**2020年全国大学生“互联网+”创新大赛**

**暨 第八届“发现杯”全国大学生互联网软件设计大奖赛**

‘基于大数据的基金组合智能推荐系统’项目

创意设计报告

（2020年3月8日）

目录

[一、 创意介绍 3](#_Toc66214268)

[1.1 项目介绍 3](#_Toc66214269)

[1.2 项目可行性分析与应用前景 3](#_Toc66214270)

[1.3 与同类产品比较 4](#_Toc66214271)·

[二、功能介绍 5](#_Toc66214272)

[2.1 总体功能结构图 5](#_Toc66214273)

[2.2 模块功能介绍 5](#_Toc66214274)

[三、总体设计 8](#_Toc66214275)

[3.1 数据设计 8](#_Toc66214276)

[3.2 界面设计 10](#_Toc66214277)

[3.3 架构设计 17](#_Toc66214278)

[四、技术难点 18](#_Toc66214279)

[4.1 数据爬取技术难点 18](#_Toc66214280)

[4.2说明某一个框架的应用场合 19](#_Toc66214281)

# 创意介绍

## 项目介绍

最近两年投资理财的热度越来越高，特别是刚过去的2020年，基金投资成为了人们投资理财的热点，一大批90后和00后的投资小白都涌入的基金投资的市场，基金投资其实就是将钱聚集到基金公司，有专门的职业基金经理进行资金的管理，本质上也是投资股票，债券和货币的产品，所以也存在较大的投资风险，所以市面上又涌现出了一大批基金组合，包括一些大V和基金经理参与其中。基金组合是选择多个基金进行搭配，已达到进一步较低风险的目的，涌现出的基金组合投资平台包括天天的组合宝、蛋卷平台，且慢平台等等。诚然基金组合的想法非常好，但是里面也拆咋各种良莠不济的组合管理人，牛市中各种吹嘘，实际上他们管理基金组合并不适合新入场的投资小白。

市面上有5000多种发行的基金，每天会产生大量的数据，符合大数据的特点。所以本项目就利用大数据的相关技术，分析各个平台的基金组合的外在和内在表现，实现一个基金组合智能推荐系统。本系统的数据来源主要是通过爬取天天组合宝和蛋卷基金平台等组合信息。通过深度的处理后，形成用户可见的数据。

通过该系统，用户可以查询天天组合宝和蛋卷基金平台所有的组合情况，包括组合中相关的数据和图表，并通过人性化的操作界面，按照相关顺序查询所有的基金组合情况，如果用户登录注册后，填写意向投资表，系统会实时分析平台中所有的数据，并通过一定的友好算法来推荐最适合的此用户的基金。

未来还会此系统还用通过用户点击、查询和打分等功能，实时的分析用户个各种爱好需求，并分析此用的投资倾向以做到更优的推荐。

## 项目可行性分析与应用前景

项目分为4个部分：

1、数据爬取系统

爬取系统的所需数据，并进行简单处理。采用Java语言进行开发，使用Retorfit+Okhttp+Gson框架进行网络请求，通过正则表达式来分析网页数据，同时使用Jpath对JSON字符串进行处理，把处理好的数据解析为JSON的标准数据格式。一每行一行字符串的存储在指定文件夹中。

爬虫系统通过Java多线程技术来启动5-10个爬虫服务。并使用定时器每天凌晨来更新最新数据，同时把过期的冗余数据进行删除。

3、数据收集系统：

主要使用的框架为flume和kafka，Flume来监听数据的文件夹，文件里面存储的爬虫和业务端产生的日志，爬虫产生数据通过flume传输到HDFS文件仓库上，同时业务数据通过flume，传输到kafka消息队列上，Spark-Steaming读取kafka里面的数据并进行处理。

4、数据仓库：

数据仓库包括两个部分，一个用存储爬虫产生的数据的HDFS仓库，另一个主要是业务数据库MongoDB，它主要为用户业务系统服务。在爬虫把文件源源不断的爬取下来后，存储在指定的文件夹中，

