

基于 HTML5 的 3D 多人网页游戏实现方案

冯科融,王和兴,连加美,刘伟,贾贺杰

(东北大学秦皇岛分校 计算机与通信工程学院,河北 秦皇岛 066004)

摘要: HTML5 标准增强了网页在图形绘制、媒体播放、消息传送等方面的能力,为提高网页游戏性能创造了条件。基于该标准开发出一个 3D 多人网页游戏。介绍了其实现方案,测试了其相关性能,分析了该技术的可行性。

关键词: HTML5; WebGL; WebSocket; THREE.js; JavaScript

中图分类号: TP311

文献标识码: A

文章编号: 1674-7720(2013)01-0004-03

Design of 3D browser-based multiplayer online game based on HTML5

Feng Kerong, Wang Hexing, Lian Jiamei, Liu Wei, Jia Hejie

(College of Computer and Communication Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China)

Abstract: HTML5 standard enhances the ability of the page in the graphics rendering, media playing and messaging, it can improve the performance of browser-based multiplayer online game (MOG). This paper created a 3D browser-based MOG based on HTML5, and discusses the implementation and performance of our game and analyses the prospects of HTML5.

Key words: HTML5; WebGL; WebSocket; THREE.js; JavaScript

中国网游产业已走过了 11 年,文化部发布的《2011 中国网络游戏市场年度报告》显示,2011 年中国网游市场规模为 468.5 亿元^[1]。近年来,网页游戏(页游)因其方便省时等优点受到玩家青睐,正不断占领市场。2011 年开始,3D 页游的出现又使页游品质大幅提升。可以预测,高品质页游将在未来网游市场中占据一席之地。本文以自主研发的基于 HTML5 的 3D 多人网页游戏 BillyRun 为例,介绍了其实现方案,分析了其程序性能并展望了技术前景。

1 核心技术分析

1.1 以 HTML5 标准中 Canvas 元素为载体呈现游戏图像

HTML5 是最新的网页标准。广义论及 HTML5 实际是指包括 HTML、CSS 和 JavaScript 在内的一套技术组合,其目的在于减少浏览器对插件的依赖性和丰富网页的表现形式^[2-3]。Canvas 元素是 HTML5 中新增的页面元素,该元素是一个图形绘制容器,拥有一个基于 JavaScript 的 API,可由脚本动态绘制图像^[4]。BillyRun 中的 Canvas 元素用于两个方面:一是呈现和更新 3D 游戏画面,二是绘制游戏中的部分 2D 画面。目前,IE、Firefox、Chrome、Opera 等主流浏览器均支持 Canvas 元素。

1.2 基于 WebGL 的游戏画面渲染

WebGL 是一种 3D 绘图标准,该标准通过增加 OpenGL ES 2.0 的一个 JavaScript 绑定,为网页中 Canvas 元素提供硬件 3D 加速渲染,使网页 3D 程序能够流畅运行^[5],无需浏览器插件。游戏中场景创建和渲染使用了 THREE.js。THREE.js 是一个 JavaScript 开源 3D 库,能创建包括摄像机、灯光、材质、模型在内的三维场景,并由 WebGLRenderer 模块以 WebGL 标准渲染^[6]。目前 Firefox、Chrome、Opera、Safari 等主流浏览器均支持 WebGL 标准。

1.3 基于 NodeJS 的服务端程序

NodeJS 是基于 Google V8 引擎由 JavaScript 实现的服务器端可扩展程序,可作为编写高效网络服务程序的平台^[7]。NodeJS 消除了服务器端程序与客户端程序的语言差异,使代码可以复用,方便了页游开发。BillyRun 服务器端由 NodeJS 搭建,其中定义了游戏信息,并通过 Socket.IO 模块与各客户端实时通信完成更新。服务器端连接了 MySQL 数据库,用于保存用户账号信息。目前服务器端在 Linux、Windows 7 和 Windows Server 2003 下均可运行。

1.4 基于 WebSocket 技术的网页实时通信

WebSocket 是 HTML5 提供的浏览器与服务端间全双工通信技术,主要用于实时通信服务。WebSocket 十分高效,只须浏览器和服务器进行一次握手,便可双向传递数据。Socket.IO 是 Node.JS 的一个开源模块,它对 WebSocket 进行了封装,提供了建立连接和消息推送的接口。BillyRun 中基于此模块在服务器端建立了 Socket 服务,与每个客户端相连,完成信息更新和消息处理工作。目前,IE、FireFox、Chrome、Opera 等主流浏览器均支持该技术。

1.5 Audio 以及基于 Web Audio API 的 3D 音效

Audio 元素是 HTML5 中新增的元素,可实现音频文件播放等基本功能,支持 HTML5 的浏览器都能良好实现。Web Audio API 由谷歌提出,是一套用于网页音频处理的 JavaScript API^[8],其功能强大,尤其是 3D 定位和音锥技术,可根据声源与玩家的位置关系,控制左右声道实现立体音效。该技术能大大提高游戏品质,但目前只有 Chrome 浏览器能够支持。BillyRun 中分别实现了这两套音效方案,兼顾了游戏品质和兼容性问题。

