

“大数据+云计算+微服务”在福建省精准扶贫中的应用^①



胡 波

(福建省财政信息中心, 福州 350003)

通讯作者: 胡 波, E-mail: 358352752@qq.com

摘 要: 如何解决全国精准扶贫和精准脱贫问题, 确保扶贫对象精准、项目安排精准、资金使用精准, 成为中央亟需解决的重要问题. 福建省财政厅利用大数据、云计算、微服务等前沿先进技术在全国率先研究开发福建省扶贫(惠民)资金在线监管系统, 实现对扶贫(惠民)资金的全流程精准监控, 确保资金精准发放. 本文对该系统的体系架构、关键技术应用、实现效果及建设经验等进行了详述. 该系统建设经验可为国家建设其他领域运用大数据、云计算、微服务等技术提供参考和借鉴.

关键词: 大数据; 云计算; 微服务; 精准扶贫

引用格式: 胡波.“大数据+云计算+微服务”在福建省精准扶贫中的应用. 计算机系统应用, 2020, 29(5): 94-102. <http://www.c-s-a.org.cn/1003-3254/7378.html>

Application of Big Data, Cloud Computing and Microservices in Precision Poverty Alleviation in Fujian Province

HU Bo

(Fujian Province Finance Information Center, Fuzhou 350003, China)

Abstract: How to solve the problem of precise poverty alleviation and shaking off poverty in the whole country, ensure the precise target of poverty alleviation, the precise arrangement of projects and the precise use of funds, has become an important problem that the central government urgently needs to solve. Fujian Provincial Department of Finance takes the lead in researching and developing the Fujian Province online supervision system of poverty alleviation (benefiting people) funds by using advanced technologies such as big data, cloud computing, and microservice, so as to realize the accurate monitoring of the whole process of poverty alleviation (benefiting people) funds and ensure the accurate distribution of funds. In this study, the system architecture, key technology application, implementation effect and construction experience of the system are described in detail. The construction experience of this system can provide reference for other fields of national construction to use technologies such as big data, cloud computing, and microservices.

Key words: big data; cloud computing; microservices; precision poverty alleviation

大数据、云计算是目前较为前沿的信息技术, 被广泛用于智慧城市、物联网、金融分析、军事、公检法等各个领域; 微服务技术是近年兴起的先进信息技术, 我国目前虽然有在一些小领域、小规模应用该技

术, 但仍处于试验探索阶段, 特别是如何利用微服务技术优势, 解决传统 SOA 架构普遍存在的并发量瓶颈和无法充分利用云计算资源动态分配功能等问题, 还在进一步探索中. 如何将大数据、云计算、微服务等信

^① 收稿时间: 2019-09-13; 修改时间: 2019-10-10, 2019-10-31; 采用时间: 2019-11-05; csa 在线出版时间: 2020-05-07

息技术同国家建设过程有机结合,解决国家政策执行过程无法及时监督、执行效果无法及时掌握、出现问题无法及时发现和处理结果无法及时了解等一系列“及时”问题,提高政策执行效率和人民满意度,成为目前信息领域研究的重点课题。

在全国精准扶贫、精准脱贫背景下,本文以福建省扶贫(惠民)资金在线监管系统为例,介绍如何运用大数据、云计算、微服务等信息技术解决扶贫对象精准、项目安排精准、资金使用精准等问题,实现对扶贫(惠民)资金全流程精准监控,确保扶贫资金精准发放。

1 技术介绍

1.1 大数据技术

1.1.1 Hadoop

Hadoop 是由 Apache 基金会开发的一个分布式系统架构^[1],由 HDFS (Hadoop Distributed File System, 分布式文件系统)、MapReduce (并行运算编程模型)、HBase (Hadoop Database, 数据库)、ZooKeeper (分布式协调服务)和 Hive (数据仓库架构)等成员组成,提供分布式数据存储和并行处理数据方式,高效实现对海量数据的分布式存储和处理,并为应用程序提供高可靠性透明接口。

其中 HDFS 是专门为海量数据处理分析而设计的高容错性分布式文件系统^[2]。MapReduce 用于计算海量数据^[3,4]。HBase 是建立在 HDFS 之上面向列的分布式存储系统^[5,6],它利用 MapReduce 处理 HBase 中的海量数据^[7,8]。Hive 为数据仓库使用者提供海量数据存储、数据 ETL、数据查询和分析^[9]。ZooKeeper 为分布式应用提供域名、分布式同步等一致性服务^[10]。