5、数据分析系统：

主要使用Scala语言。使用Spark进行数据分析，Spark从HDFS来加载系统需要的数据，并通过Spark-sql，sqark-ml等子框架来进行数据的清洗聚合和数据的分析。最后通过MongoDB的驱动来把处理好的数据存储再MongoDB中。

6、业务系统：

主要使用Java语言。采用SpringBoot来搭建Web服务器。SpringBoot使用JPA和mongodbTemplate来进行数据的查询服务。通过标准的JSON格式传输到前端。前端使用Vue和echarts等技术形成可视化的网页。

## 与同类产品比较

本项目的同类产品有：蛋卷平台，天天基金组合宝，且慢App。这三个平台或功能模块都包含自家平台的基金组合信息，蛋卷平台，天天基金组合宝这两个可以查看本组合的所有与基金相关的信息，信息确实是全而杂，但是他没有相应的推荐系统，只是纯粹的根据收益率的高低进行排序推荐，缺乏了相应的灵活性和智能性，所有说他说只是纯粹的投资平台，而不是真正基金组合推荐平台，且慢App更加极端，信息完全不透明，只能跟随一些大V。

而本系统吸取三个平台的缺点，做到不仅可以通过本产品查看所以基金组合的关键信息，并进行灵活排序，提供给用户更加的灵活性和可选择行。并且通过用户填写用户投资计划表，生成用户画像，通过一定算法推荐符合用户自身投资倾向的基金组合。

# 二、功能介绍

## 2.1 总体功能结构图

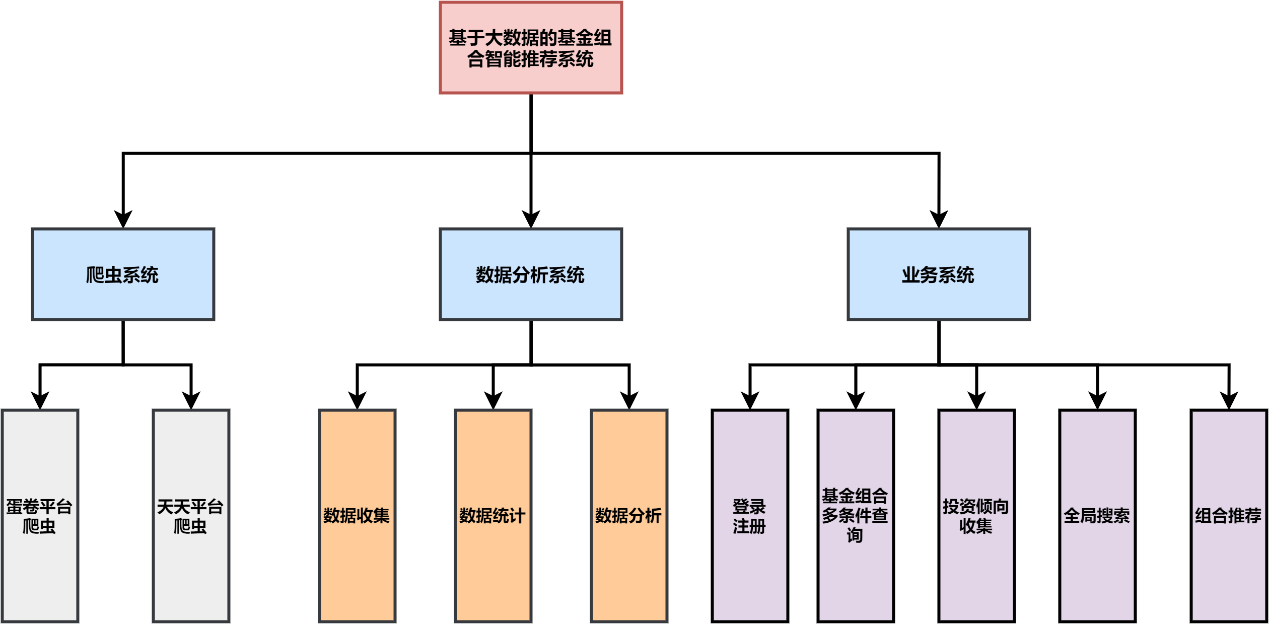


图2.1 系统功能图

## 2.2 模块功能介绍

#### 2.2.1 爬虫系统功能模块

爬虫分为另个部分，蛋卷平台爬虫和天天平台爬虫。蛋卷平台爬取了有7个URL链接，包括组合list、组合的Homepage、plan、indicator、summary、detail和growth。这些链接基本包括所有的数据，把这些数据以类型+平台+时间的形式存储为文件，