2 具体实现方案

2.1 游戏内容设计

BillyRun 是一个空中对战射击游戏,游戏场景包括地面、天空和建筑物,每个玩家操作一架直升机在场景中飞行,可与其他玩家展开攻击或防御,已有多种游戏模式判定胜负。

2.2 程序整体结构

BillyRun 由服务器端程序和客户端程序两部分组成。服务器端程序实现了数据库交互、玩家事件处理、界面跳转等功能;客户端程序实现了界面显示、音效播放、游戏主循环等功能,运行在客户浏览器上。BillyRun 功能结构图如图 1 所示。

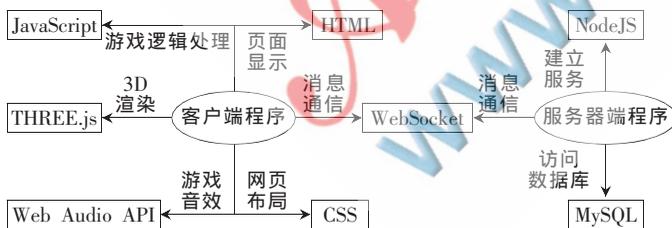


图 1 功能结构图

2.3 程序执行流程

2.3.1 程序整体流程

程序由客户端和服务端组成,两端通过通信保持同步,其整体流程图如图 2 所示。

2.3.2 客户端程序流程

客户端包含界面、加载、主循环、消息处理和输入系统 5 部分。

(1) 游戏有 5 个 div 元素封装的界面。玩家先后进入

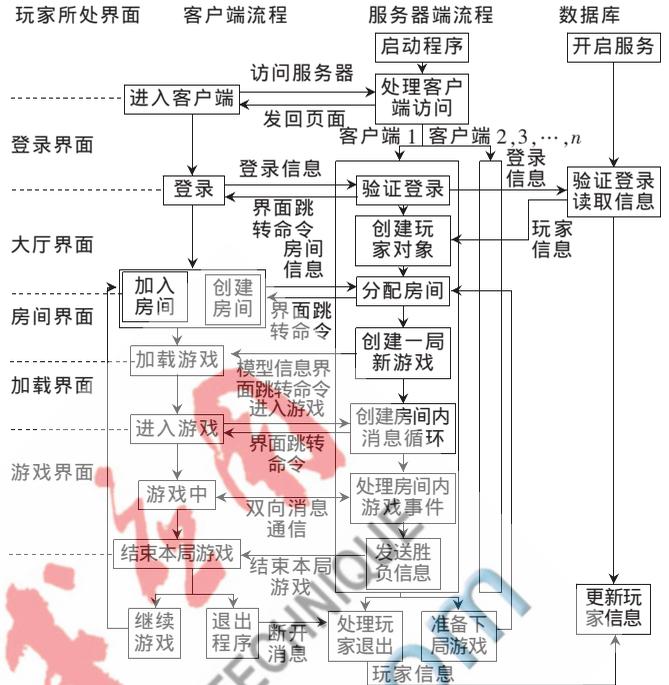


图 2 程序流程图

登录界面、游戏大厅界面、游戏房间界面、加载界面和最后的游戏界面,每局游戏结束时返回游戏大厅界面。

(2) 游戏的加载系统

包括 4 部分:①场景模型和音频文件。为固定游戏资源,加载后保存在程序中,仅进行一次;②游戏信息。加载在每局游戏开始前执行,更新客户端信息;③玩家模型。加载在每局开始前执行;④加载本局游戏所需的玩家模型,使用如图 3 所示的动态加载方法,避免了重复加载,提高了程序效率。

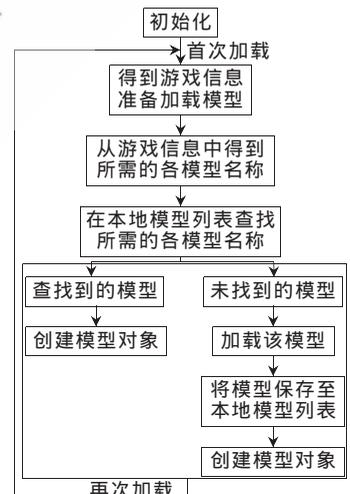


图 3 加载图

(3) 主循环由

requestAnimationFrame 构建。循环首先检测玩家所处界面,若处于游戏界面,程序会获取与上次循环的时间间隔,完成信息更新、碰撞检测、画面重绘等工作;若处于其他界面(如加载界面),表示游戏未进行,程序将等待下次循环。客户端更新频率为 45 F/s,主循环每帧执行,以保证游戏的流畅性。

(4) 消息处理模块核心是一个 socket 对象,与服务端连接,向服务器端发送自身玩家信息,并接收服务器端发送的其他玩家信息和系统命令。

(5) 输入系统由文档对象 document.addEventListener 方法监听鼠键事件。游戏输入信息将缓存在程序中,由主循环处理;非游戏输入将即时处理。

