

基于 Ionic 开发平台的数据库创建

李雅雯^{1,2} 宁哲^{2,3} 钱森^{2,3} 朱纳⁴ 高峰^{2,3} 江晓山^{2,3} 张银鸿^{2,3} 蔡浩¹

1 (武汉大学 物理科学与技术学院 武汉 430072)

2 (中国科学院高能物理研究所 北京 100049)

3 (核探测与核电子学国家重点实验室 北京 100049)

4 (陕西科技大学 西安 710021)

摘要 针对多个大科学工程中各种实验器件,如光电倍增管(Photomultiplier Tube, PMT)、读出电子学电缆等性能批量测试数据管理的实际需求,开发了基于 Ionic 平台的相关数据库。该数据库利用 Angular.js、Ionic 和 HTML5 为前端,以 Node.js 和 Express 为后端服务,生成的 MySQL 数据库可运行于手机客户端与 Web 端的数据管理系统。以同一套代码完成跨平台运行,实现在不同浏览器(Chrome、IE、Safari)、不同系统(Windows Phone、Android、IOS)上运行的具有完整功能的数据库软件。具备对实验项目中部分测试数据进行跟踪、记录和查询等功能,为项目的进行提供有力的保障。

关键词 Ionic, 数据库, Angular.js, Express, MySQL

中图分类号 TL822⁺.6

DOI: 10.11889/j.0253-3219.2017.hjs.40.040401

A database with the Ionic platform

LI Yawen^{1,2} NING Zhe^{2,3} QIAN Sen^{2,3} ZHU Na⁴ GAO Feng^{2,3} JIANG Xiaoshan^{2,3}
ZHANG Yinhong^{2,3} CAI Hao¹

1(School of Physics and Technology, Wuhan University, Wuhan 430072, China)

2(Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049, China)

3(State Key Laboratory of Particle Detection and Electronics, Beijing 100049, China)

4(Shaanxi University of Science and Technology, Xi'an 710021, China)

Abstract Background: Large-scale scientific facilities need bunch test for their key devices, and need the special database to manage the test results. **Purpose:** This study aims at the practical requirements of data management for experimental devices, such as photomultiplier tube (PMT) and readout electronics cable, etc. and the design of related database based on the Ionic development platform for scientific facility. **Methods:** Angular.js, Ionic, HTML5 are employed to implement the front-end, whilst Node.js, Express are used for the back-end services. Combine with the MySQL, a data management system is generated to run across various platforms such as the mobile client and the Web side. **Results:** The database software can execute in different browsers (Chrome, IE, Safari) via various operating systems (Windows Phone, Android, IOS) with the same code. **Conclusion:** With functionality of track, record and query the variety of test data for all experimental devices, this database provides effective protection for the conduct of the experimental project.

Key words Ionic, Database, Angular.js, Express, MySQL

国家自然科学基金(No.11175198、No.11475209、No.11611130020、No.11675205)资助

第一作者: 李雅雯, 女, 1992 年出生, 2015 年毕业于武汉大学, 现为硕士研究生, 现从事数据库的创建工作

通信作者: 钱森, E-mail: qians@ihep.ac.cn; 宁哲, E-mail: ningzhe@ihep.ac.cn

收稿日期: 2016-12-27, 修回时间: 2017-02-03

Supported by National Natural Science Foundation of China (No.11175198, No.11475209, No.11611130020, No.11675205)

First author: LI Yawen, female, born in 1992, graduated from Wuhan University in 2015, master student, now engaging in the creation of the database

Corresponding author: QIAN Sen, E-mail: qians@ihep.ac.cn; NING Zhe, E-mail: ningzhe@ihep.ac.cn

Received date: 2016-12-27, accepted date: 2017-02-03

大科学装置及工程往往会对某种型号的器件或者探测器进行批量使用。为此,在项目研发阶段,除了对某种探测器的性能进行详细的研究测试,还需要对大量的同款探测器进行批量测试和性能刻度,不但保证每个器件都符合该项实验的要求,还要保证探测器的性能均匀性和一致性。

