

Clouda 实时编程应用开发技术^①

吴 慧, 卞艺杰, 马瑞敏

(河海大学 商学院, 南京 211100)

摘 要: 针对开发者在移动端开发原生应用时, 面临数据碎片化、终端分裂化、用户量剧增三方面的问题, 一种新的开源技术框架 Clouda 应运而生. 简单介绍了 Node.js, 重点介绍了 Clouda, 分析了它的 5 大技术优势, 与同类框架进行对比; 搭建 Clouda 框架环境, 在该环境下面实现了 RSS 阅读器, 最后探索 Clouda 存在的不足, 以及其未来发展前景.

关键词: Clouda; 轻应用; 实时编程; 云端一体化

Real-Time Programming Application Development Technology Clouda

WU Hui, BIAN Yi-Jie, MA Rui-Min

(Business School, Hohai University, Nanjing 211100, China)

Abstract: When developing native applications on mobile terminal, developers are faced the data fragmentation, terminal fragmentation and user surge problems. A new open source technology framework — Clouda is emerged. This paper briefly describes Node.js, highlights Clouda, and analyzes its five major technical advantages. This work compares Clouda with similar frameworks, and sets up a development environment, develops RSS reader by Clouda. At last, the future direction of Clouda is analyzed.

Key words: Clouda; light applications; real-time programming; cloud integration

在瞬息万变的移动互联网时代下, 曾经主流的 Native APP 开发者正面临着前所未有的挑战. 终端用户的 Native APP 数量在增多, 但日均启动个数却在减少; 用户的使用时间不断向高频 Native APP 转移, 加剧了“头部效应”. 对于低频和不知名的 Native App, 则面临着更严峻的“分发”和“使用”长尾困境, 开发者已经迫切需要改变这种局面. 同时, Web 技术也是跨终端高成本的最好解药. 而且在当前的应用商店中, 90% 以上的下载流量往往集中在 30% 的“头部应用”上, 开发者所开发的新应用, 通过传统的分发渠道很难在短时间内获得预期的用户量、知名度. 同时伴随着 iOS、Android 逐渐成熟运转, 开发者在移动端开发原生应用时, 将面临数据碎片化、终端分裂化、用户量剧增三方面的问题. 为了解决以上问题, 并考虑到 Web 技术对各种移动终端具有良好的兼容性, 并可自动升级等特征, 一种新的开源技术框架 Clouda 应运而生. 它作为一款实时 Web App 开发框架, 具有跨终端、云端

统一、高速实时、随动反馈式编程、可检索等特性, 可帮助开发者轻松进行“轻应用”开发, 也让用户真正进入全实时移动互联网时代.

1 基本概念介绍

1.1 Node.js

Node.js 采用 C++ 语言编写而成, 是一个基于 Google Chrome V8 引擎的 JavaScript 的运行环境(支持的系统包括 *nix、Windows), 你可以编写系统级或者服务器端的 Javascript 代码, 交给 Node.js 来解释执行^[1]. 它具有单线程的特性, 代码运行起来就是一个独立线程, 它充分利用了 js 语言天生的事件式编程, 以及非阻塞 I/O(Non-Blocking)的特性, 它所有的库和 API 几乎都首选异步调用的方法, 它受益于 V8, 性能优越运行速度快, 而且只要浏览器的性能竞赛还在继续, 那么 V8 的性能还会继续提升. 作为一个新的运行环境还不成熟, 版本更新很频繁, API 等还在继续变动中,

^①收稿时间 2013-12-09;收到修改稿时间:2014-01-20

异步编程提高了编程的难度和复杂度。

1.2 Clouda

Clouda 是百度历时两年研发出来的新一代可被搜索引擎检索的 WebApp 开发框架,但它并非百度完全打造,其中也有社区贡献的力量,为回报于社区,百度将永远免费开源 Clouda,全部代码托管在 Github 上。Clouda 使用 MIT 协议,开发者可对其使用、修改,甚至商用。Clouda 是基于 node.js 的 Webapp 开发框架,在使用 Clouda 时需要安装 node.js 和 MongoDB^[2]。在 Clouda 中开发 app,可以在多终端下实现数据同步、任务迁移。Clouda 支持面向数据的编辑模式,根据数据的变化自动实现界面实时更新,开发者不需描述跃迁过程。它的核心在于云端使用统一的 Javascript 语言,可以同时运行在浏览器、iOS、Android 以及百度框中^[3]。clouda app 可以同时开发服务端和客户端,客户端的数据与云端实时同步,其内部数据也可供搜索引擎检索,解决 app 孤岛以及 99.9%长尾应用只有 30%分发量的搜索分发问题。