天天平台包括三个URL连接，包括基金组合排行，详情页和基金增长情况，因为数据量比较乱，进行了简单的处理，方便spark读取。同样以类型+平台+时间的形式存储为文件，放在指定的文件夹中，等待flume的读取。

爬虫系统模块还使用了Java定时器功能，每天凌晨进行数据的爬取，以便获取最新的数据。定时器开始时，会先获取需要爬取的基金列表，然后更加列表分为5个部分，每个部分对应一个线程，等最后一个线程终止了，把数据组合起来并写入文件。

#### 2.2.2 数据分析系统功能模块

1、数据收集

爬虫数据部分的收集使用了flume和HDFS，flume作为一个常用的日志收集工具，经常用在分布式系统系统中，本系统使用flume，并配置它的spooling源，指定spooling需要监听的文件夹，然后配置flume的hdfs\_sinks，指定hdfs一些配置信息，包括type、path和fileSuffix等信息。

另外一个功能是收集用户的行为信息包括用户浏览内容、时间和打分等信息，数据的来源主要是通过业务系统的业务层的日志埋点输出到指定文件夹，然后通过flume监听输出给kafka消息缓存队列中。

2、数据统计

数据统计较为简单，首先需要通过配置Spark一些基本信息如需要开启的内核数，读取文件是以上面的规则类型+平台+时间的形式读取hdfs的文件链接，数据读取完成后会形成spark的dataframe，然后就能通过Spark-SQL的一些操作进行数据的统计，比如使用select和orderby等函数来计算跟投人数的顺序，使用agg和groupby等函数计算基金组合类型的分类等等。

#### 2.2.3 业务系统功能模块

1、登录注册功能

登录注册功能较为常规，本系统主要需要有用户的功能，才能做推荐，如果需要精准推荐，必须要保留用户信息。用户信息主要有用户的账号和密码，账号用于登录系统，并且作为此用户的唯一ID来做相似性基金组合推荐功能。这业务系统中，登录注册分别对应/user/login和/user/register请求地址，请求方法都是POST。请求时SprintBoot框架会通过validate模块来做参数校验，比如校验用户账号和密码不能为空、密码不能小于6位等。

2、用户投资倾向填写功能

用户投资倾向填写主要通过填写一些用户基本信息来更精准的给用户推荐适合的基金组合信息，它主要包括用户名、投资金额、投资时长、投资类型、期望的利润率、能接受的最大回撤等信息。把这个信息收集存储在user\_tendency表中，并提供用户的修改和查询功能。

3、基金组合多条件浏览功能

基金组合多条件浏览主要是通过本系统，用户可以直观的查看基金组合的一些基本信息，方便用户作决断。支持的查询条件包括综合排序、年化从大到小，总收益从大到小、回撤从小到大，夏普比率从大到小，跟随人数和持仓金额从大到小等排序规则。同时也提供投资策略的选择，包括激进型、灵活性和稳健性策略选择。管理者类型中可以选择基金达人和普通基民的选项。并且可以选择这条基金组合的来源平台，包括天天基金平台和蛋卷基金平台。系统默认以综合排序进行显示，其他选择条件默认为空，每页显示10条，并且分页显示。

4、基金组合全局模糊搜索功能

此功能主要是提供用户搜索基金组合功能，搜索的条件有基金组合代码、基金组合名、基金组合管理人。三个搜索合为一个链接，搜索时实时提示，搜索内容中还可以显示必要的基金组合的基本信息，方便用户可以快速浏览想要搜索的内容，并且快速判断搜索的内容的正确性。业务后台主要是通过多条件过滤实现，每个查询条件都进行一次查询，查询后组合在HashMap中，同时通过HashMap的可以key来去重复。

