 image/。然后在 JavaScript 脚本中使用 HTML5 的 FileReader 来接收所上传的图片。最后使用 Canvas 的 drawImage 方法将图片渲染到画布中。第二种是 word、ppt、pdf 等形式的文档型资源。对于这种资源，使用 input 接收之后，将文档传递给文档转换服务器，然后等待文档转换服务器将结果回传给客户端，客户端拿到转换结果之后再将其渲染到画布之上。

2.2 画板功能的实现

系统采用 HTML5 的 Canvas 实现画板的功能。通过记录起始点的坐标（start_x，start_y）和终点的坐标（end_x，end_y）就可以连成一个线段，把这些线段拼接起来就是一副图形。期间通过设置不同的颜色和笔触的粗细，就可以实现五颜六色的画板了。首先，如表一所示，为 Canvas 绑定鼠标等监听事件（EventListener），记录画笔的坐标和移动轨迹。

表 1 电子白板客户端监听事件
Tab. 1 EventListener of Electronic Whiteboard Client

事件	说明
mousedown	按下鼠标，记录起始坐标（start_x，start_y），开启 mousemove 监听
mouseup	松开鼠标，关闭 mousemove 监听
mousemove	移动鼠标，记录当前坐标（end_x，end_y），在画布上画出当前坐标和起始坐标的连线，将数据发送给同步服务器，最后将起始坐标替换为当前坐标
mouseout	鼠标移出画布，关闭 mousemove 监听
touchstart	移动设备中（多个）手指放在画布上，将各个触点的起始坐标记录在数组中
touchend	移动设备中（多个）手指从屏幕上移开，将各个触点的当前坐标和起始坐标在画布上连线，发送到同步服务器，将起始坐标从数组中移除
touchcancel	停止跟踪触摸，清空数组中的元素
touchleave	同 touchend
touchmove	同 touchend
message	接收文档转换服务器发送的转换完成的信号

通过在 JavaScript 脚本中对鼠标绑定监听事件，能够实现画笔的功能。鼠标按下，初始化起始点坐标，开始记录轨迹。鼠标移动，将当前坐标和起始点坐标连线绘在画布之上，将这个行为发送给同步服务器以传递给处于同一个会话中的其它客户端实现同步，最后将起始点坐标更新为当前坐标。鼠标松开或移出画布，结束这次绘制。特别的，本系统采用 HTML5 中的触摸事件（touch）来支持移动终端的多点触控的绘制。因为移动设备既没有鼠标也没有键盘，所以在为移动浏览器开发交互性网页的时候，PC 端的鼠标和键盘事件是不够用的。触摸事件是 HTML5 中新添加的事件，受到了 IOS 和 Android 上的浏览器的广泛支持。触摸事件会在（多个）手指放在屏幕上面、在屏幕上滑动和从屏幕上移开等时候触发。此时，除了将移动轨迹的坐标记录下来，还需要记录是哪一条轨迹。本系统将坐标和轨迹标识打包为一个对象存储在数组中，每次绘图时遍历数组取得对应轨迹标识的坐标值进行绘制，触摸结束后清空这个数组。

在 Canvas 上实现画笔颜色和粗细的选择比较简单。Canvas 的颜色选择通过将其

strokeStyle 属性设置为 CSS 中 RGBA 值来实现，具体的值在前端选择后由 JavaScript 获取。选择白色的画笔，rgba(0,0,0,1)，就相当于橡皮擦的功能。画笔的粗细通过给 lineWidth 属性赋值实现。lineWidth 是一个整形的值，以像素为单位，默认为 1，值越大画出来的线条越粗。

2.3 电子白板的保存

电子白板的画布是 Canvas 元素，Canvas 提供了一个接口 toDataURL 可以将画布上的图形转换为 Base64 编码。Base64 将图形的二进制信息中每 3 个 8 位字节（38=24）转化为 4 个 6 位的字节（46=24）^[8]，是网络上最常见的用于传输字节代码的编码方式之一。利用这段编码，本系统实现了三种电子白板的保存方式。第一，在客户端的画布的下方添加了一个 img 标签作为对比图，将 src 属性设置为画布的 Base64 编码，直接将 Canvas 转换成了可右键保存的图片。第二，在客户端的后台接收到编码后，将头部“data:image/png;base64”字段去掉，然后解码，以图片的形式保存在后台服务器中。第三，将编码发送给同步服务器，同步服务器将之存储在 MySQL 数据库中作为当前画布的一种快照进行保存。当用户想要回溯到之前的版本或执行撤销操作时，从同步服务器请求对应的编码可以对画布进行覆盖式还原。

3 电子白板服务器的详细设计

3.1 同步服务器

本系统设计同步服务器的目的是将用户在电子白板上进行的操作实时地呈现到其它用户的电子白板之上，实现协同编辑的功能。具体实现上，采用 Node.js 语言编写了一个常驻进程，和客户端之间建立 WebSocket 连接，对编辑操作进行存储转发。Node.js 是 JavaScript 语言的服务器运行环境，提供了大量工具库，使得 JavaScript 语言能够与操作系统和数据库互动。重要的是，Node.js 提供了 WebSocket 的实现库，并且得到了广泛的支持，因此选择 Node.js 来实现同步服务器。