北京谱仪(Beijing Spectrometer, BESIII)缪子探测器(Muon Chamber, MUC)的核心部件为阻性板探测器(Resistive Plate Counter, RPC),在其研发阶段,无论是裸室 RPC,还是探测单位模块,都进行了严格的批量测试,通过使用专用数据库记录分析测试数据,为 MUC 的成功调试和运行维护提供基础数据支撑^[1]。同样,在大亚湾中微子实验(Daya Bay Reactor Neutrino Experiment, DayaBay)的建造过程中,研究人员对使用的 20.32 cm 的光电倍增管(Photomultiplier Tube, PMT)和 RPC 进行了批量测试,使用基于 MySQL 的数据库统一管理测试数据^[2],不但实现对合格 RPC 和 PMT 的筛选,还为探测器的离线刻度提供原始数据支持。

在建的江门中微子实验(Jiangmen Underground Neutrino Observatory, JUNO)^[3]和高海拔宇宙线观测站(Large High Air Altitude Shower Observatory, LHAASO)^[4]实验,将会使用上万只 50.8 cm 的大面积 PMT^[5],为了保证工程的顺利进行,需要对每只 PMT 进行严格的性能标定,从裸管到封装分压器前后,安装前后都需要详细而繁杂的刻度测试。在 JUNO 中, PMT 产生的模拟信号被紧贴其后的读出电子学转换成数字信号,通过将近 100 m 的超 5 类耐水压特种网线传输到岸上的路由器。因此整个实验不仅用到两万只 PMT,还要使用两万套读出电子学和两万根超 5 类特种网线。

每只 PMT 和每根电缆,从生产到安装完成的过程中都要经过多次测量,这期间保存了大量关于生产部件性能状态的实验数据。为对这些实验数据进行跟踪、记录、查询,实现合作方之间的数据共享、信息交流,非常有必要创建一个数据库系统实现相关的功能。

1 数据库架构的选择

数据库系统主体分为前端界面、后端服务和数据库。鉴于进行数据库系统开发的资源限制,希望能够以一套代码完成跨平台的后端开发,实现前端页面在不同浏览器、不同系统的运行,这样也有利于将来的维护。

1.1 常规数据库架构

在定义该数据库系统结构之前,我们对以往成功的类似功能数据库的结构: BESIII 以及 DayaBay 的质量控制数据库采用了 MySQL+PHP+Apache 的组合。

其中超文本预处理器(Hypertext Preprocessor, PHP)实现与数据库通讯的后端服务, Apache 作为 Web 应用服务器解释后端服务 PHP。其结构如图 1 所示。这些软件都可免费获得,不同操作系统的数据库开发者都能运用^[1]。在 BESIII 数据库建造时期,这种组合也被称为开发数据库的黄金组合,可以较好地实现人机交互,操作方便。特别是 PHP 可以方便地嵌入到 HTML 语言中,又可以很好地与后端 MySQL 数据库实现交互,可以在 Web 端实现对实验数据及时更新、查询和显示等功能。

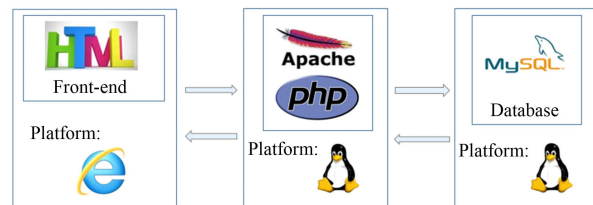


图 1 BESIII、DayaBay-RPC 数据库系统架构图
Fig.1 The architecture diagram of database used in RPC for BESIII/DayaBay.