5、基金组合详情页查看功能

基金组合的详情页是显示基金组合的详细信息的页面，通过点击首页的列表或者搜索结果进入，它可以显示有关基金组合的所有信息，包括成立以来收益，成立天数，日涨跌，成立以来年化，最新净值等关键统计信息，还能绘制基金成立之后的收益曲线，还可以查看基金组合的持仓分布和持仓的类型等信息。

6、基金组合推荐列表功能

获取本系统的通过大数据计算后，最符合用户倾向的组合10条，以相似度进行排序，越往上，相似度越高。点击后，可以跳转到基金组合的详细页。

# 三、总体设计

## 3.1 数据设计

#### 3.1.1 数据流程设计

主要有两条主线，一条是爬虫系统到前端页面，一条是从业务系统到前端页面。爬虫系统到前端页面数据流通线，以文本文件的形式经过了文件夹，Flume和HDFS，然后经过处理以带格式的数据流的形式被Spark读取，Spark处理数据后，将DataFrame的格式存储在MongoDB中信息JSON，然后业务系统读取形成Java对象，经过进一步处理后，以标准JSON的形式响应给前端页面，呈现给用户。

另个一条业务系统到前端页面的数据流通线，通过Flume和kafak实时的被spark读取处理，处理后的结果存入业务数据库MongoDB，最后流入前端页面。

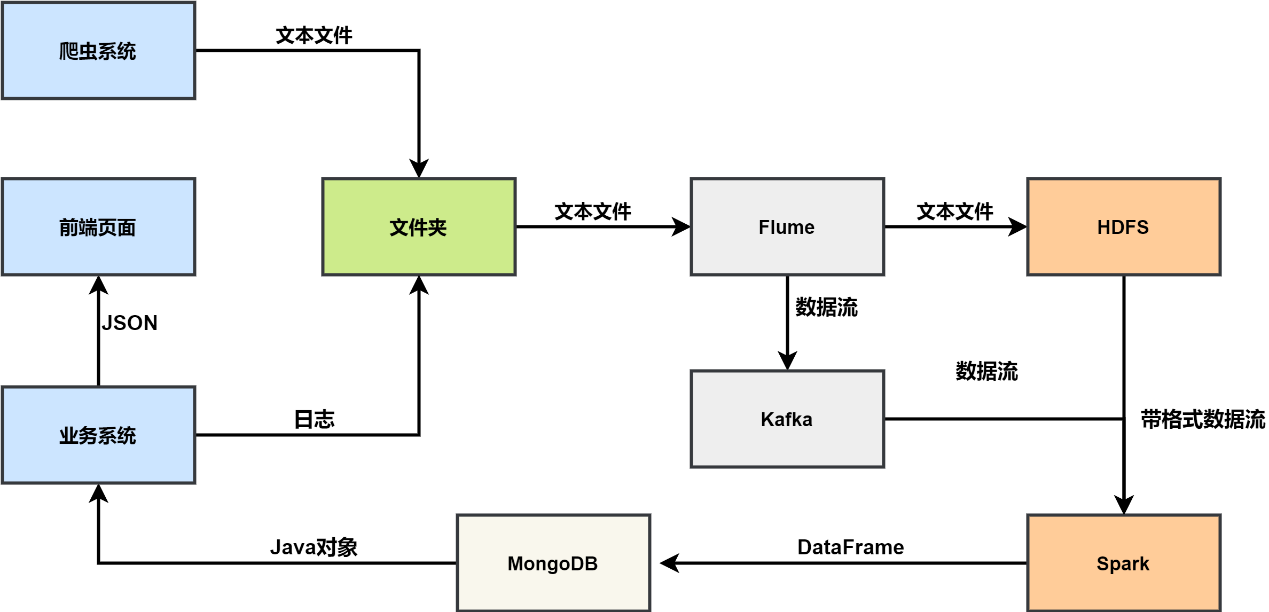


图3.1 数据流图

#### 3.1.2 数据仓库设计

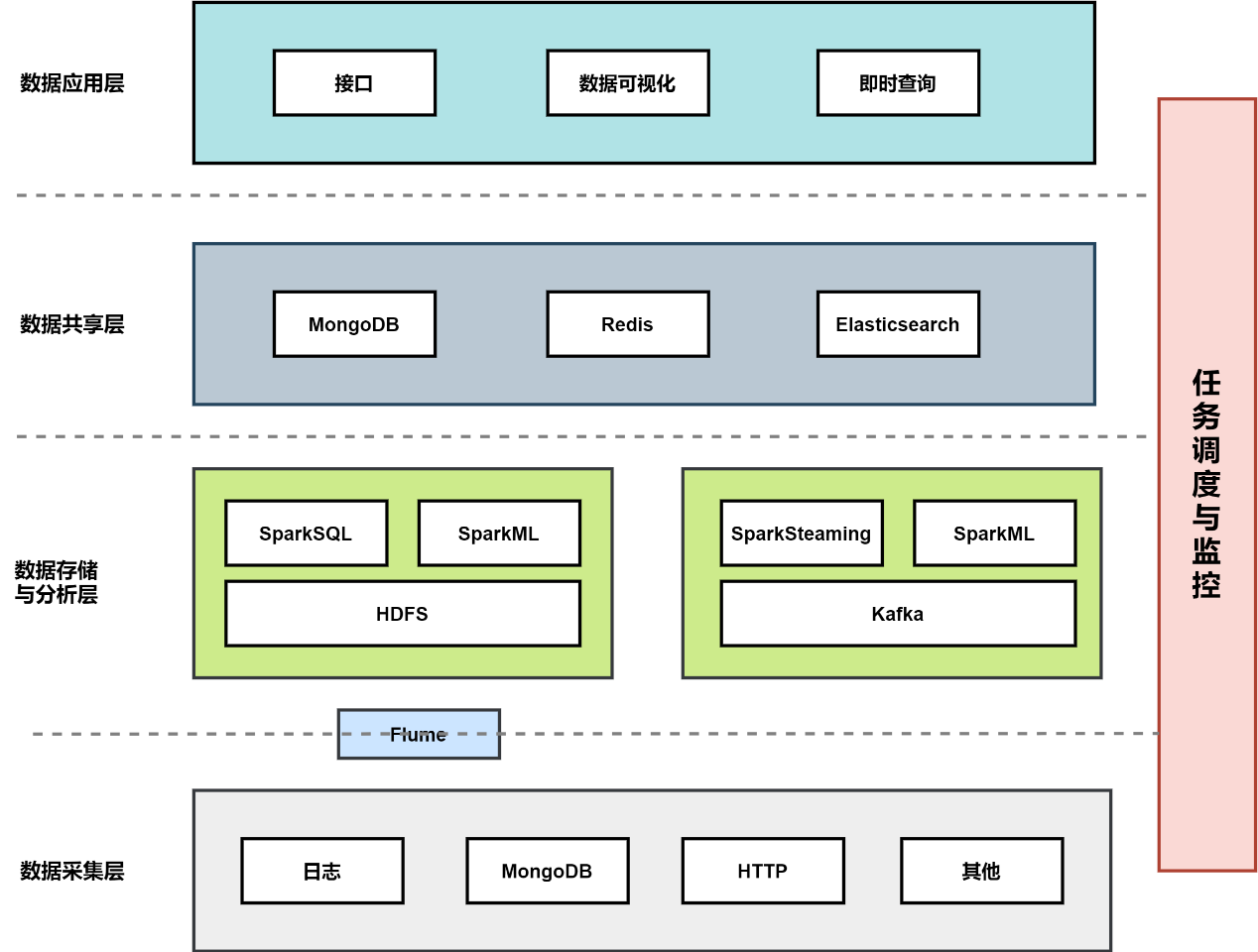


图3.2 数据仓库架构图

数据仓库分为四层架构，从下至上依次为：

1. 数据采集层

日志采集层主要采集信息来自业务系统的日志， 业务数据库，HTTP协议的请求数据和其他各种数据。

1. 数据存储与分析层

非结构化数据主要存储在HDFS中，它接收来自Kafka和Flume里面的数据或者文件。SparkSQL或SparkML读取其中的数据进行分析。同时使用SparkStreaming读取kafka里面的数据结合SparkML进行实时分析。