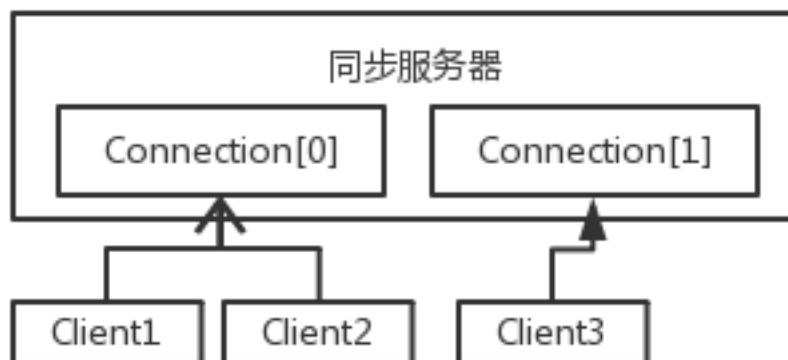


图 3 同步服务器

Fig. 3 Synchronous Server

首先, 同步服务器要对每一个新的会话生成一个唯一的标识 ID。生成方法是在 52 个大写字母和 10 个数字里随机取 9 个进行拼接。之后新建一个对象对这个会话进行维护, 命名为 Connection。有新的客户端连入同步服务器时, 根据客户端所携带的会话 ID 分配到各自的 Connection 中, 如图 3 所示。Connection 记录有客户端的地址信息, 创建时间, 用户数量和会话 ID。当用户数量为零时, 删除这个 Connection 对象。

本系统定义电子白板客户端向同步服务器发送的消息类型主要有以下几种: 1. hello, 每一个客户端进入会话时发送给同步服务器, 其它客户端收到后发送 init 消息; 2. init, 客户端将当前画布上的图形以 Base64 编码格式发送给同步服务器, 其它客户端收到后将自己的画布替换为收到的图形; 3. draw, 客户端绘制时发出, 包括起点坐标、终点坐标、画笔颜色和画笔粗细, 其它客户端收到后在各自的画布中执行对应的绘制; 4. clear, 清空画布; 5. syn, 同步客户端画布的长和宽以及画布上的内容。

同步服务器接收到客户端发来的以上消息后, 根据客户端的会话 ID 选择对应的 Connection, 然后对这个 Connection 进行遍历, 向里面每一个客户端转发接收到的消息。对于 draw 类型的消息, 表示客户端修改了白板上的图像, 同步服务器将图像的二进制编码存储在 MySQL 数据库之中, 以供用户执行撤销操作。总之, 同步服务器的主要工作是维护客户端的路由表, 对客户端发来的消息以单播的形式进行转发。

3.2 文档转换服务器

文档转换服务器的功能是将 Word、PPT 和 Excel 等无法直接以图片形式打开的文件转换成图片, 以在电子白板中打开再进行编辑。由于要利用两个开源的 JAVA 程序, 因此文档转换服务器选择使用 JAVA 语言来实现 Web 服务器。其中一个程序叫做 JODConverter, 它利用 OpenOffice 提供的接口, 将 Word、PPT 和 Excel 等 Office 格式的文档转换成 PDF 格式。另一个程序叫做 ICEpdf^[9], 本系统利用它将 PDF 格式的文档转换成 JPG 格式的图片。