近些年来随着智能手机的普及和 HTML5 标准的落地,为充分利用当前科技发展成果来实现更丰富和更方便的用户体验,满足用户的不同需求,迫切需要采用一种新的数据库架构。我们希望实现开发混合移动应用程序,而不仅限于 Web 端数据库界面的开发。而混合移动应用程序只是一个普通的移动优化网站,用 CSS、HTML 和 JavaScript^[6]编写,在 Webview 上显示。这样只需要编译一个应用程序,在大多数情况下,几乎不用修改源代码就可以在 Android、IOS 系统上运行^[7]。

1.2 新设计数据库架构

通过调研工业界当前的技术发展应用,我们发现随着前端技术的发展,涌现出了许多开源的混合移动应用开发框架,例如 React-Native、Ionic 等^[8],都提供了跨平台开发的功能。

React-Native 正如其名称一样,目的是构建真正的 native 应用,用户体验高于 HTML,开发效率较高,开发完全由 JavaScript 和 React 来完成。但是这个框架不太适合 Web 开发的新手,当官方封装的控件、应用程序接口(Application Programming Interface, API)无法满足要求时,就必然需要应用一

些 native 方法去扩展, 扩展性能远不如 Web, 也不如直接写 native code, 发展还不成熟。目前很多用户界面(User Interface, UI)组件只有 IOS 的实现, Android 系统中的应用则需要用户实现, 而且文档不够完整, 学习起来较困难^[7]。

Ionic 是基于 JavaScript 开发的, 基本上可以实现 IOS 和 Android 共用代码, 纯 Web 思维、开发速度快、简单方便, 一次编码全域运行。基于 phone-gap 的编译平台, 可以实现编译成各个平台的应用程序。开发文档比较完整, 开发者社区较为成熟, 遇到问题容易找到解决的方案。综上, 我们选择 Angular.js+ Ionic+phone-gap/cordova 的方案作为数据库前端的开发工具。

既然前端是基于 JavaScript 开发, 因此希望后端的开发也能采用 JavaScript。通过调研发现, Node.js 是一个开放源代码、跨平台的、可用于服务器端和网络应用的运行环境。Node.js 实际上就是运行在服务端的 JavaScript, 使之具备了构建基于

JavaScript 的高并发的 Web Application 的能力, 又不会因多线程本身的特点而带来麻烦。Express 是一种保持最低程度规模的灵活 Node.js Web 应用程序框架, 为 Web 和移动应用程序提供一组强大的功能和丰富的 HTTP 工具^[9]。

利用 Angular.js、Ionic 和 HTML5 为前端, Node.js 和 Express 为后端服务以及 MySQL 数据库能够生成可运行于手机客户端与 Web 端的数据管理系统, 其结构如图 2 所示。表 1 给出了两种数据库架构特性的关键特性对比。

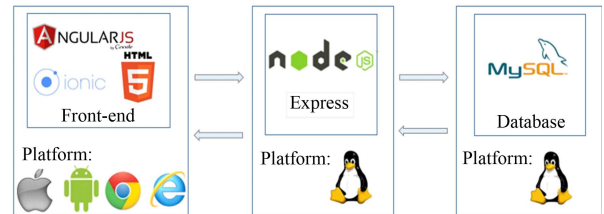


图 2 基于 Ionic 开发平台数据库系统架构图
Fig.2 The architecture diagram of the database within Ionic.

表 1 两种数据库架构对比
Table 1 Contrast of the two database architectures.

特性 Characters	PHP+Apache	Ionic+Angular.js+Node.js
平台 Platform	Web	Web+PC+APP
开发语言 Language	PHP	JavaScript
中间件 Middleware	较少 Less	丰富 More
扫描条码 Scan the barcode	×	√
自动更新 Auto-update	×	√
三维成像 3D imaging	×	√

2 数据库的创建

该数据库系统以 Ionic 开发平台为基础, 以同一套代码完成跨平台运行, 实现在不同浏览器、不同系统上的运行。整个开发过程始终以同一套程序语言 (JavaScript) 开发, 拥有友好的交互页面。MySQL 数据库部署在 Linux 系统的服务器上, 可以使用数据库浏览软件 HeidiSQL^[10]或者 Navicat^[11]实现远程连接并查看数据库中的内容。

2.1 接入文档数据库

为了能更好地面向用户, 使其无需二次注册登录该数据库系统, 通过采用如图 3 所示的单点登录方式, 用户采用登录文档数据库的用户名和密码即可登录本数据库。具体的实现方式是采用 Shibboleth 接入方式, 我们需要把服务器配置成一个 SP (Shibboleth Service Provider), 再将我们的 SP 接入到 IDP (Identity Provider) 服务器管理系统即可。