1. 数据共享层

数据共享层分为主要给业务系统所使用，采用文档数据库MongoDB作为主要的业务数据库，Redis保留缓存数据，ES来做快速的分布式查询功能。

1. 数据应用层

数据应用层主要包括业务接口，数据可视化和即时查询功能。

## 3.2 界面设计

#### 3.2.1 登录注册界面



图3-1 登录界面



图3-2 注册界面

#### 3.2.2 投资倾向填写界面

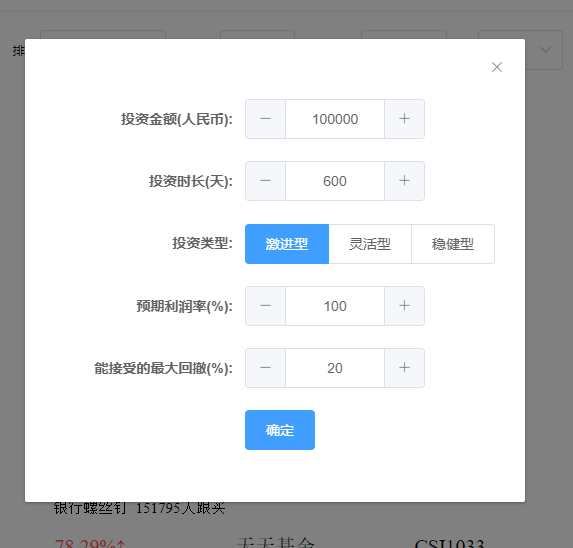


图3-3 投资倾向填写界面

#### 3.2.3基金组合主页



图3-4 系统主页



图3-5 排序规则1



图3-6 排序规则2

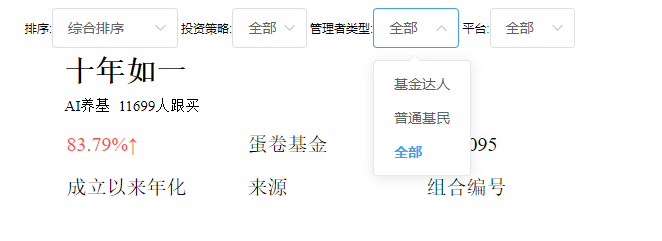


图3-7 排序规则3