表 4 网络传输性能测试表

客户端型号	服务器端型号	消息延迟/ms	模型大小/KB	加载用时/ms	首次进入游戏时/ms	再次进入游戏时/ms
Client3	Server1	30	3 028	934	1 214	348
Client4	Server1	32	3 028	659	858	513
Client2	Server2	10	3 028	9 434	12 207	305

2.3.3 服务器端程序流程

服务器端程序启动后监听端口,为登录玩家建立服务。对每个玩家,程序先访问数据库验证登录信息,验证通过后为其建立玩家对象和 socket 连接,加入玩家列表。当玩家所处房间可进入游戏时,程序命令该房间所有玩家进入游戏,开启该房间消息循环。游戏中重要逻辑处理如胜负判定都由服务器端完成。一局游戏结束后,服务器端命令玩家返回大厅房间准备下轮游戏。当玩家退出时,服务器端访问数据库更新信息,在程序中将该玩家删除。

3 程序性能测试

本文选择了不同配置的 4 个客户端和 2 个服务器端测试程序性能,机器配置如表 1 所示。

表 1 机器配置信息表

机器型号	CPU	主频 /GHz	内存 /GB	显卡
Client1	Intel Core i5 M520	2.40	2.00	NVIDIA GeForce GT 240M
Client2	Intel Core i7 Q720	1.60	2.99	NVIDIA GeForce GT 240M
Client3	Intel Core2 Duo T6670	2.20	1.90	ATI Mobility Radeon HD 3400 Series
Client4	Pentium Dual-Core T4200	2.00	2.00	NVIDIA GeForce G 103M
Server1	Intel Xeon E5620	2.40	1.00	-
Server2	Intel Core i5 M520	2.40	2.00	-

3.1 客户端程序性能测试

如表 2 所示,在 4 款不同配置的客户端机器上,游戏稳定时帧速均为 45 F/s;内存占用情况相近,约为 350 000 KB;而显卡与 CPU 性能越强的客户端,其 CPU 占用率越低。总体来说,在 4 个客户端上均能流畅运行,达到了 3D 游戏的要求。

表 2 客户端性能测试表

机器型号	帧速/FPS	内存/KB	CPU 占用/%
Client1	45	343 644	10
Client2	45	340 432	5
Client3	45	353 464	28
Client4	45	355 836	27

3.2 服务器端程序性能测试

如表 3 所示,服务器端程序对内存和 CPU 的占用较低,能较轻松地支持 2~8 人游戏,并有增加玩家负载的潜力(但负载上限还需要进一步测试)。

表 3 服务器端性能测试表

机器型号	在线人数	内存/KB	CPU 占用/%
Server2	2	16 988	1
Server2	4	29 320	2
Server2	6	33 240	2
Server2	8	36 580	2

3.3 网络传输性能测试

网络传输性能测试结果如表 4 所示。由表 4 可以看出,受网络情况影响,加载用时差异较大。而动态加载技术避免了重复加载模型。因此当首次加载完后,再次进入游戏时用时大大减小,也即减少了玩家的等待时间。

HTML5 的出现为页游提供了新的实现方案,通过对 BillyRun 的分析测试,证实了该方案的可行性。目前 HTML5 标准还不够成熟,部分浏览器尚不能全面支持,基于该技术的页游也相对较少。但随着 HTML5 标准的推广,尤其是 PC 端和移动端浏览器的不断支持,其跨平台的优势将得以充分体现。在不远的将来,基于 HTML5 的页游将有机会成为市场的主导。

参考文献

- [1] 新华社. 文化部发布《2011 中国网络游戏市场年度报告》[EB/OL]. (2012-03-30) [2012-08-06]. http://www.gov.cn/jrzq/2012-03/30/content_2103576.htm.
- [2] 刘杰. 浅谈 HTML5 技术[J]. 科技创新导报, 2012(10): 29-35.
- [3] CHEN B J, XU Z Q. A Framework for browser-based multiplayer online games using WebGL and WebSocket [C]. Multimedia Technology (ICMT), 2011 International Conference, 2011:471-474.
- [4] YANG J P, ZHANG J. Towards HTML 5 and interactive 3D graphics [C]. Educational and Information Technology (ICEIT), 2010 International Conference, 2010:522-527
- [5] Khronos Group. WebGL 1.0—2011 WebGL[S]. 2011.
- [6] Mrdoob. JavaScript 3D library[EB/OL]. (2012-04-22) [2012-08-06]. <https://github.com/mrdoob/three.js/>.
- [7] KIESSLING M. The node beginner book[EB/OL]. (2012-02-12) [2012-08-06]. <http://www.nodebeginner.org/>.
- [8] ROGERS C. Web audio API W3C working Draft 02 august 2012—2012 Web Audio API[S]. 2012.

(收稿日期:2012-09-04)

作者简介:

冯科融,男,1990年生,本科,主要研究方向:3D 页游。