

青海湖生态水文监测数据可视化平台^①

萧晓俊^{1,2}, 罗万明¹, 罗泽¹, 阎保平¹

¹(中国科学院 计算机网络信息中心, 北京 100190)

²(中国科学院大学, 北京 100049)

通讯作者: 罗万明, E-mail: lwm@cinc.cn

摘要: 青海湖是中国最大的内陆湖泊和咸水湖, 是候鸟的栖息地, 具有重要的生态研究价值. 为了有效管理和应用青海湖获取采集的生态监测数据, 本文设计并实现了青海湖生态水文监测数据可视化平台. 平台使用 Spring MVC 作为后台开发框架. 数据库使用开源时序数据库 InfluxDB, 并利用 InfluxDB 的数据保留策略和连续查询对数据进行聚合存储. 可视化功能使用开源的 JavaScript 工具 Echarts 实现, 对青海湖的大气环境、水文、土壤等生态数据进行可视化.

关键词: 青海湖; 生态监测; 可视化

引用格式: 萧晓俊, 罗万明, 罗泽, 阎保平. 青海湖生态水文监测数据可视化平台. 计算机系统应用, 2018, 27(10): 75-79. <http://www.c-s-a.org.cn/1003-3254/6583.html>

Qinghai Lake Ecological Hydrological Monitoring Data Visualization Platform

XIAO Xiao-Jun^{1,2}, LUO Wan-Ming¹, LUO Ze¹, YAN Bao-Ping¹

¹(Computer Network Information Center, Chinese Academy of Sciences, Beijing 100190, China)

²(University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Qinghai Lake is the largest inland lake and saltwater lake in China. It is the habitat for migratory birds and has important ecological research value. In order to effectively manage and apply the ecological monitoring data collected in Qinghai Lake, this study designed and implemented the Qinghai Lake ecological hydrological monitoring data visualization platform. The platform uses Spring MVC as a background development framework. The database uses the open source timing database InfluxDB and uses InfluxDB's data retention strategy and continuous queries to aggregate data. The visualization function is implemented by Echarts, an open-source JavaScript tool. It can visualize the Qinghai Lake's ecological data such as atmospheric environment data, hydrology data, and soil data.

Key words: Qinghai Lake; environmental monitoring; visualization

青海湖位于中国青海省, 在刚察县、共和县及海晏县交界处, 面积 4435.69 平方公里. 青海湖长 105 公里, 宽 63 公里, 湖面海拔 3196 米, 平均深度 21 米, 最大深度 32.8 米. 它是中国最大的内陆湖泊和咸水湖, 湖面东西长, 南北窄. 它四面有环山包围, 北面是大通山, 东面是日月山, 南面是青海南山, 西面是橡皮山. 青海湖水资源丰富, 湖周大小河流有 70 多条, 湖北岸、西

北岸和西南岸河流多, 流域面积大, 支流多, 具有重要的水文研究价值^[1,2].

青海湖生态水文监测数据的采集已经持续了很多年. 但是不同的传感器数据类型不同, 而且数据量庞大, 有属于大气环境的风速风向、空气温度、空气湿度、太阳总辐射、地表红外温度、饱和水汽压、二氧化碳等的的数据, 也有属于土壤的含水量、电导率、介电系

① 收稿时间: 2018-02-28; 修改时间: 2018-03-19; 采用时间: 2018-04-08; csa 在线出版时间: 2018-09-28

数、温度的数据,还有降水量、蒸发量等水文数据.丰富的数据类型和庞大的数据量迫使我们的使用可视化平台对青海湖生态监测数据进行管理和应用.

为此,本文设计实现了青海湖生态水文监测数据可视化平台.该平台使用 Echarts 作为可视化工具^[3-5], InfluxDB 时序数据库^[6-8]作为存储手段,可以对青海湖的大气环境、水文、土壤等生态数据以柱状图、折线图、散点图等多种图形进行展示,便于管理和应用.