图3-8 排序规则4

#### 3.2.4 基金组合详细页



图3-9 基金组合详情1

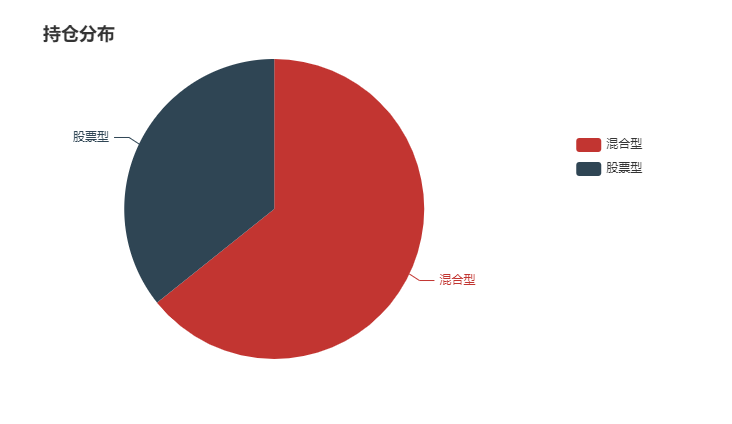


图3-10 基金组合详情2



图3-11 基金组合详情3

#### 3.2.5 基金推荐界面



图3-12 推荐

## 3.3 架构设计

#### 3.3.1 架构设计

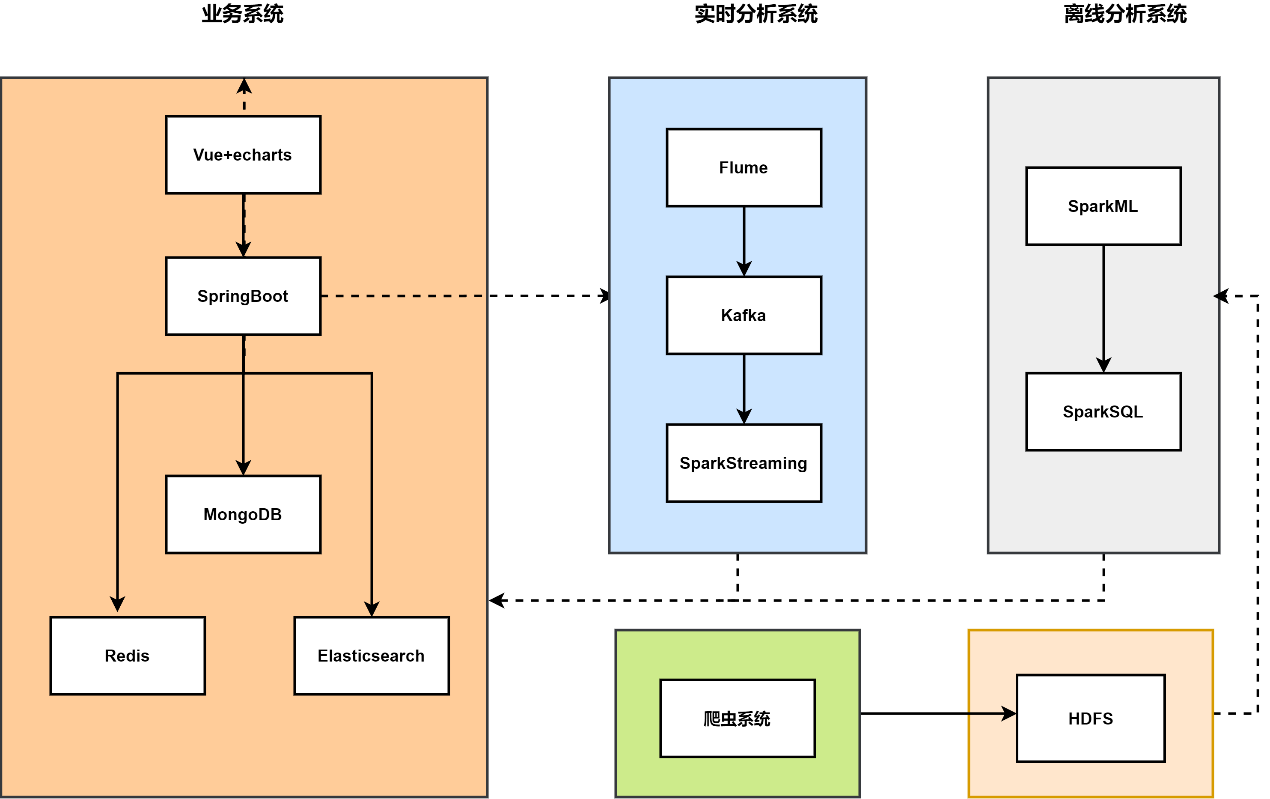


图3-13 系统架构

本项目采用多层架构，业务系统的前端使用Vue+echarts框架来实现可视化界面，业务后端私用SpringBoot做作为web服务器，SpringBoot调用MongoDB、Redis和Elasticsearch来实现业务功能。这样能做到请求响应快，服务器负载低。