1.1.2 Spark

Spark 是专为大规模数据处理设计的快速通用计算引擎,是 Hadoop MapReduce 通用并行框架。它启用了内存分布数据集,不仅能够提供交互式查询,还能优化迭代工作负载,因此能更好适用于数据挖掘与机器学习等需要迭代的 MapReduce 的算法^[11]。

1.1.3 Kerberos

Kerberos 是一种网络认证协议,认证过程不依赖主机操作系统认证,无需基于主机地址信任,不要求网络上所有主机物理安全,并假定网络上传送的数据包可被任意地读取、修改和插入数据。

1.2 Elastic Search

Elastic Search (ES),一种基于 Lucene 构建的全文搜索引擎、数据分析引擎和分布式文档数据库,每个字段均可被索引和搜索,能在极短时间内存储、搜索和分析大量数据,通常用于复杂搜索场景下的全文检索、结构化检索和分析。相对于传统搜索引擎,ES 具有高扩展性、高实时性、高可用性等三大明显优势,通过分片分布处理方式提升处理效能,可横向扩展至数以百计的服务器存储以及处理 PB 级数据。

1.3 微服务架构

微服务是一些协同工作的,具有高内聚性和高自治性的小而自治服务^[12],核心理念在于将复杂应用拆分成多个可单独构建和部署的功能,每个功能称之为服务^[13]。微服务架构旨在实现对复杂应用的快速开发,本质是由一组可独立交付的微服务业务单元构成的分布式系统^[14]。相对于 SOA 架构来说,微服务架构具有复杂度可控、架构灵活、技术多元化、功能易扩展、独立自主等主要特点^[15]。

2 总体建设思路

利用大数据、云计算、微服务等信息技术,建立覆盖省、市、县、乡四级的扶贫(惠民)资金在线监管系统,及时掌握项目和资金动态,实现对资金各环节无死角在线跟踪监督,加强对政策和资金的审计监督检查,消除监管“盲区”,强化信息主动公开,让老百姓由被动参与变为主动监督,让扶贫(惠民)工作在阳光下运行,确保资金精准安全有效使用,实现精准扶贫和脱贫。

3 系统体系架构

考虑扶贫(惠民)资金涉及省、市、县、乡四级财政,为及时获取全省资金动态执行情况,实现对资金的全流程监管,系统采用全省集中部署模式,部署在福建政务云上。系统体系架构从纵横两个方向保障系统安全、稳定、高效运行,纵向分为应用服务层(SaaS)、平台服务层(PaaS)及基础设施服务层(IaaS)等三层,横向包括运维体系和安全体系等两个体系。

其中 SaaS 云应用,基于云架构,微服务构建扶贫(惠民)资金在线监管云应用。PaaS 云平台,提供云中间件服务、微服务治理、运行时应用管理和能力共享中

心等. 以 PaaS 平台为核心构建大数据平台, 并预留规划相关专题技术平台能力, 全面支撑 SaaS 应用. IaaS 云基础设施, 基于 OpenStack 产品, 实现计算、存储、网络资源虚拟化和资源管理功能, 按需为 PaaS 云

平台提供资源. 运维体系和安全体系作为全局公共能力贯穿各个层面, 为整个系统长期稳定运行提供运维和安全保障服务.

系统体系架构如图 1 所示.

