

基于 B/S 架构的 NBI 实验数据发布系统^①

张睿^{1,2}, 胡纯栋¹, 盛鹏¹, 赵远哲¹, 杨振汉^{1,2}, 崔庆龙¹

¹(中国科学院等离子体物理研究所, 合肥 230031)

²(中国科学技术大学, 合肥 230026)

摘要: 东方超环中性束注入 (EAST- Neutral Beam Injection, EAST- NBI) 系统的实验数据发布方式众多, 因开发语言不同难以集成, 且多使用客户端/服务器 (Client/Server, C/S) 模式进行设计, 服务器只对单一局域网内客户端提供服务, 外网客户不能获得数据服务. 针对这一现状, 设计了基于浏览器/服务器 (Browser/Server, B/S) 架构的实验数据发布系统, 满足多用户通过外网获取实验数据的需求. 浏览器端使用 HTML+CSS+JavaScript 进行开发, 服务器端使用 Java 语言开发. 系统包含最新数据自动显示和历史数据查询两种模式, 最大程度上集成现有发布项目. 数据基于 WebSocket 协议进行传输, 处于全双工状态下进行交互, 这样既实现了新数据的自动推送, 又为历史数据的请求和回应提供通道. 同时, 系统还提供评论功能. 该系统已投入到 EAST-NBI 实验中, 获得很好的实验效果.

关键词: 中性束注入; 数据发布; WebSocket 协议; 推送; 查询

引用格式: 张睿, 胡纯栋, 盛鹏, 赵远哲, 杨振汉, 崔庆龙. 基于 B/S 架构的 NBI 实验数据发布系统. 计算机系统应用, 2017, 26(11): 109-113. <http://www.c-s-a.org.cn/1003-3254/6053.html>

Experimental Data Publishing System for NBI Based on B/S Architecture

ZHANG Rui^{1,2}, HU Chun-Dong¹, SHENG Peng¹, ZHAO Yuan-Zhe¹, YANG Zhen-Han^{1,2}, CUI Qing-Long¹

¹(Institute of Plasma Physics, Chinese Academy of Sciences, Hefei 230031, China)

²(University of Science and Technology of China, Hefei 230026, China)

Abstract: Experiments on the traffic information auto-extraction and mixed traffic travel schemes auto-creation system show that the system has high precision and is adaptive to web pages in different domains with different structures. Experimental data of EAST-NBI (EAST- Neutral Beam Injection) System are published in several pieces of software. It is difficult to integrate them because of the difference in languages. Besides, these pieces are based on C/S (Client/Server) architecture. The server only offers service to specific client in LAN (Local Area Network). Clients on the internet can't get access to the data. To meet the need via the internet, a data publishing system based on B/S (Browser/Server) is designed. The browser side is developed by HTML+CSS+JavaScript and the server side is developed by Java. The system consists of latest data publishing mode and historical data query mode, and integrated existing data items. The data are transmitted based on WebSocket protocol, which enables a full-duplex channel for automatic push of new data and request and response for historical data. Meanwhile, a comment function is provided in this system. The system has been put into the EAST-NBI experiment and got positive results.

Key words: NBI; data publishing; WebSocket protocol; push; query

EAST 装置是我国自行设计研制的国际首个全超导托卡马克装置. NBI 系统是加热效率最高, 物理机制

最清晰的托卡马克进行加热的方式之一^[1]. 在近一年的实验中, EAST-NBI 系统的两条束线均投入实验并取

① 基金项目: 国家国际科技合作专项 (2014DFG61950)

收稿时间: 2017-03-01; 修改时间: 2017-03-16; 采用时间: 2017-03-20

得良好的实验成果,为 EAST 装置实现高参数稳态运行发挥了重要作用.在实验过程中,运行人员通过分析实验数据获得系统状态,注入能量等关键信息,处于外网的人员也通过分析实验数据,对实验提出优化的意见和建议.所以,对 NBI 实验进行实验数据发布尤为重要.