2 Clouda的五大技术优势

Clouda 的强大之处在于其裹挟了五大技术优势。首先,跨终端^[4]。支持 WebApp 和 HybridApp 等多种载体,在不同设备和不同分发渠道中始终提供一致的用户体验,实现一个产品处处运行。不同终端间的数据可以流转互通并进行数据同步,用户可随时切换不同设备,享受不间断的持续产品体验。

其次,云端统一。“只用 Javascript 一门语言,抹平云端界限,一份代码处处运行”^[5]。Clouda 开发方式彻底的打通了服务器和客户端,不需要再书写接口文档,不需要生成接口代码,服务器和客户端代码本身就在一起编写。以 Javascript 为统一语言,提供云端统一的编程模型,抹平了传统云与端的开发界限。一种语言可以在云、端两个方向通用,大大降低了开发周期、成本,减少了重复开发的繁琐。

第三,数据传输高速、实时性。借用 Clouda 的开发框架,通过内置高等级的全实时数据通道,全双工上下行推送,完美支持了各终端之间的实时数据同步。即使是离线设备,重新上线后便能自动同步和解决冲突^[6]。这一性能归结服务器与客户端之间始终保持常连接状态,使得服务器的数据信息可以实时推送到服务端,从而保证了服务器与客户端的高效交互。这一

特点弥补了采用传统的方式(比如 php+js)开发 web 项目中,在实时高效性方面存在的不足。

第四,随动反馈式编程^[7]。Clouda 首创了面向数据的编辑模式,开发者只需描述状态的最终形式,不需描述跃迁过程,即能根据数据的变化自动实现界面实时更新,并实现数据订阅状态的自动跃迁,应用该技术后,可减少 30%以上的代码开发量,增强程序稳定性,提高所有类型开发者的开发效率。

最后,可检索性^[8]。这一技术来源于百度天然的搜索优势,也是国内唯一一家能实现改技术的开源技术框架。在 App 应用分发颓势,以及 WebApp 应用趋势的映衬下,可检索的 App 应用,在未来的搜索分发新时代,将具备更强的渠道能力。

3 Clouda与同类框架的对比

过去的观念认为轻应用不如 Native App 体验好,云端技术能力的进一步丰富和开放,让轻应用具备了在前端体验上相匹配的表现。百度 Clouda 开发框架的强大高效是一种技术开发模式上的创新,不仅跨终端、同步,保证体验和数据的一致性,还实现了云端统一的开发语言和编程模型,抹平了云端界限,高速实时性支持各终端实时数据同步。在与同样宣称实时 WebApp 开发的框架对比中, Meteor 同样收到了极大关注。在技术方面两者有着极大的相似点,两者均采用了 Node.JS、MongoDB API 等框架的基础特性,模糊了服务器端和客户端,弱化网络连接质量造成的影响。单从使用方面来看,两者都简单易用。Meteor 宣称其可以帮助程序员轻松构建高品质的实时 Web 应用程序,而 Clouda 同样以高效开发为导向。应用两者完成的应用程序同样运行速度非常快。应用领域方面: Clouda 更加专注于移动方面,全面兼容手机、平板、PC。Clouda 除了开发迅速、开发门槛低等优势外,Clouda 的实时技术足以保证开发者通过内置高等级的全实时数据通道,完美支持各终端之间的实时数据同步。即使是离线设备,重新上线后也能自动同步和解决冲突,保证开发者从容面临挑战。而 Metero 则更多应用于游戏和实时在线文档编辑。

4 Clouda开发环境搭建

Clouda 是简单,可依赖的实时 Javascript 框架,是基于 node.js 的开发框架^[9],我们首先需要安装 node.js,

http://nodejs.org/download/, Clouda 使用 MongoDB 作为本地数据库, http://www.mongodb.org/downloads.