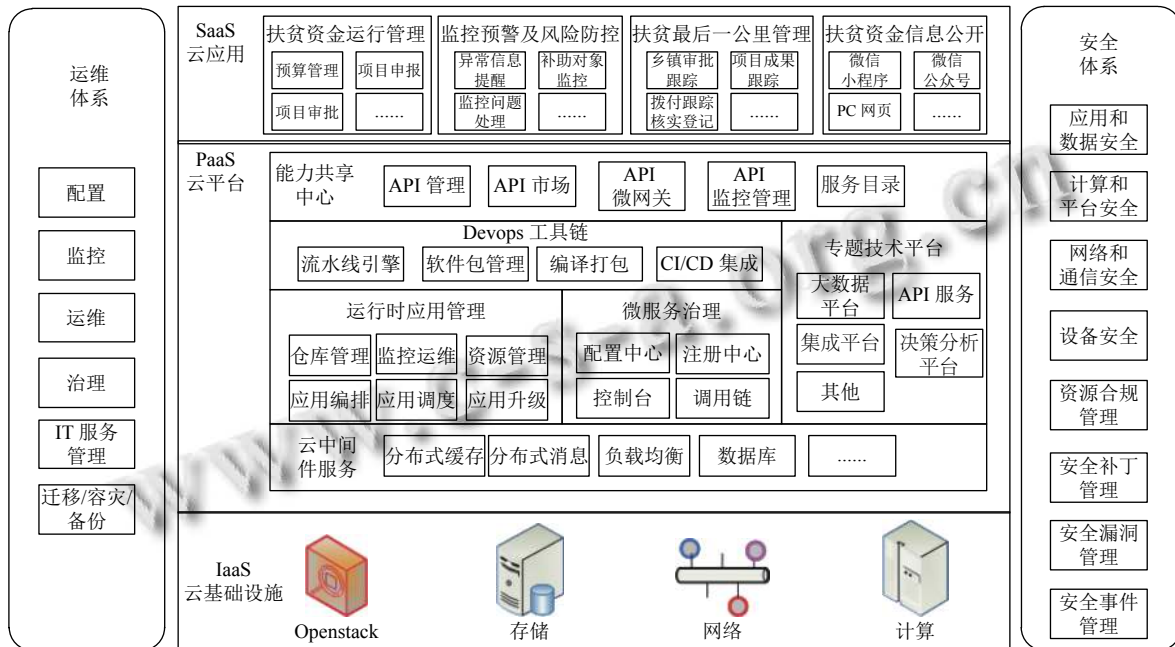


图 1 系统总体体系架构

4 关键技术应用

4.1 大数据技术应用

扶贫(惠民)资金覆盖省、市、县、乡四级财政, 涉及全省所有关心扶贫(惠民)信息的人员. 系统除预算编制, 预算执行等内生数据外, 为防止出现扶贫(惠民)资金冒领骗领现象发生, 系统外接工商、税务、社保、公安、住建、19 家代理银行等部门, 分别获取企业工商登记注册、企业和个人纳税、个人社保、住房购买、车辆购买、银行到账等外生数据, 构建精准扶贫大数据分析库, 后期还将接入火化人口、公积金缴纳等数据, 因此系统数据量和访问量都相当巨大. 为保证系统稳定和时效性, 特别是公众访问系统的响应速度, 系统采用 Hadoop+Hive+Spark+ES 等大数据技术, 确保响应时间控制在 5 ms 以内, 大数据应用示意图如图 2 所示.

如图 2 所示, 系统使用 Hadoop 平台做为基础平台部署, 提供资源调度及 HDFS 存储服务, 采用 Hive 做为数据仓库来存储原始业务数据以及各分层汇总数据. 使用 Spark 做 ETL 处理引擎, 由 Spark 读取 Hive 元数

据做 ETL 清洗、转换及汇总操作. 汇总数据存放在 Hive 数据库的汇总结果表中, 使用 ES 做搜索引擎, 提供查询检索功能, ES 读取 Hive 库结果表数据, 创建索引库. 应用 Nginx 代理服务器等成熟中间件产品, 提高查询效力. 使用 Kerberos 做大数据集群安全框架.

数据展现层分为传统 PC 网站和微信小程序+手机 APP 等两种展现形式. 系统通过 Spring Cloud 微服务提供 Restful 方式的对外数据访问接口, 供展现层调用. ES 提供统一数据查询接口为展现层提供数据服务.

4.2 微服务架构设计应用

为减少数据量和负载因素对系统性能的影响, 系统采用微服务架构, 将系统按功能拆分为多个独立的服务组件; 同时利用云计算技术, 根据负载情况动态合理分配系统资源. 政务云通过公共组件微服务化, 避免业务应用对接公共组件时, 因特定技术要求给业务应用开发带来更多工作量和架构灵活性限制. 各功能均以微服务方式独立部署, 平台、应用、运维能力均实现微服务化, 实现平台和应用彻底解耦.