1 技术现状及研发目标

NBI 系统的实验为脉冲式,每次放电过程称为一次实验,时长 0 s-100 s 不等,实验之间相互独立.目前,系统的实验数据发布依赖于多个与数据服务器交互的客户端程序,这些程序开发语言不同,难以对数据项目进行集成.且数据服务器因早期开发的局限,仅能对这些客户端提供局域网内的一对一的服务,每增加一个客户端都需要对代码进行修改,这种方式的维护成本非常高,使实验数据的发布难以推广.

针对这一现状,设计了 B/S 架构的实验数据发布系统.用户可以在外网通过浏览器获得数据服务.每次实验结束后,数据可以自动推送到客户端.不同的数据项目也有不同的展现形式,如系统最关键的信号高压电压、高压电流以及 EAST IP、EAST Ne 信号均采用折线图的形式展现,方便用户直观地查看束流参数;其他子系统的电源统计结果、光谱诊断结果等数据使用常量展示,提供关键参数统计结果,压缩视图空间使两条束线的实验结果在一个页面上得到发布.同时提供评论功能,获得权限的用户可以在评论区对实验结果进行评论并广播到所有在线用户;所有数据项的历史结果也可以通过系统查询.满足运行人员及关注运行 NBI 实验的相关人员的需求.

2 系统概述

本节从硬件部署和软件设计两方面进行阐述.

2.1 硬件部署

NBI 控制系统是 NBI 系统的重要组成部分,对整个 NBI 系统的实验运行过程进行全面的监控报警和连锁保护,实现数据采集处理与网络通讯.控制系统网络分为 4 个子网,远程监控网、服务器总控网、现场测控网及视频监控网^[2].Web 服务器部署在控制系统的服务器总控网,与同属于一个子网的总控服务器和数据服务器进行交互.交互的内容包括:实验状态信息,重要参数采集结果及统计信息等.同时,web 服务器又

经过防火墙与外网相连,为外网用户提供服务.拓扑图如图 1 所示.

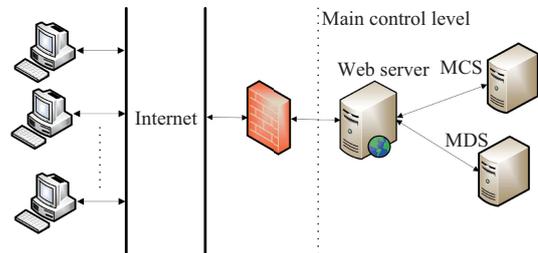


图 1 实验数据发布系统硬件部署拓扑图

2.2 软件设计

软件设计使用 B/S 架构,软件服务器使用 Apache Tomcat,服务器端使用 Java Web 技术进行开发;客户端使用 HTML+CSS+JavaScript 进行开发.

软件部分的核心需求是为客户端提供实时的实验数据推送和历史数据查询服务.在实时数据的推送中,为了解决浏览器客户端的 HTTP 请求无状态无连接的特点,通用的做法是使用轮询方法,这种方法的缺点是传输效率低,占用服务器的资源大.故此处选择 WebSocket 协议作为通信协议^[3].在 WebSocket API 中,浏览器和服务器只需要做一个握手的动作,然后,浏览器和服务器之间就形成了一条全双工通信通道.两者之间就直接可以数据互相传送.

软件设计分为 3 个部分:Web 服务器与控制系统交互的辅助软件部分、浏览器端部分和服务器端部分.总体设计为:辅助软件和浏览器客户均作为服务器的客户端,使用 WebSocket 协议进行交互,浏览器在获得辅助软件的实时消息后,将消息推送给在线的浏览器客户.如果浏览器客户选择历史数据查询模式,那么请求和回应消息也通过 WebSocket 消息通道进行交互.请求的数据将通过服务器程序向数据服务器请求获得.

2.2.1 辅助软件设计

辅助软件使用 C/C++ 语言进行开发,其作用有两点:一是收集实验状态信息,如炮号、放电模式、状态参数信息、关键参数的统计结果等;二是转换通信协议.其工作流程如图 2 所示.