1 青海湖生态水文监测数据可视化平台设计

1.1 系统整体架构

图1是青海湖生态水文监测数据可视化平台的系统整体功能设计结构图.整个可视化平台主要分成三个部分,一个用户管理模块负责管理用户的登录和注销,一个数据处理模块负责对原始数据进行处理导入到数据库,一个数据可视化模块负责拉取数据并进行可视化.

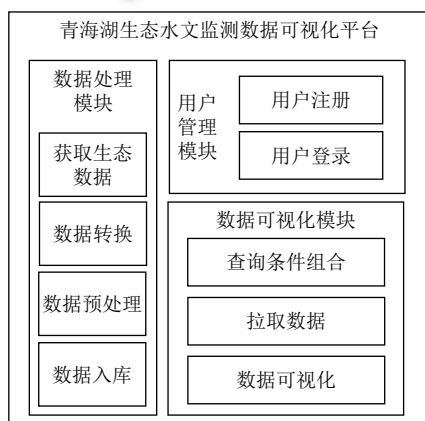


图1 系统整体架构

首先,用户管理模块是必不可少的.然后,由于原来的青海湖收集数据的系统把收集的传感器数据都放到 dat 文件中,所以我们需要对原始数据进行处理并导入到数据库中.数据处理模块用于完成该工作.数据处理模块会定时地读取文件,获取新采集的数据,对数据进行格式的转换,再对转换的数据进行过滤处理,去掉不符合规则的脏数据,最后再把数据存到数据库中.

系统采用 B/S 结构,由浏览器端和服务端组成.浏览器端是用户对可视化系统进行操作互动的平台,通过一个用户登录验证的功能然后进入到青海湖生态水文监测数据可视化的主平台.具体的响应逻辑则由服务器端完成,两者通过 HTTP 报文进行交互.整个平台

采用现在主流的 MVC 结构^[9,10],分别是模型 (Model)、视图 (View) 和控制器 (Controller) 三个部分.模型封装后台生态数据和相关实体类的操作.控制器响应浏览端请求,完成具体的业务逻辑.视图完成图形的渲染,生成与用户进行交互的各个界面.

1.2 工作流程

图2是青海湖生态水文监测数据可视化平台的整体工作流程.首先用户通过登录验证后进入到可视化平台的 Web 端.在可视化平台的 Web 端,用户选择观测站地点,观测数据类型,开始时间和结束时间等筛选要素.之后通过 HTTP 请求到可视化平台后台.后台进行主要的逻辑处理,把用户的选择组装成数据库查询,并把结果按照前端的要求进行返回.最后,前端通过可视化工具对数据进行相应的可视化.

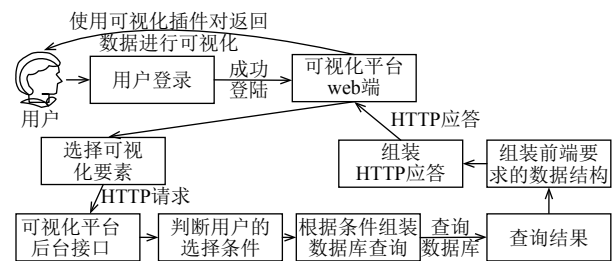


图2 工作流程

2 关键技术

2.1 Spring MVC

Spring MVC 使用简单,性能高效,可扩展性和可读性良好,是当前主流的后台开发框架.图3是 SpringMVC 的工作流程.首先,前端控制器会拦截 HTTP 请求.然后查询处理器映射器,根据 HTTP 请求的 URL 匹配对应的处理器.成功匹配处理器后,前端控制器会转发请求,处理器负责处理业务请求并返回 ModelAndView 对象.前端控制器再请求视图解析器对 ModelAndView 对象进行解析.视图解析器会根据视图名称形成真正的视图,一般是 JSP 文件,作为视图对象进行返回.获取渲染视图,其实就是将 ModelAndView 对象中的数据放到 HTTP 应用中的一个域中.最后前端控制器进行 HTTP 响应,返回的 HTTP 中包含 JSP 文件的地址和渲染需要的数据,前端就可以渲染视图并展现给用户.