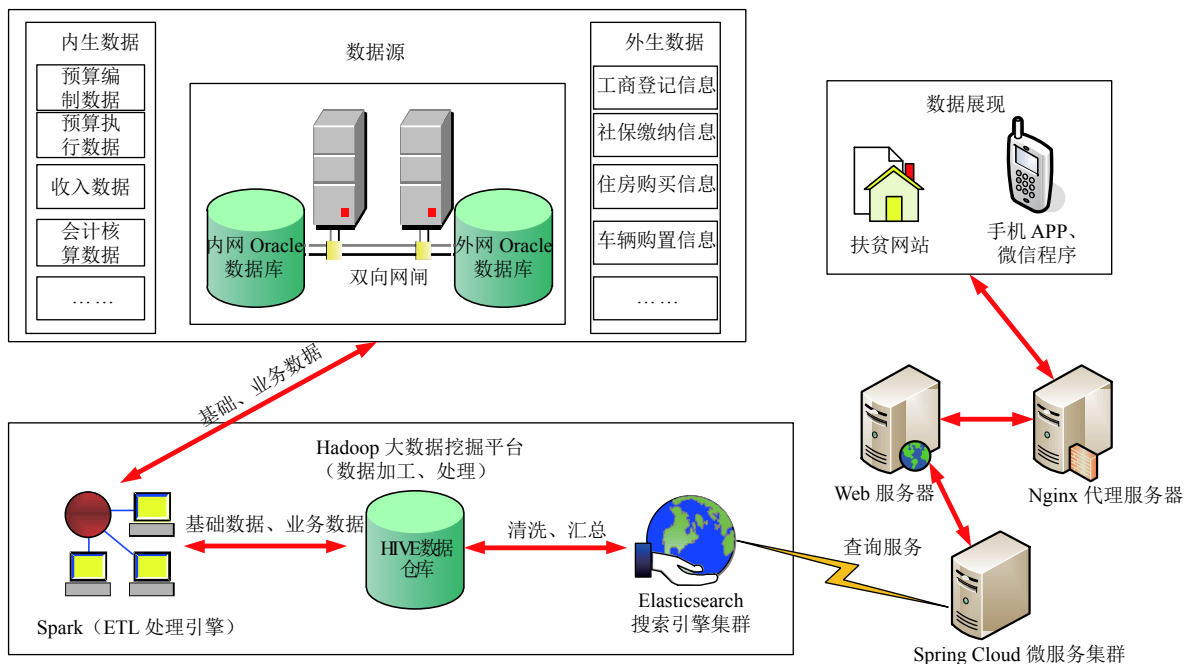


图2 大数据应用示意图

4.2.1 系统分层微服务架构

结合业务管理特征和发展趋势,统一规划业务应用系统,将系统划分为前台、中台、后台3层,每层包含相应的微服务,解耦业务办理、公共支撑、业务管控和决策分析等能力,使它们有效协同。同时系统采用前端轻应用、后端微服务的前后端分离模式,具体系统业务分层微服务架构图如图3。

如图3所示:前台以业务和角色为中心构建作业平台,支撑灵活高效运作。前台连接财政用户、扶贫相关部门、社会公众等3类用户,构建以OA为基础的统一门户,快速响应业务变化和业务应用,支持弹性伸缩。

中台以公共组件为核心为微服务提供支撑,实现业务、技术能力共享化和服务化,支撑前台生产类应用作业平台构建。中台面向前台应用,构建服务中心和API中心,实现服务治理和数据治理;构建通用应用能力,包括公共业务、公共办公、应用支撑等服务,实现业务运作服务化、共享化。

后台面向领导和决策层,实现作业过程与结果透明可视,如资金流可视、业务流可视、扶贫(惠民)对象画像等。后台围绕数据资产,建设统一的数据底座,基于大数据构建决策支持、监控预警、查询分析3类应用,实现业务洞察和监管能力。

4.2.2 微服务应用关键步骤

系统微服务应用关键步骤具体如下:

(1) 构建接口契约优先的运作机制,协同工作。微服务设计坚持契约优先原则,即先有契约后有微服务实现。首先基于Swagger Open API规范定义接口契约,契约以人与机器均可读懂格式描述微服务及其API接口,作为服务提供方对外功能和服务水平的承诺。服务提供方基于契约实现微服务,微服务使用方基于契约调用微服务,各方基于契约协同工作。微服务均可基于契约进行替换,即当微服务出现严重问题时,可按契约无缝替换,不影响系统运行,并形成以微服务为核心的可拆可合、开放的“扶贫(惠民)信息化生态”。

(2) 构建服务中心、API中心,管理和运营“扶贫(惠民)信息化生态”。将微服务注册到服务中心,微服务API接口注册到API中心,通过在线、统一排错和度量评价体系,界定问题边界,识别需要改进或替换的微服务,保证各方有效协作,为“信息化生态”运营提供支撑,机制如图4所示。

如图4所示,服务中心提供服务市场,供管理用户在线浏览、查询服务详情,支持线上评价和提出改进建议;提供微服务治理,通过调用链功能,支持端到端定位排错、界定问题边界和责任。API中心对API接口全生命周期进行管理,管控微服务和接入应用之间的调用关系,并对API接口调用情况进行监控,提供各

种性能指标、异常指标、SLA 指标及运营报表, 全面度量微服务及其 API 接口质量. 通过上述机制, 避免在

公共微服务集成过程中可能出现的问题定位不清、互相推诿等情况, 支持对公共微服务持续改进.

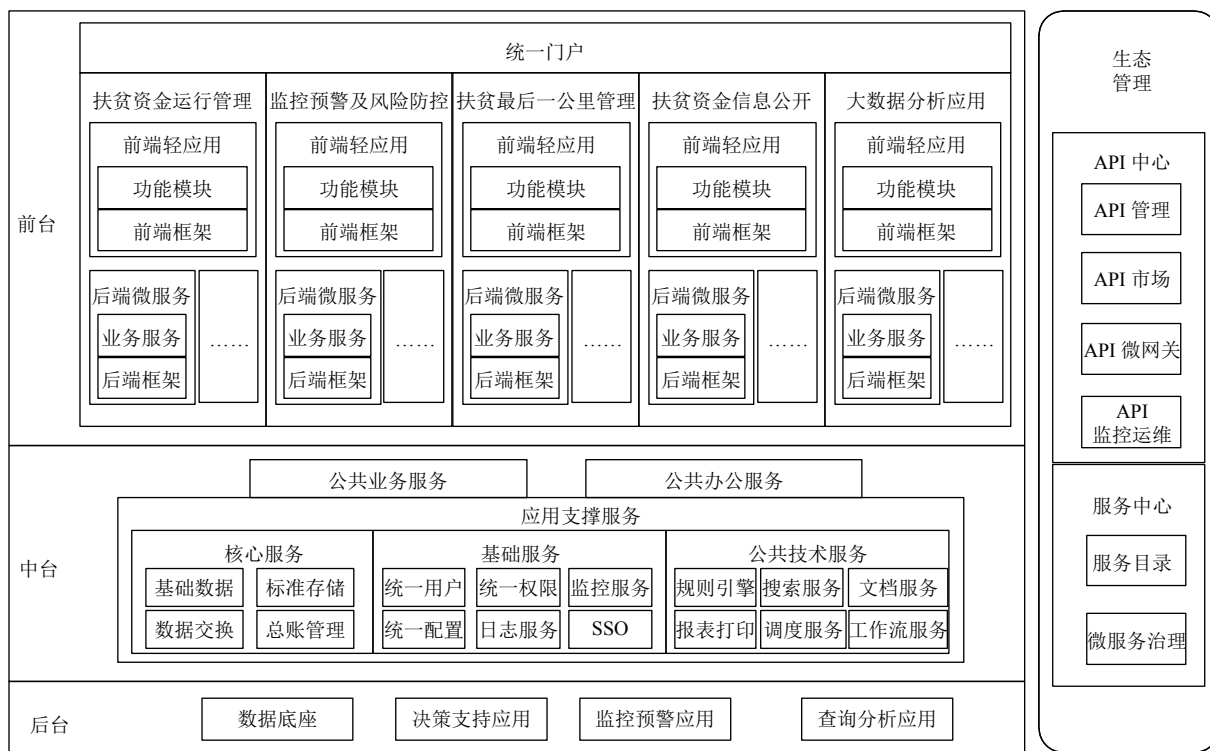


图3 系统分层微服务架构

对于标准存储、基础数据等核心公共微服务, 由用户统一定义微服务接口契约, 并将接口契约发布到 API 中心. 微服务接口契约经用户审核后, 将接口契约发布到 API 中心. 软件开发商根据契约提供微服务, 并由微服务中心、API 中心提供相关度量和改进机制, 最终形成一个完整的生态管理体系.