首先,辅助软件需要作为客户端,向 MCS 注册^[4],成为其合法用户.在实验过程中,MCS 会向辅助软件发送实验状态信息,之后,辅助软件将 MySQL 数据库的实时状态表单内容更新为这些数据.最新的实验数

据会成为热点信息,方便 Tomcat 容器中的服务器端程序初始化时使用.同时,在 Tomcat 服务器端程序运行期间,辅助软件会将数据会以 WebSocket 协议转发,供服务器端程序对浏览器客户广播使用.

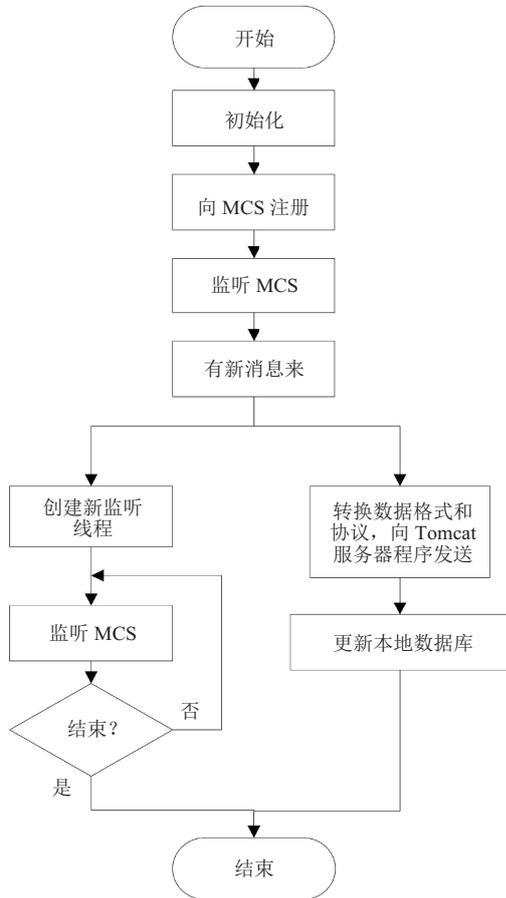


图2 辅助软件的工作流程图

MySQL 数据库存储最新的实验信息,不同的项目,如实验炮号、实验模式、关键参数统计值、光谱诊断信息等存储进不同的表单,以关键参数统计值为例,其表单设计如表1所示.

表1 关键参数统计值表单设计

Field	Type	Null	Key	Default
ChName	Varchar(32)	No	PRI	NULL
SigUnit	Varchar(32)	Yes		NULL
MaxVal	Float	Yes		NULL
AvrVal	Float	Yes		NULL
MinVal	float	yes		NULL

表1中,ChName 字段表示信号名称,类型为最大长度为 32 字节的字符,非空主键;SigUnit 字段表示信号单位,类型为最大长度为 32 字节的字符;MaxVal、

AvrVal、MinVal 三个字段分别表示该信号在上次实验过程中统计得到的最大值、平均值、最小值,类型为浮点型.

MCS 与辅助软件之间的数据交互格式如图3所示.

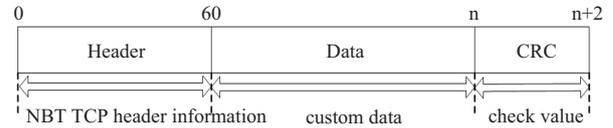


图3 NBI 控制系统数据交互格式

第一部分为头部信息,占用 60 个字节,包括源地址号、目的地址号、服务号、命令号、数据长度信息,以此区分数据的发送者、接收者、数据内容和数据长度;第二部分是定制数据,根据不同的服务号和命令号确定数据格式,数据长度由头部数据中的数据长度信息确定;第三部分是循环校验值,采用控制系统规定的校验算法计算获得,占用 2 个字节,用于对数据的完整性和准确性进行确认^[5].

辅助软件与 Tomcat 服务器端程序交互格式采用 JavaScript Object Notation (JSON) 格式,它易于人阅读和编写,同时也易于机器解析和生成,是一种轻量级的数据交换格式.以一则 JSON 格式消息为例: {“content”: “acpdata”, “role”: “Middleware”, “acptime”: “0”, “accvvalue”: “0”}.

