

# 服务器端javascript技术研究

高原

(四川大学 计算机学院, 四川 成都 610065)

**摘要:** 如果要构建一个互联网应用, 需要前端技术和服务器端技术的支持。主流的服务器端实现技术有java、php等, 相对应的前端技术无疑是javascript。而新出现的nodeJS就是将前端和服务器端巧妙的结合起来, 将javascript部署到服务器端, 在服务器端构建高速javascript引擎。本文旨在讨论什么是nodeJS, 为什么要用nodeJS, 怎么部署使用nodeJS。

**关键词:** nodeJS; 服务器端javascript; 前端技术; 服务器; 并行

**中图分类号:** TP393.4 **文献标志码:** B **文章编号:** 1003-9767 (2012) 01-0078-02

## 引言

随着富客户端概念的流行, 很多基于javascript的大型互联网应用出现在人们面前。但是javascript是有其天生的局限性的, 比如对于浏览器环境的过分依赖, 网络因素引起的代码出错的不稳定性。尽管已经有方案来解决这些问题, 但是是以耗费额外的人力与效率为代价的。Javascript是基于客户端的引擎解析执行的, 不同的浏览器引擎会有差异, 既然这样为何不能把引擎搬到服务器端, 彻底解决对客户端的依赖性, nodeJS就是基于这种想法而生。

## 1. 什么是node

回答这个问题之前, 我们需要先了解V8引擎。V8是一个由丹麦iGoogle开发的开源javascript引擎。V8在执行javascript代码之前将其编译成了机器码, 而非字节码或者是直译它, 因此提升效能。而且, V8还使用了如内联缓存等方法来提高性能。由于这些优化, javascript程序与V8引擎的速度媲美二进制编译。

了解了什么是V8引擎后, 现在可以讨论什么是node了。概括的讲,node是一个javascript运行环境。它的实质是对google V8引擎进行了二次封装, 并且优化了一些特殊用例, 提供了替代的API, 是的V8引擎在非浏览器环境得以顺畅运行。nodeJS只是javascript运行环境, 这个学习node的基本认知。

## 2. 为什么要用node

Node公开宣称的目标是“旨在提供一种简单的构建可伸缩网络程序的方法”。目前的服务器程序有什么问题呢? 假设在Java和PHP这些服务器端语言中, 每一个连接都会在服务器上生成一个新线程, 假设每个线程需要2MB的配套内存。在一个拥有8GB RAM的服务器上, 理论上支持的最大并发连接数就是4000个。随着网站访问人数的增加, 就需要更多的服务器来满足更多并发数需求。这无疑会增加业务成本, 具体体现在服务器成本、运输成本和人工成本。与此同时会产生一个新问题, 用户可能与不同的服务器发生交互, 因此需要在每台服务器上维护一套共享资源。这就是目前整个web应用程序框架的瓶颈: 服务器所支持的最大并发连接数。

针对这一问题, node给出了自己的解决办法: 改变连接连接到服务器的方式。每个连接生成一个进程, 但是不需要给这个进程分配配套的内存块。Node不支持锁机制, 也不会阻塞I/O调用。Node将整个系统的瓶颈从最大连接数转移到单个系统的流量。这将对成本和效率的一次优化。

## 3. Node的工作原理

Node采用事件驱动、异步编程。事件驱动应该是个不陌生的概念。比如在java等编程语言里面, 我们会用到回调函数。当socket对象

满足一定的状态的时候, 我们注册的回调函数就会执行。使用node的开发人员可以根据自己的业务需要来定制相应的回调函数。同时, 这些定义的回调函数是异步执行的, 意思是, 这些回调看似在程序里是按顺序注册的, 但是他们的触发是与顺序无关, 只跟触发条件相关。同时Node采用了一系列“非阻塞”库来支持事件循环。其本质就是为文件系统和数据库之类的资源提供接口。当向文件系统发送请求的时候, 不需要等待硬盘就绪, 当硬盘准备好的时候, 非阻塞接口会发出事件来通知node。这样做最重要的优势就是, 充分利用了系统资源, 节约了任务之间的等待时间, 节省出来的资源可以为其他任务服务。在服务器开发过程中, 并发数是一个很大的问题, 传统阻塞式的函数会导致资源的浪费和延迟。通过事件机制和非阻塞编程, 会大大提高资源利用率。因此, 使用node是有其意义所在的。

同时node被设计为以单线程、单进程模式运行, 和javascript的运行方式相一致。内部单线程高效的维护一个消息队列。

## 4. node的不足

node不是万能的, 也有其不足之处。

1. 之前提到node是单线程、单进程模式, 因此node会有稳定性方面的问题。在node中所有的请求是被维护在一个高速队列中, 但是由于单线程、单进程模式, 如果队列中有一个请求发生崩溃, 那么整个队列都会受到影响。

2. Node虽然实现了将系统RAM的对性能的瓶颈转移到了系统的运算能力上, 也就是对cpu的依赖加强, 但是单进程单线程的模式是不能充分利用目前主流的多核处理器的。这也是一个很大的问题。

同时, node也针对这些问题给出了一些解决建议:

1. 使用软件进行负载均衡: 就是为每个处理器单独启动一个NodeJs进程, 每个进程都提供http服务并绑定到不同的端口, 通过一个负载均衡软件来分发不同的请求到各个进程。

2. 使用操作系统内核做负载均衡: 这个补丁是由雅虎在node-v0.1.98中提供的。改方法是创建一个进程来监听端口(如80端口), 但是不同的是这个进程不接受socket链接, 而是转发到其他子进程。OS内核本身负责进程间的负载均衡。