(3) 构建包括应用支撑服务在内的服务化中台, 规范和支撑业务应用建设公共能力服务化设计, 包括公共业务、公共办公、应用支撑和管控等服务, 最终构建服务化平台, 是业务应用建设的一个关键点.

4.2.3 微服务设计关键要求

在设计微服务时, 需遵循或注意几个原则和要求, 以达到采用微服务最佳效果.

(1) 系统遵循前后端分离原则, 即前端轻量级应用、后端微服务. 前端轻应用采用成熟的前端单页面技术开发, 通过 ajax 异步请求与后台微服务交互, 快速响应业务变化的轻量级应用; 后端微服务为独立部署运行的业务逻辑单元, 通过统一的 API 网关供前端轻应用调用, 数据交换格式为 JSON 格式. 前后端分离优势在于前后端独立打包发布, 相互独立, 互不影响, 便

于系统开发维护, 同时前端页面静态化, 便于使用客户端缓存, 提高系统并发性能.

(2) 遵循业务驱动原则, 将业务应用分解为微服务.

(3) 微服务治理要求, 所有业务微服务统一注册到服务中心. 所有微服务接口统一采用 REST 通信协议, 均注册到 API 中心. 微服务须采用平台统一用户、角色、权限管理体系, 统一 SSO 认证.

(4) 运维服务化设计, 依靠服务化工具链, 在运维阶段实现高效服务化运维. 前后端应用实现容器化部署, 所有配置通过统一配置服务获取.

5 实现成果

5.1 全省扶贫部门、人员全覆盖

按照“横向到边, 纵向到底”要求, 系统横向覆盖扶贫(惠民)资金涉及的 12 个主管部门, 并与福建省内 19 家商业银行实现数据对接; 纵向贯穿省市县乡四级, 包括 9 个设区市和平潭综合实验区、83 县(市、区)、1105 个乡镇. 目前系统操作用户达 1.8 万人, 帮扶对象涉及 1.7 万个村居、628 万人.