基础数据项目包括 content 和 role,分别表示数据内容和角色,如上例中,数据内容是高压电源数据“acpdata”,角色是辅助软件“middleware”.在此基础上可以根据数据内容的不同定义自定义的数据项目,如 accpdata 项目下的数据项为“acptime”和“accvvalue”.

2.2.2 服务器端实现

JDK1.7 以上版本即提供对 WebSocket 应用的实现^[6].在 WebSocket 实现的核心类 MonitorDataWebSocket 类中,以注解的形式对 URL 以及 onopen、onClose、onMessage、onError 四种方法进行定义.因涉及到多线程操作影响线程安全的问题,分组用户使用 ConcurrentHashMap 类型的类成员进行存储. ConcurrentHashMap 类型引入了一个“分段锁”的概念,即把 Map 分成了 N 个 Segment, put 和 get 的时候,都是现根据 key.hashCode() 算出放到哪个 Segment 中,每个 Segment 有自己的锁,只要多个修改操作发生在不同的段上,它们就可以并发进行.这大大提高了运行效率,减少了因线程安全考虑而引起的开销.

WebSocket 的业务需要使用的工具类有: AgentMessageDealer、ClientMessageDealer、SqlQuery、PubService. AgentMessageDealer 类处理来自辅助软件的消息, ClientMessageDealer 类处理来自客户端的消息, SqlQuery 类封装了对数据库的操作, PubService 类定义了广播方法.

服务器端程序的运行流程为: 初始化, 通过查询数据库得到最新的实验数据. 等待客户端连接, 如果辅助软件客户端来连接, 获取新的实验数据, 同时向数据服务器请求关键参数的采集数据, 将这些数据广播给需要对应服务的用户分组, 并将这些数据存储在内存中, 方便调用; 如果是浏览器客户端来连接, 根据其请求的内容将该连接归组, 如果得到用户对历史实验数据的请求, 则向数据服务器请求响应数据, 推送给用户.

2.2.3 浏览器端设计

用户界面借鉴 EAST-NBI 系统监控屏幕的数据排布, 使用 Bootstrap 框架开发. Bootstrap 来自于 Twitter, 是目前很受欢迎的前端框架, 它基于 HTML、CSS、JavaScript, 简洁灵活, 使得 Web 开发更加快捷^[7]. 借助于 Bootstrap 的栅格系统, 开发页面布局如图 4 所示.

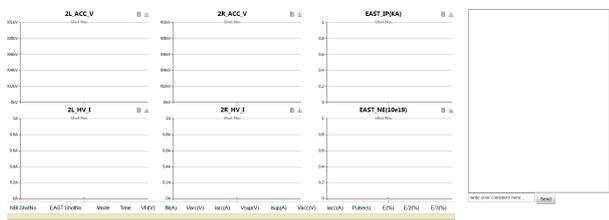


图 4 用户界面设计图

页面主要的数据表现形式是表格和折线图, 折线图使用 ECharts 进行开发, ECharts 是基于 JavaScript 的图形库, 支持多种图表实现, 如折线图、甘特图、柱状图、饼图等. 同时 ECharts 具有自适应的特性, 在移动端有很好的表现^[8].

浏览器端对 WebSocket 协议的实现使用 JavaScript 接口. 浏览器端和服务端的数据交互使用 JSON 格式, 它易于机器解析和生成, 是一种轻量级的数据交换格式. 在 onmessage 方法中加入逻辑代码即可实现对服务器消息的解析和使用.

无论是当前实验数据的展示模式还是历史实验数据的查询模式, 数据均通过 WebSocket 进行交互. 处理的流程为: 当有消息到来时, 根据当前的模式对消息进

行区分处理. 如果是最新实验数据显示模式, 则将新消息解析后显示在人机界面上, 更新 cookie 数据; 如果是历史实验结果查询模式, 则将新消息中的炮号与查询的指定炮号进行对比, 如果一致则显示, 如果不一致, 则将最新数据更新到 cookie 数据, 以便用户进行模式切换时进行显示.

用户评论是与炮号对应的, 用户在提交评论后, 服务器会将评论存储进数据库, 存储成功后, 服务器会将该条评论进行广播, 浏览器端接到广播的评论消息会与当前炮号对比, 一致的话会将评论显示出来, 否则不予显示.