2.2 InfluxDB 数据库

InfluxDB 数据库的存储引擎使用 TSM Tree (Time Structured Merge Tree).在 InfluxDB 数据库中,增删改

操作都是直接追加到对应的文件的末尾,并不会马上就对数据进行改变,而是通过一个定时任务把数据处理真正完成.而且内存则是直接插入到 TSM Tree 中,进行查询可以查询到最新的数据的数据状态,不会发生查询到已删除数据的情况.这样的设计可以很好地支持重写轻读的场景.

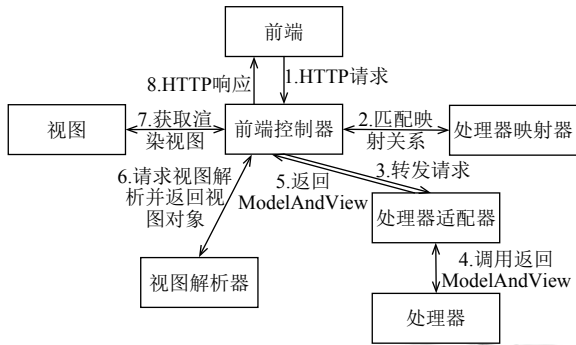


图3 SpringMVC 工作流程

InfluxDB 数据库有数据保留策略和连续查询的功能. InfluxDB 的数据保留策略,就是可以设置数据在数据库的生存时间,如果超过生存时间,数据会被删除.而连续查询可以定期地对数据库表进行查询,把查询的结果导入到另一张表中.用于对数据进行压缩聚合.两者可以配合使用,可以使得新的数据有较高的数据精度而旧数据有较低的数据精度.

2.3 Echarts 可视化工具

Echarts 是开源的 JavaScript 可视化工具.和大多数的开源可视化工具相比,Echarts 使用简单而且性能良好. Echarts 只需要向 echarts 实例设置统一的 Option 对象就能完成可视化.具体流程:先定义一个 div 区域用于可视化(如图4所示).然后使用如图5所示的 JavaScript 语句先为定义的区域初始化一个 echarts 实例,再把 HTTP 响应的 json 格式的字符串数据反序列化成 Echarts 可以识别的 Option 对象,最后把 Option 对象设置到 echarts 实例中,即可完成可视化.

```
<div align="center" style="clear: both">
  <div id="main" style="width: 900px;height:400px;"></div>
</div>
```

图4 div 区域

```
// 基于准备好的dom,初始化echarts实例
var myChart = echarts.init(document.getElementById('main'));
// 转换json成JavaScript对象
var myOption = JSON.parse(getData);
// 使用刚指定的配置项和数据显示图表。
myChart.setOption(myOption, true);
```

图5 可视化 JavaScript 语句

3 系统实现

3.1 数据可视化模块

数据可视化模块的主要功能是完成数据的可视化.对数据进行可视化也是青海湖生态水文监测数据可视化平台的主要任务.