实时分析系统采用SparkSteaming读取kafka中的数据来做到数据的实时分析。

离线分析系统，使用SparkSQL功能来查询、聚合和分解相关的数据，然后用现成的SparkML推荐分析库，能快速的搭建一个高可用的大数据分析系统。

爬虫系统独立于主服务之外，它是一个定时服务，每一天执行一次，定时的更新数据到HDFS中供Spark读取。

#### 3.3.2 架构的扩展情况

业务系统的MongoDB+Redis+Elasticsearch组合，他们都可以搭建分布式的高可用数据服务。Elasticsearch提供了一个分布式多用户能力的全文搜索引擎，基于RESTful web接口，与具体的业务系统可以做到高度奋力，很容易拓展。Redis内存数据库也可以独立开来，充当用户请求缓存的数据库，能极大的应对负载高的情况。MongoDB独特的数据处理方式，可以将热点数据加载到内存，故而对查询来讲，会非常快（当然也会非常消耗内存）；同时由于采用了BSON的方式存储数据，故而对JSON格式数据具有非常好的支持性以及友好的表结构修改性，文档式的存储方式，数据友好可见，数据库的分片集群负载具有非常好的扩展性以及非常不错的自动故障转移。

HDFS具有高容错性，数据自动保存多个副本。它通过增加副本的形式，提高容错性，某一个副本丢失以后，它可以自动恢复，这是由 HDFS 内部机制实现的。能方便的处理数据达到 GB、TB、甚至PB级别的数据。并且可构建在廉价机器它通过多副本机制，提高可靠性。它提供了容错和恢复机制。

系统全部采用通用的技术，很好的延展性和可维护性，可以针对某一处的性能瓶颈而单独的拓展服务器资源，HDFS独有的机制也以充分利用廉价闲置的计算资源，减少运营的成本。

# 四、技术难点

本项目开发过程中，主要遇到1个技术难点，具体技术难点及解决方法如下：

## 4.1 数据爬取技术难点

由于基金组合过于复杂，也就是基金组合的字段太别多，需要访问的请求参数研究很庞杂，并且天天基金客户端平台有防爬机制，更加大的爬取难度。天天基金的防爬直接是通过获取用户手机mac地址来辨别用户请求的真伪，也就是一个mac地址只能在一段时间内访问有限的几十次。所有需要动态虚拟出mac地址，每次隔一段时间进行更换。

数据爬取完后，需要进行处理，由于过多的JSON字符串需要组合，拼接。不可能把JSON映射成对象来做各种处理。所有就需要使用JSON字符串组合拼装和查询技术。这时无意间发现的Jpath工具解决提取的这个难题，组合拼装技术使用了org.json，这个较为常规。

## 4.2说明某一个框架的应用场合

Spark的应用场景

1、大数据处理场景分析

当前大数据处理场景主要分为以下三种类型：

（1）复杂的批量处理（Batch Data Processing），该场景重点在于处理海量数据的能力，而对于处理速度没有很高的要求，通常处理周期在数十分钟到数小时之间；

（2）基于历史数据的交互式查询（Interactive Query），通常处理周期在数十秒到数十分钟之间；

（3）基于实时数据流的数据处理（Streaming Data Processing），通常处理周期在数百毫秒到数秒之间。

对于以上三种大数据处理场景，Spark框架能够以一站式平台的形式分别满足三种需求。

2、Spark适用场景分析

（1）迭代计算：Spark是基于内存的迭代计算框架，适用于需要多次操作特定数据集的应用场合。需要反复操作的次数越多，所需读取的数据量越大，受益越大，数据量小但是计算密集度较大的场合，受益就相对较小；

（2）实时查询：数据体量中等，但是要求严格实时统计分析需求。

3、Spark不适用场景分析

（1）内存容量有限的场景：在内存不足的情况下，Spark任务下放到磁盘，会降低应有的性能；

（2）有极高实时性要求的流式计算业务，例如实时性要求毫秒级的业务；

（3）由于RDD的只读特点，Spark不适用于待分析数据频繁变动的情景，例如数据集频繁变化，而又需要结果具有很强一致性的场景；

（4）流线长或文件流量非常大的数据集：集群压力大会导致内存负载过重，一个task的失败会导致流线所有的前置任务全部重跑，从而造成线性跑包失败的情况，降低整体运行效率。这种情况下MapReduce是更好的替代方案。

4、Spark业务应用分析

（1）推荐系统：主要适用于优化相关排名、个性化推荐以及热点点击分析等；

（2）广告业务：主要适用于应用分析、效果分析、定向优化等。