## 5. 结论

Nodejs虽然只有两年的生命, 但是他的成长速度是惊人的。Nodejs的出现, 彻底解决了javascript跨浏览器的难题。提供了对DOM和BOM的完美支持。并且js程序可以无缝移植到服务器端。通过巧妙的吧内存瓶颈转移到cpu瓶颈, 大大提高了服务器的并发能力。但是同时也要看到, nodejs是有其缺点的。因此, 需要根据具体的项目特点来考虑是否在项目中使用nodejs。

(下转第80页)

提供基准定时,区内采用主从同步方式建立区域同步网,区域间内采用准同步方式。

#### 4.同步网中的定时分配

在全同步网中,网同步的任务就是通过节点时钟设备,将基准定时信号传送给网内需要同步的所有通信设备,这就是定时分配。定时分配分为局内定时分配和局间定时分配。

##### 4.1局内定时分配

就是节点时钟设备(BITS)跟踪上游时钟信号,滤除传输损伤及抖动和漂动,产生高质量的定时信号,直接馈送给本局的各种通信设备(通信设备工作于外时钟方式),并将该定时信号通过定链路传递给下游时钟设备。

局内定时分配一般采用星形网结构,从BITS到被同步设备之间采用2Mbps或2MHz连接。当局内设备较多时,同类设备或组成系统的设备,一般进行同步串联,以节省同步网资源,减小定时分配的复杂性。同步串联即在系统内通过业务线向下游传递,同类设备则通过外时钟输入、输出口进行同步串接。

##### 4.2局间定时分配

就是将同步网上级节点时钟信号向下游节点时钟设备传递。局间定时传递通常采用树状结构,将来自基准钟的定时信号,通过定时链路逐级向下游节点时钟设备传递。定时链路一般有两种,即PDH定时链路和SDH定时链路。

#### 5.几种业务网的同步

同步网是支撑网,其功能是为各种业务网提供同步定时,业务网是同步网的用户。SDH传输网有别于其它业务网,它既是同步网的用户,又是同步网的载体。

##### 5.1传输网的同步

目前传输网主要有PDH传输网和SDH传输网,或以SDH传输网为主的二者混合网。PDH传输采用码速调整技术,网同步的实现比较简单,对系统同步和设备同步也没有特别要求。SDH网称为同步传输网,根据其工作原理,需要有严格的网同步。

SDH网从一个点或多个点接受同步网定时供给单元(SSU)提供的定时,并按一定的方式将定时信息在网内传递。定时传递方式分为:全程定时、分段定时和主备用定时等。SDH的设备定时,一般以数字交叉连接设备DXC作为定时分配中心,局内没有DXC时,选择一

个线路复用设备ADM作为定时分配中心。

##### 5.2程控交换网的同步

程控交换是一种电路交换,是通过时隙交叉连接实现交换功能,因此需要严格的网同步。根据在网络中所处的地位不同,交换机采用不同级别的时钟。一般长途交换中心采用二级时钟,本地交换中心采用三级时钟,其分别相当于G.812中的转接时钟和本地时钟。

##### 5.3ATM网、帧中继网FR、分组交换网和IP网的同步

ATM网、帧中继网FR、分组交换网(X.25)中,数字信息在设备内部采用“存储转发”的分组交换方式,各端口之间无须时隙一一对应,速率也可以不同,具有异步传递特性。所以,不需使终端及业务与网络保持同步,一般不需要网同步,只要求设备同步。设备同步包括外同步和业务同步两种同步方法,实际中多采用业务同步。

##### 5.4 DDN网的同步

DDN网是以DDN节点设备和数字传输链路构成的数字数据网,节点设备本质上是电路交换,相当于“半永久性连接”交换机,而且承载的主要是数据业务。所以,DDN网需要严格的网同步,否则会引起数据丢失。

DDN节点设备有两种同步方式:一是通过外时钟接口,直接接受同步网定时;一是从上游业务码流中提取定时。一般首选从业务码流中提取定时。

#### 参考文献:

- [1]程根兰 编著: 电信技术丛书--《数字同步网》 人民邮电出版社 2001年4月.
- [2]邓忠礼 赵晖 编著: 《光同步数字传输系统测试》 人民邮电出版社 1998年5月.
- [3]魏珍等编 测控通信技术书--《时频技术应用》 西安卫星测控中心 2003年8月.
- [4]《中国有线电视》, 2003年22期.
- [5]《光通信技术》, 2006年第10期.

作者简介: 李光明(1959-),男,陕西西安人,毕业于西安电子科技大学,本科,现为西安政治学院网络管理中心高级工程师(副教授级)。

(上接第78页)

#### 参考文献:

- [1] Tom Hughes-Croucher. Node. O'Reilly Media[M]. 2011-12-31.
- [2] Mike Cantelon, TJ Holowaychuk. Node.js in Action[M]. Manning.
- [3] Mick Thompson. Getting Started with GEO, CouchDB, and Node.js. O'Reilly Media[M]. 2011-8-3.
- [4] Brett McLaughlin. What is Node. O'Reilly Media[M]. 2011-7.
- [5] David Herron. Node Web Development. Packt Publishing[M].

2011-8-10.

- [6] taobao ued. node.js调研与服务性能测试[Z]. <http://www.tbdata.org/archives/1285>.
- [7] ibm developer. Node.js究竟是什么[Z]. <http://www.ibm.com/developerworks/cn/opensource/os-nodejs/>.

作者简介: 高原(1986-),男,辽宁调兵山人,学历:硕士,研究方向: web前端技术,单位: 四川省成都市四川大学。