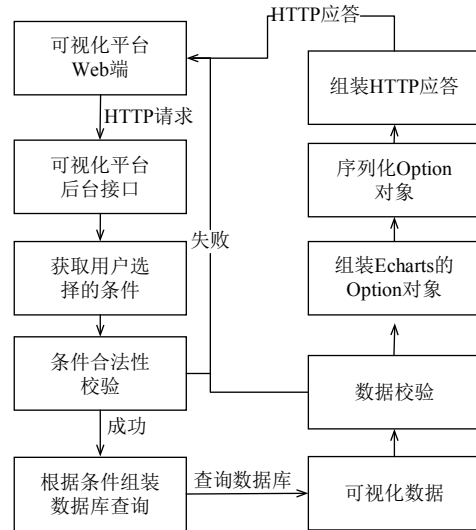


图6 可视化处理接口的具体流程

数据可视化模块对外主要提供一个接口.该接口接收用户的可视化请求.该接口的具体流程如图6所示.首先,接口接收 HTTP 请求,然后获取 HTTP 请求中用户的可视化筛选条件值.接着要对条件值进行校验,验证非空、边界等等.成功校验后根据条件组装数据库查询,根据观测站地点选择查询的数据库表,根据观测数据类型选择查询的字段,根据查询的时间区间组装 Where 语句,根据统计维度(按小时、天、月、年)组装 Group By 语句.数据库查询成功后,对得到的可视化数据进行非空校验,如果返回为空,直接返回失败.否则,组装 Echarts 的 Option 对象.然后把 Option 对象序列化成字符串.最后返回 HTTP 应答.

以下是一个完整的 Option 对象结构示例(其中数据都被隐藏):

```
{
  "yAxis": {},
  "xAxis": {
    "data": Array[97]
  },
  "legend": {
    "data": [
      "PTemp"
    ]
  }
}
```

```

    ]
  },
  "series":[
    {
      "data":Array[97],
      "name":"PTemp",
      "type":"bar"
    }
  ],
  "tooltip":{},
  "toolbox":{
    "feature":{
      "saveAsImage":{
        "show":"true"
      },
      "restore":{
        "show":"true"
      },
      "magicType":{
        "show":"true",
        "type":[
          "line",
          "bar"
        ]
      }
    }
  },
  "dataView":{
    "show":"true",
    "readOnly":"false"
  },
  "mark":{
    "show":"true"
  },
  "show":"true"
},
"title":{
  "text":"graph"
}
}

```

浏览器前端则主要使用 bootstrap 框架, 优化界面的感官效果, 为用户提供更良好的用户体验. 同时使用 Ajax 技术完成条件选择和可视化展示的动态交互.

3.2 数据处理模块

因为收集传感器数据的程序读写频繁, 而读写文

件与读写数据库相比, 前者是一个轻量级的操作而后者是一个重量级的操作. 所以, 青海湖的观测数据是以文本文件形式存储的. 但是以文件保存的生态数据不便于管理和应用, 所以系统使用一个数据处理模块定时地把传感器收集的原始数据导入到数据库中. 青海湖的数据收集系统会按传感器地点和类型用一个文件保存一天的数据, 所以我们使用定时器每天处理一次数据. 定时器设置在每天的早上 4 点触发, 处理前一天收集的数据.

具体流程如图 7 所示. 定时器触发后, 检查收集数据的目录下的文件, 匹配文件名, 查看是否有前一天的数据文件. 如果有, 则开始遍历读取数据文件. 对于每个数据文件, 逐条数据读取并判断看数据是否合法, 如果数据符合要求则缓存到一个缓存区中. 每当缓存区的数据行数达到 5000 条, 则批量写入到数据库中 (InfluxDB 官网文档表示批量写入数据每次 5000 条性能最佳). 如果该数据文件的数据已经读取完成, 也将缓存区的数据批量写入到数据库中. 该数据文件处理完成后, 将该文件移动到已处理数据的目录下, 避免收集数据目录下文件过多, 也省去对重复文件的判断. 之后继续遍历文件, 直到完成对前一天新收集的数据文件的处理.