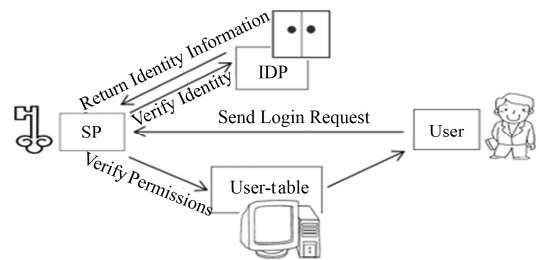


图 3 用户登录逻辑示意图
Fig.3 The logic of user login the database.

如图 3 所示, 当用户输入用户名及密码申请登陆时, 首先向服务器中 SP 发送请求, SP 接收到登录请求信息后发送指令向 IDP 请求验证; IDP 收到 SP 发送的验证信息, 将用户名和密码与存储的用户信息进行验证, 通过后将验证通过的信息再返回到 SP, SP 收到验证通过的信息再与保存有用户不同权限的用户表 User-table 进行比对, 使得该用户获得相应的登录权限。

2.2 实验数据的上传

实验数据生成后需要操作者上传到数据库后才能供使用者调用查看。因为数据文件较多, 我们无法/也没必要通过手动上传的方式将实验数据键入到 MySQL 数据库中, 这将极为低效而且容易出现人为因素的数据错误。可以通过编程方式实现表格文件的上传, 同样该方式还可实现大批量上传, 方便快捷, 同时最大程度地保证数据的正确性。

还可以通过手机 APP 扫描基于实验数据生成的二维码, 实现数据的上传, 这将更加方便快捷。包含实验数据的二维码如图 4 所示。因为服务器部署在远端, 该方式将会很好地解决用户因电脑存在保密管理问题或无法连接网络造成数据无法上传至服务器数据库的问题。

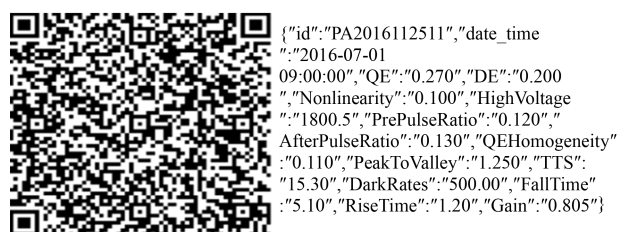


图 4 包含实验数据的二维码, 右侧为其内容
Fig.4 The two-dimensional code contains the experimental data and its content is on the right.

2.3 热更新功能

传统上实现相关的软件功能应用, 一般需要先生成.exe 文件, 用户下载该文件并安装, 每次应用更新都需要重新生成新的.exe 文件以供用户安装。热上传功能是指开发人员不需要将更新过的程序重新打包上传, 仅需下载更新补丁就可完成软件功能的升级。

要实现热更新功能只需要安装热更新插件, 即 cordova 的 cordova-hot-code-push 插件。首先官网下载此插件, 然后安装生成检测配置文件并更新插件^[12]。配置完成后, 运行代码或打包 APP, 修改需要更新的代码, 在手机上打开应用实现了动态更新。

开发热更新功能将为用户提供极大的便利, 用户无需重新下载安装 APP 就可在第一时间获取最新版本的 APP 并节省流量。程序员也可以及时得到用户的反馈, 这样此数据库应用软件将会更加地满足多种用户需求。

3 数据库的功能

数据库需要实现的基本功能: 用户登录与分级、

数据的查询与录入、数据统计、部件状态跟踪以及各系统关系映射。

3.1 用户登录与分级

为保证数据安全, 数据库 APP 首先需要登录页面。同一项目中合作组成员只需要使用文档数据库中的用户名和密码即可完成登录, 无需重新注册。

在大科学工程中, 每个科研项目、每个模块都有众多的国内外研究人员参与。为保证每个模块只有相关人员可以查看, 我们对用户的权限加以限制。用户的权限分为三级: 管理员、实验员和普通用户。管理员拥有除程序员以外的最高权限, 不仅可以对所有的数据实现增删改查, 还可以实现给其他的用户分配权限; 实验员是参与某一实验的人员, 能对某模块数据实现增删改查, 实现对测试数据的监督, 但不包括管理其他用户权限的功能; 普通用户仅有查看部分实验数据的功能。给用户添加不同的权限, 这样可以最大限度地保障数据的安全。