4.1 安装 MongoDB

1) 将下载的压缩包解压, 并在根目录下新建两个文件夹, 分别命名 data(存放数据)和 logs(存放日志), 在 logs 文件夹下新建 mongodb.log 文件以记录日志。

启动服务:

```
D:\mongodb\bin>mongod --dbpath D:\mongodb\data
Thu Nov 15 14:09:45
Thu Nov 15 14:09:45 warning: 32-bit servers don't have journaling enabled by default. Please use --journal if you want durability.
Thu Nov 15 14:09:45
Thu Nov 15 14:09:45 [initandlisten] MongoDB starting : pid=7068 port=27017 dbpath=D:\mongodb\data 32-bit host=ikaros-PC
```

图 1 启动服务

2) 在浏览器中输入 http://localhost:27017 可验证 mongoDB 服务是否成功启动。

3) 将 mongodb 的启动加入到 windows 的服务下, 具体操作如下:

```
C:\Users\ikaros>D:
D:\>cd D:\mongodb\bin
D:\mongodb\bin>mongod --logpath D:\mongodb\logs\mongodb.log --logappend --dbpath D:\mongodb\data --directoryperdb --serviceName MongoDB --install
```

图 2 具体操作

4) 上面的操作(3)成功后就可直接在 cmd 中用 net start MongoDB 指令来启动 MongoDB

4.2 安装和使用 Clouda

1) 安装 Clouda: 用 cmd 进入 node.js 安装的根目录. 然后输入如下指令: npm install -g sumeru 回车, 完成 Clouda 的安装。

2) 使用 Clouda 创建工程: 用 cmd 进入 node.js 安装的根目录. 然后输入如下指令: sumeru init ./myproject, 完成名叫 myproject 这个工程的创建。

3) 运行 Clouda: 用 cmd 进入 node.js 安装的根目录, 在运行 Clouda 前需要启动 MongoDB, 在 cmd 中输入指令(cd myproject 回车 sumeru start 回车)已完成打开 myproject 的操作。

4) 如何访问应用: 在 Clouda 框架中有 works 的实例, 可以在浏览器中输入 localhost:8080/debug.html/works 来运行. (注意端口是 8080 而不是 27017)。

5 Clouda应用实例

本文使用 clouda 框架开发了一个 RSS 阅读器, 现在列出核心代码, 本段代码实现获取 RSS 内容的功能。

```
1) var iconv = require('iconv-lite');
2) var extpubConfig = {}
3) extpubConfig['pubnews'] = {
4) geturl : function(params){
5)     return 'http://www.cnblogs.com/iou123lg/MyPosts.html?type=Home&onlytitle=1'; },
6) resolve : function(originData){
7) data = iconv.decode(originData,'utf-8');
8) var reg = /<item>[\s\S]*?<title>(.*?)</title>[\s\S]*?<guid>(.*?)</guid>[\s\S]*?</item>/img;
9) var arr = [];
10) data.replace(reg, function(m,n,p){
11) arr.push({title:n,link:p}); });
12) var resolved = { topnews: arr };
13) return resolved; },
14) fetchInterval : 8 * 1000, buffer : true
15) }
16) module.exports = extpubConfig;
```

以下代码实现了对 RSS 内容的输出

```
① sumeru.router.add({ pattern: '/news',
② action: 'App.news' });
③ sumeru.router.setDefault('App.news');
④ App.news = sumeru.controller.create(function(env, session){ var view = 'news';
⑤ var getNews = function(){
⑥ session.news = env.subscribe('pubnews', function(newsCollection){
⑦ var obj = newsCollection.getData()[0];
⑧ session.bind('newsBlock', {'topNews': obj['topnews'] }); });
⑨ env.onload = function(){ return [getNews]; }
⑩ env.onrender = function(doRender){ doRender(view, ['push','left']); }});
```

页面显示效果展示:



图 3 页面显示效果

Clouda框架采用了MVC模式来实现web开发,应用代码结构清晰有条理^[10]。Controller是每个场景的控制器,负责实现App的核心业务逻辑。作为最重要的枢纽,Controller,三个主要时态分工明确,Onload()是Controller的第一个时态,Controller中需要使用的数据都在这个时态中加载,它是MVC中Controller和Model建立联系的过程;这个函数中的代码如果开启了Server渲染,则很可能会在Server端执行,这就是为什么Clouda框架开发的应用冷启动速度优于一般的HTML5应用,因为在onload()函数中,服务器执行了部分js代码,使得客户端节省了这部分代码在服务器上执行的时间。当数据获取完成后,这些数据需要显示在视图(View)上,这个过程通过onrender()中的代码来实现。onrender()是Controller的第二个时态,是MVC中Controller和View建立联系的过程,负责完成对视图(View)的渲染和指定转场方式。Onready()是Controller的第三个时态,在View渲染完成后,事件绑定、DOM操作等业务逻辑都在该时态中完成;每段逻辑使用session.event包装,从而建立事件与视图block的对应关系,其中的代码都是运行在客户端的,所以可以使用前端js中的变量和函数,比如window, document, Localstorage等。但是这个技术是在用空间换时间,服务器执行了js代码,渲染了HTML,结果会一起发送给客户端,相比原来的页面,HTML内容应该会更多。由于数据绑定在onload函数中运行,而Server渲染是默认开启的,也就是这段代码是在Server端运行的,所以搜索引擎的网络爬虫是可以再次运行这段代码,获取到应用内的数据,而传统的数据只有在客户端才可以访问,如果搜索引擎要抓出应用内的数据,那就意味着他必须重建环境,在服务器端运行客户端程序,目前只有在搜索服务器上搭建移动端虚拟机,例如android虚拟机、iphone虚拟机。

6 展望

Clouda作为一个开源项目,起始于社区,并在社区的力量下不断壮大,同时受到了百度的大力支持,在这样的一种背景下,将会为Clouda开发队伍的壮大起到巨大的助推作用。目前Clouda已经拥有了众多应用,并受到开发者的青睐,包括大姨吗、超级课程表、景点通、嘀嘀打车、网上厨房等应用,均采用了Clouda的开发平台和环境,除了快速开发、开发门槛低等因素外,Clouda框架符合未来WebApp、轻应用的

应用趋势,开发者早已看重未来移动搜索在应用分发上的能力,提前拥抱Clouda。

未来越来越多的创业团队会选择Clouda进行快速研发,短期内就可以得到产品验证和反馈,大公司由于有历史原因,原有的服务都是使用java或PHP编写,数据库是mysql或者mongodb,和Clouda对接有一定的难度,Clouda是基于nodejs的,并且数据库用的MongoDB,性能和速度上没有任何问题,但凡事都有两面性,这样写死的绑定也限制了开发者采用其它数据库的可能性,如果能通过配置来修改这些绑定就比较完美了。即使数据库采用的是mongodb,原有的客户端改写了mongodb数据,如果不进行进一步开发,Clouda是无法感知数据库中数据的变化,失去了实时性这个特色。另一方面,大公司在原有的平台上已经考虑了HTML5 Web应用,从UC的网页应用数量可以看出,一般的HTML5 Web应用开发方式和传统的Android, ios, Winphone开发方式类似,web独立代码,作为第四个平台,服务器端复用,使用ajax方式请求接口,可以满足目前移动网页端的布局。传统从来都不会短期消失,习惯也不会一天改变,对于新兴的优秀技术,只要先进,能加快研发进度,实现效果,最终一定会成为一股潮流。

而作为一个优秀的WebApp框架,Clouda也许为在不久的将来在UI库的支持与扩展方面为大家带来更多的惊喜,同时,已被一些知名公司所认可。此时的Clouda对于很多开发者仍属于陌生的面孔,如何加快Clouda的推广,降低使用门槛,使这样的—个框架能够在重大项目中担任主要角色,将为其奠定在开发者心目中的定位,也将直接影响到Clouda的长期发展,希望百度能够坚持下去,有大公司支持的开源项目生命力会更顽强,有百度的大力宣传,才会有更多的开发者知道Clouda。

参考文献

- 1 <http://nodejs.org/>
- 2 <http://net.chinabyte.com/15/12697515.shtml>
- 3 <http://cnodejs.org/topic/5214df180a746c580b7966b0>
- 4 <http://www.oschina.net/p/clouda>
- 5 <http://tech.qq.com/a/20130822/018397.htm>
- 6 <http://www.donews.com/net/201308/1585797.shtml>
- 7 <http://cloudajs.org/>
- 8 <http://mobile.51cto.com/news-408664.htm>
- 9 <http://mobile.51cto.com/exp/Clouda/>
- 10 <http://mobile.51cto.com/hot-408128.htm>