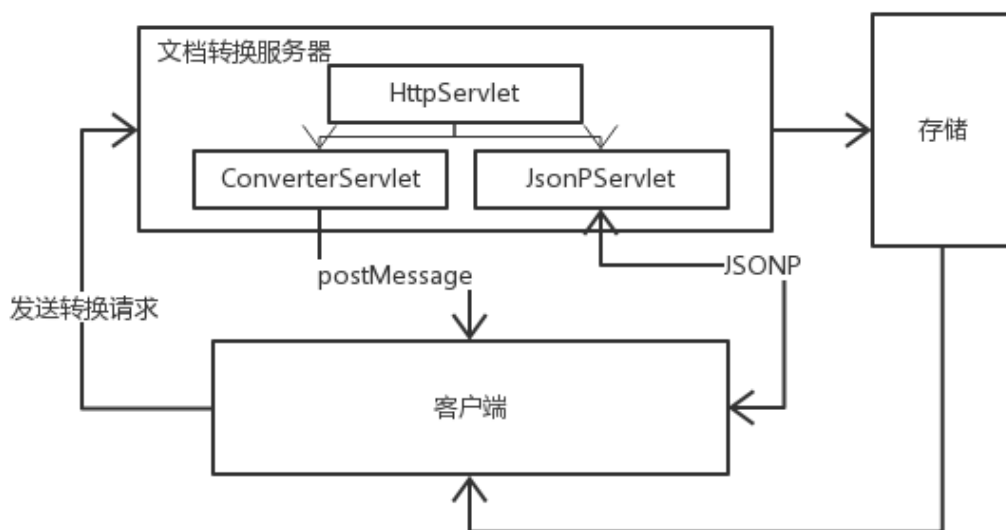


图 4 文档转换服务器

Fig. 4 Document Conversion Server

如图 4 所示, 客户端通过 form 表单将本地文档上传到文档转换服务器, 文档转换服务

器将请求指派给转换器 ConverterServlet 进行文档转换。ConverterServlet 将转换结果 JPG 图片放到一个统一的存储文件夹下。为了防止冲突,每份转换结果都在存储文件夹下新建一个子目录,以当前时间、源文件名、进程号 PID 和线程号 TID 来命名。利用 Apache 配置一个虚拟主机到存储文件夹,就可以实现从网络上对转换结果的访问。由于转换结果是以 0.jpg, 1.jpg, 2.jpg……的递增形式命名的,因此客户端只需要知道子目录的名字和结果的份数就可以通过网络位置获取转换结果了。

当转换完成后,需要给客户端返回一个“成功”的信号,让客户端去获取结果加载到画布上。本系统采用 HTML5 中引入的消息机制 postMessage 来实现。postMessage 允许来自不同源的脚本采用异步方式进行通信,实现跨域消息传递十分方便。在服务器端 ConverterServlet 部分, response 时加上 postMessage 的 JavaScript 标签即可。在客户端部分,如表 1 所示,对 message 进行事件监听,收到转换完成的消息后,客户端发出 JSONP 格式的异步跨域请求,文档转换服务器的 JsonPServlet 将结果目录和结果份数以 JSON 格式返回。与 postMessage 不同的是,JSONP 连接是由客户端发出的,而且可以多次发送,最重要的是相对于字符串类型的消息,JSON 格式的解析不易出错。

表 2 文档转换后的图片大小

Tab. 2 Picture Size after Document Conversion

源文件格式	转换后图片大小
Excel	宽度 1190 像素, 高度 1684 像素
PDF	宽度 1190 像素, 高度 1684 像素
PPT	宽度 1560 像素, 高度 1080 像素
Word	宽度 1190 像素, 高度 1684 像素

客户端通过网络位置向存储请求转换结果,对当前页号加减一实现了“上一页”和“下一页”的功能。实验表明,如表 2 所示,不同格式的文档经过转换后生成的图片尺寸并不一致,尤其是要支持各种尺寸的本地图片。因此,客户端的画布尺寸需要根据要加载的图片尺寸进行动态调整,并向同步服务器发送 syn 消息。总之,以上的各种消息通信对于用户都是透明的,用户只需要上传本地文档就可以在画布中进行编辑了。

4 结论

HTML5 的发展使浏览器可以承担电子白板客户端的功能,并且可以适配移动端,具有跨平台的特点。本文利用 HTML5 的 Canvas、WebSocket、FileReader、postMessage 和触摸事件等特性设计并实现了一套电子白板系统。使用 Node.js 编写了同步服务器,使用 JAVA 编写了文档转换服务器,模块划分清晰,可以分别部署,也可以为其它应用提供服务,具有高扩展性。在客户端,无需安装任何插件,与后台通信透明,提供了良好的用户体验。

[参考文献] (References)

- [1] 莫正德. 浅谈电子白板在教学中的优势[J]. 中学教学参考, 2015, 12: 111-111.
- [2] 洪晟, 熊华钢, 张其善. 一种改进的协同式电子白板的设计与实现[J]. 计算机工程, 2008, 2(2):261-263.
- [3] 阎赛, 张雪, 连立杰,等. 电子白板技术在我国的应用状况[J]. 电视技术, 2013, 37(z2).
- [4] 陈丹霞. 基于 J2ME 电子白板系统的研究与实现 [OL]. 北京: 中国科技论文在线 [2009-01-04].http://www.paper.edu.cn/releasepaper/content/200901-80.
- [5] 李岸岑. 以写字本为例辨析 Canvas 绘图机制的优劣[J]. 计算机时代, 2014, 8:17-18.
- [6] I. Fette, RFC6455-2011, The WebSocket Protocol[S]. USA: IETF, 2011.
- [7] 才鑫, 双错. 基于 WebRTC 技术的多人视频通信系统 [OL]. 北京: 中国科技论文在线 [2014-12-16].http://www.paper.edu.cn/releasepaper/content/201412-461.

195

- [8] Zhang D, Liu Z, Liu D, et al. Research on application of XML-based raw image coding in embedded network devices[A] Computer Science and Service System (CSSS), 2011 International Conference on. IEEE[C], 2011:2423-2425.
- [9] Ajedig M A, Li F, Rehman A U, et al. A PDF Text Extractor Based on PDF-Renderer[J]. Lecture Notes in Engineering & Computer Science, 2011, 2188(1).