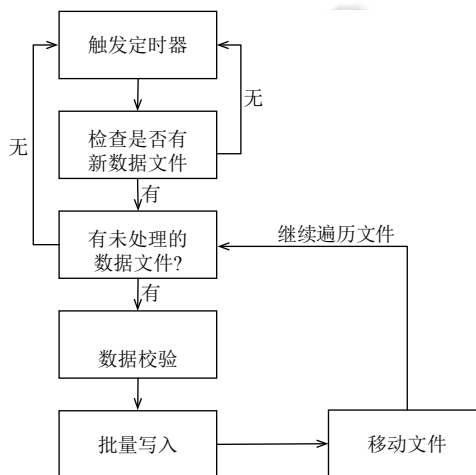


图 7 数据处理的具体流程

3.3 数据聚合

现在青海湖的数据收集系统运行已经超过 10 年, 每年数据量大概是 300 万条. 我们使用 InfluxDB 的数据保留策略和连续查询对数据进行压缩聚合. 我们设置数据的有效时间是一年 (图 8), 并把一年后的数据聚合成每天的均值 (图 9).


```
CREATE RETENTION POLICY "1_year" ON "sensor_data" DURATION 365d REPLICATION 1 DEFAULT
```

图8 数据保留策略

```
SELECT avg(*) INTO "cr3000_3_lcd_day_1" FROM "cr3000_3_lcd_min_1" GROUP BY time(1d)
```

图9 连续查询

3.4 系统展示

用户完成登录认证进入到可视化平台后,通过在

左边选择观测的数据类型,在右边选择观测站地点(蛋岛、鸬鹚岛)、图表类型(柱状图、折线图、散点图、涟漪散点图、面积图等),选择开始时间和结束时间,选择按小时、日、月或者年进行统计,然后点击确定按钮就可以把相应的生态数据进行聚合展示,如图10所示。生成的可视化图还可以以图片的形式下载到本地,也可以查看视图的原始数据。还支持在图上通过点击转换的按钮,快速地完成向折线图和柱状图的转换。



图10 平台可视化示例

4 结论与展望

青海湖生态水文监测数据可视化平台用于对青海湖生态数据进行可视化,在百万级数据量的情况下依然能够快速响应。整个平台各个模块解耦,互相之间通过接口进行调用,便于以后扩展新的功能。下一步将添加更多的功能,给各个站点添加实时的数据监控,支持更多的图表类型,增加数据分析模块,实现更细粒度的可视化展示,并在系统的性能方面展开进一步的研究工作,提升系统的性能水平。

参考文献

- 1 朱延龙,韩昆,王芳. 青海湖流域气候变化特点及水文生态响应. 中国水利水电科学研究院学报, 2012, 10(4): 260-266. [doi: 10.3969/j.issn.1672-3031.2012.04.005]
- 2 刘小园. 青海湖流域水文特征. 水文, 2004, 24(2): 60-61. [doi: 10.3969/j.issn.1000-0852.2004.02.016]
- 3 王子毅,张春海. 基于 ECharts 的数据可视化分析组件设计实现. 微型机与应用, 2016, 35(14): 46-48, 51.
- 4 冀潇,李杨. 采用 ECharts 可视化技术实现的数据体系监控系统. 计算机系统应用, 2017, 26(6): 72-76.
- 5 段晓东. 基于 JSP 的水资源监测系统的设计与实现[硕士学位论文]. 呼和浩特: 内蒙古农业大学, 2017.
- 6 朱亚楠. 基于 OpenStack 云资源监控系统设计与实现[硕士学位论文]. 成都: 电子科技大学, 2016.
- 7 Balis B, Bubak M, Harezlak D, et al. Towards an operational database for real-time environmental monitoring and early warning systems. Procedia Computer Science, 2017, 108: 2250-2259. [doi: 10.1016/j.procs.2017.05.193]
- 8 黄红远,李高明,欧阳为年,等. 一种新型时序数据库在大型配电网自动化系统数据中心中的应用. 中国电业·技术, 2014, (7): 35-37. [doi: 10.3969/j.issn.1002-1140.2014.07.013]
- 9 唐永瑞,张达敏. 基于 Ajax 与 MVC 模式的信息系统的设计与研究. 电子技术应用, 2014, 40(2): 128-131. [doi: 10.3969/j.issn.0258-7998.2014.02.040]
- 10 薛峰,梁锋,徐书勋,等. 基于 Spring MVC 框架的 Web 研究与应用. 合肥工业大学学报(自然科学版), 2012, 35(3): 337-340.