3.2 数据的查询与录入

数据库 APP 最主要的功能就是实现实验中测试数据的录入和为用户提供丰富的查询功能。要实现数据的查询, 我们在数据库的主页面设有相关查询的链接, 包括不同模块信息的分组, 另外还包括扫描条形码查询按钮和按产品编号查询按钮。

写入数据表单的数据都是基于测试的原始数据经分析后得到的结果, 而不是原始数据, 所以不需要查询后再进行处理。当需要处理的数据量增大后, 批量读取 MySQL 多行 (比如根据 BESIII 预计有两万行, 对应两万支光电倍增管), 不到 1 s 的时间就能够把图画出来。实际上无论是查询还是插入数据, 要处理的数据量远低于 MySQL 数据库每秒能够处理速度上限 (几十万次), 而且我们使用 i7CPU 的服务器, 所以完全不用担心因数据量增大而造成数据库性能下降的问题。

3.3 数据统计

数据库 APP 除了有上传实验数据以及查看测试数据和图片的功能, 还具有数据统计功能。在查询测试数据的过程中, 我们可以选择某一类测试数据为变量, 例如光电倍增管的上升时间(Rise time)、探测效率(Detector efficiency, DE)等, 查看在测试时间段内的统计波形图和统计直方图。

统计波形图可以显示测试数据的实时曲线, 直观显示测试参数随着测试时间的变化趋势, 如图 5 所示。每天实际测试的样管数据为 32 只, 所以将一

只老化用 PMT 其中 33 天的老化数据当作 33 只 PMT 一天的数据导入数据库来检验数据库这部分功能。

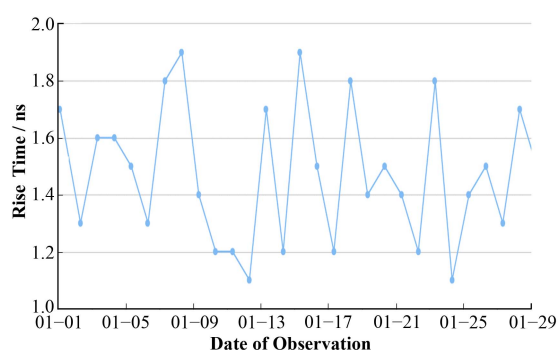


图 5 统计波形图
Fig.5 Statistical waveform.

统计直方图可以将测试的大量历史数据进行分析，从空间上判断产品的生产过程是否稳定，评估其质量特性。按照统计学原理，用史特吉斯(Sturges)公式和测定数 n 来确定区间的数量，其中区间数量为 $1+3.3\log(n)$ ，由此得到区间宽度。如此生成的一组直方图如图 6 所示。

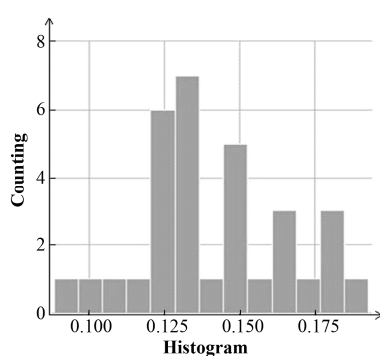


图 6 统计直方图
Fig.6 Statistical histogram.