3 系统测试

测试分为两个部分: 包括功能测试和压力测试.

3.1 功能测试

在功能测试中, 分别测试了辅助软件客户连接和普通用户连接两种情况.

辅助软件用户连接时, 服务器可以正确识别辅助软件用户; 在辅助软件用户发送来新消息后, 服务器端可以正确解析消息并启动广播.

浏览器用户连接时, 服务器可以正确识别用户请求, 将用户归组, 并向其推送最新实验结果进行页面初始化. 客户端在获得新消息后可以刷新页面, 由于页面不存在跳转所以也不会出现闪烁问题. 图 5 展示了二号线在最新实验结果模式下, 第 15946 炮的实验结果. 页面在采集结束之后即得到刷新.

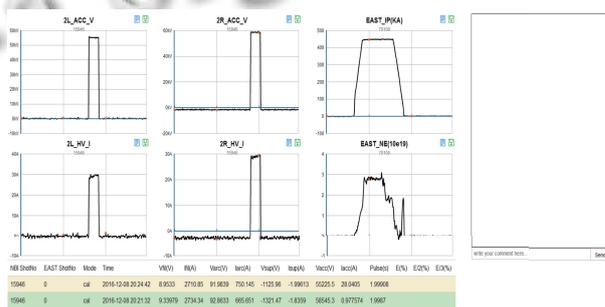


图 5 第 15946 炮实验结果显示

浏览器用户切换到历史数据显示模式, 可通过点击 prev 和 next 按钮对历史炮号下的实验数据进行查询, 经证实, 该服务器可对用户请求作出正确响应, 客户端也根据数据刷新页面. 通过点击“latest”按钮, 可回到当前实验数据页面. 浏览器用户可对实验进行评论, 之后可以广播给所有在线用户.

3.2 压力测试

压力测试也称为强度测试、负载测试。这里使用 ApacheBench 测试^[9]。在测试中, 发送 5000 条请求, 其中每次并发量为 1000, 测试 MonitorResult.jsp 服务页, 即实验结果显示页面。测试结果表明: 所有请求在 218 ms 内完成, 在所有并发的请求中, 每个请求所需的时间为 0.186 ms, 完全符合用户需求。

4 结束语

本文介绍了一套基于 B/S 架构的 NBI 实验结果发布系统, 使用 WebSocket 协议作为浏览器端与客户端的数据传输协议, 使用全双工通道进行信息交互。系统同时具备自动显示最新实验结果和查询数据结果两种模式。测试表明, 方案切实可行。本系统已在 EAST-NBI 实验中投入使用, 运行效果良好。

参考文献

1 Hu CD, Team NBI. Conceptual design of neutral beam

injection system for EAST. Plasma Science and Technology, 2012, 14(6): 567–572. [doi: 10.1088/1009-0630/14/6/30]

2 Sheng P, Hu CD, Song SH, *et al.* The NBI control system for the EAST. Journal of Fusion Energy, 2014, 33(5): 529–534. [doi: 10.1007/s10894-014-9700-y]

3 李代立, 陈榕. WebSocket 在 Web 实时通信领域的研究. 电脑知识与技术, 2010, 6(28): 7923–7925. [doi: 10.3969/j.issn.1009-3044.2010.28.016]

4 Zhao YZ, Hu CD, Sheng P, *et al.* Study of mechanics control server application software on EAST-NBI. Applied Mechanics and Materials, 2012, 214: 562–567. [doi: 10.4028/www.scientific.net/AMM.214]

5 Wu DY, Hu CD, Sheng P, *et al.* Design of main control console software in EAST neutral beam injector's control system for the first beam line. Journal of Fusion Energy, 2013, 32(5): 552–556. [doi: 10.1007/s10894-013-9612-2]

6 http://www.ibm.com/developerworks/cn/web/1112_huangxa_websocket/.

7 <http://v3.bootcss.com/css/>.

8 <http://echarts.baidu.com/echarts2/>.

9 <http://httpd.apache.org/docs/2.4/programs/ab.html>.