3.4 部件状态跟踪

为方便实验人员及时、快捷地对产品部件的状态进行更新，我们利用条形码实现类似快递投递功能的部件状态跟踪功能。相关人员使用手机摄像头扫描产品部件上对应的标签完成相应状态的更新，而不需要再进行手动输入，从而更加方便快捷，提高实验人员的效率。因为每个产品部件都有其唯一的编号，通过此功能可以对某一产品部件实现从出厂到安装完成整个过程的记录、跟踪功能，更能让用户清楚地看到每项测试实验都由哪些人负责。

3.5 各系统关系映射

为更直观地显示产品部件或器件在实验项目探测器上的安装位置，例如光电倍增管在江门中微子

实验中球型探测器上的具体位置，我们可以通过三维立体图像方式给出安装位置的映射。

三维立体图像是基于 three.js 的，three.js 是一个 3D JavaScript 库，支持许多格式的 3D 模型导入，比如我们用到的 *.obj 模型。加载 *.obj 文件需要另外引入 three.js 提供的 OBJLoader.js 和相应材质文件 *.mtl 对应的 MTLLoader.js。然后在函数中设置路径并对三维对象做一些处理：对材质进行调色、设置透明度、设置贴图模式等，从而得到我们需要的三维图像^[13]。

4 结语

本文介绍了基于 Ionic 开发平台创建的混合移动应用程序，可运行于手机客户端(APP)与 Web 端的数据管理系统，对实验数据的操作更加方便快捷，为实验的进行提供较大便利。

参考文献

- 1 姚宁, 张家文, 郑国恒, 等. BESIII MUON 探测器建造和质量控制数据库[J]. 核电子学与核探测技术, 2006, 26(2): 203-207.
YAO Ning, ZHANG Jiawen, ZHENG Guoheng, et al. Construction of BESIII MUON detector and quality control database[J]. Nuclear Electronics & Detection Technology, 2006, 26(2): 203-207.
- 2 罗伟. 大亚湾中微子振荡反符合探测器 RPC 性能测试及其质量控制数据库建立[D]. 衡阳: 南华大学, 2009.
LUO Wei. The performance of testing about DayaBay neutrino oscillation anti-coincidence detector of RPC and the database of quality control[D]. Hengyang: Nanhua University, 2009.
- 3 曹俊. 大亚湾与江门中微子实验[J]. 中国科学: 物理学力学 天文学, 2014, 44(10): 1025-1040.
CAO Jun. Daya Bay and Jiangmen underground neutrino observatory (JUNO) neutrino experiments[J]. Scientia Sinica: Physica, Mechanica & Astronomica, 2014, 44(10): 1025-1040.
- 4 Cao Z. LHAASO: science and status[J]. Frascati Physics Series, 2014, 58: 331-342.
- 5 Qian S. The 20 inch MCP-PMT R&D in China[R]. California: California Institute of Technology, 2016.
- 6 JavaScript[EB/OL]. 2016. <https://zh.wikipedia.org/wiki/JavaScript>.
- 7 Comparing the top frameworks for building hybrid obile-apps[EB/OL]. 2015. <http://tutorialzine.com/2015/10/comparing-the-top-frame>.

- 8 Ionic react-native native 优劣势对比[EB/OL]. 2016. <http://www.ionic.wang/article-index-id-69.html>.
Comparison of the ionic, react-native and native[EB/OL]. 2016. <http://www.ionic.wang/article-index-id-69.html>.
- 9 使用 NodeJS+Express 开发服务端 [EB/OL]. 2016. <http://www.jianshu.com/p/db4df1938eca>.
Use NodeJS+Express to develop the server[EB/OL]. 2016. <http://www.jianshu.com/p/db4df1938eca>.
- 10 HeidiSQL. Basic help on using HeidiSQL[EB/OL]. 2016. <http://www.heidisql.com>.
- 11 Navicat[EB/OL]. 2016. <https://www.navicat.com.cn>.
- 12 html5 移动 app 框架 ionic 热更新插件-不发布应用市场动态更新 APP 源码[EB/OL]. 2016. <http://www.ionic.wang/article-index-id-71.html>.
The hot push plugin of ionic-update app source don't need to publish on the app market dynamics[EB/OL]. 2016. <http://www.ionic.wang/article-index-id-71.html>.
- 13 ThreeJS 学习笔记 (二) -导入外部模型[EB/OL]. 2016. <http://feg.netease.com/archives/301.html>.
ThreeJS study notes(II)-import the external model[EB/OL]. 2016. <http://feg.netease.com/archives/301.html>.