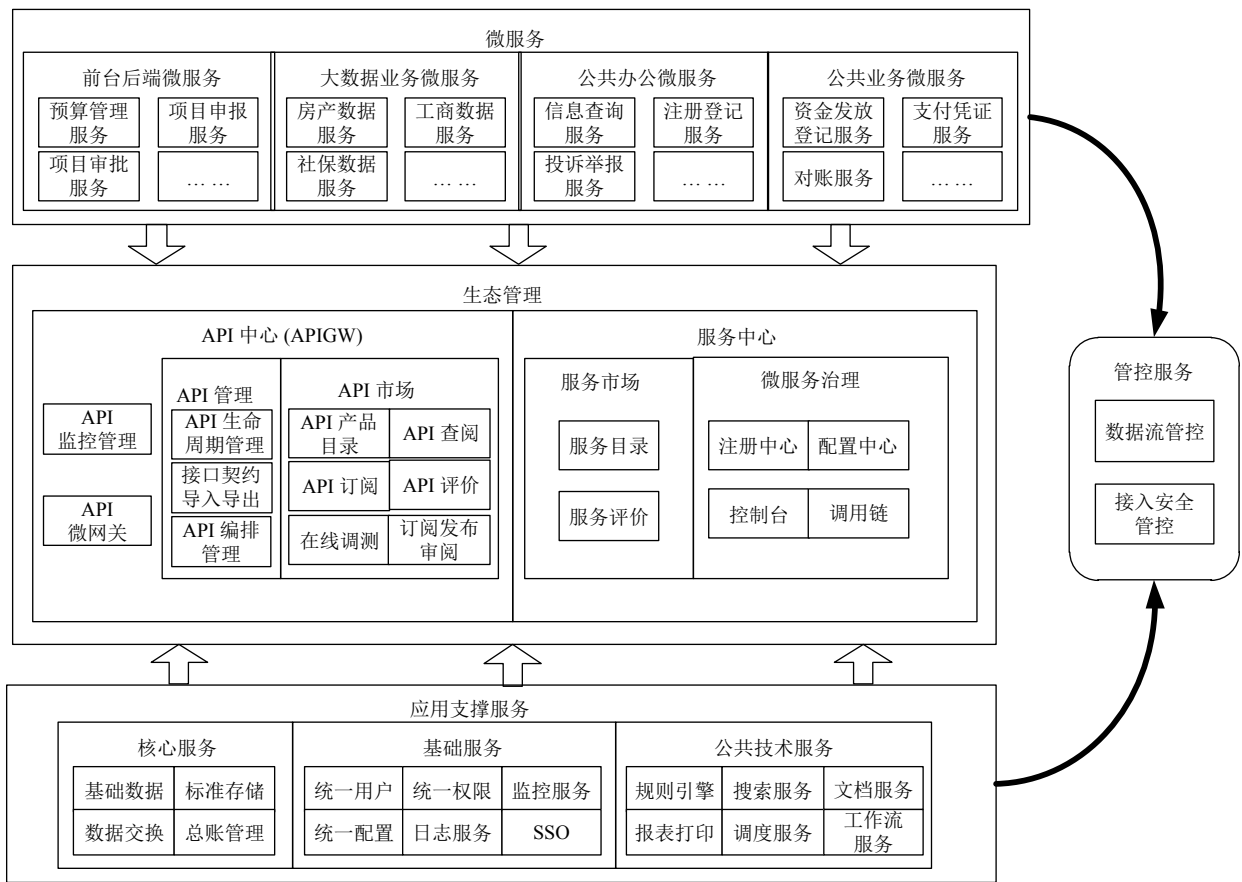


图4 扶贫(惠民)信息化生态管理图

5.2 资金流程监督全覆盖

系统对扶贫(惠民)资金的流向、流量和流速进行全方位监督。其中:流向监督主要是运用工商登记、养老保险、医保、税收、车辆、住房等数据,对建档立卡贫困户、低保户进行精准识别。流量监督主要是将银行反馈到账数据与乡镇填报的应发放数据比对,确保应发尽发,足额到位。流速监督主要是动态监控资金分配、审批、下达全过程运转时效,确保资金及时发放到位。

为及时从系统中发现资金问题节点,按照“共性+个性”思路和各类扶贫资金管理要求,通过设置预警规则、预警阈值,按预警级别对资金项目申报、审批、下达等环节进行实时动态查验。根据严重程度不同,将预警级别分为红黄两种颜色,红色表示严重预警,包括发放对象不精准和发放金额与银行反馈数据不一致等情况。黄色表示一般预警,包括资金下达不够及时、银行反馈数据不完整等情况。

2018年系统对21项扶贫资金、2项救灾资金实

行全流程监管,涉及资金40亿元,惠及118万帮扶对象;2019年系统增加对14项惠民资金监管,涉及资金70.1亿元以上,惠及614万帮扶对象。

图5及图6为福建省扶贫(惠民)资金监管结果图。



图5 扶贫(惠民)资金监管总体图

5.3 扶助对象识别更加精准

系统通过大数据技术核查建档立卡贫困户信息和资金发放信息是否准确无误,对于在城市务工获取社保、购买房屋、拥有车辆或成立公司等可能已经脱贫人员,经核对查实后,将从建档立卡贫困户名单中移除

并相应处理;对于未准确发放或者核实为冒领、骗领的,纪委将根据程序移送司法机关。2018年系统上线以来,通过大数据识别对比,发现大量贫困低保户人员存在异常信息,主要为系统显示贫困低保户有注册为公司法人或在企业缴纳社保金额已经超出低保范围等。经核实,大部分注册为公司法人的情况为其亲戚或朋友借用其身份证注册公司,本人不知情或并未从中获取利益,对于这种情况则要求本人到工商局注销公司法人身份;目前系统发现低保对象异常信息 10 176 条,由省民政厅分类核实身份信息,确定继续保障对象 4337 人,延保渐退 2147 人,取消退保 3692 人。以每低保户每月 300 元计算,每年将节省财政资金 2000 多万元,维护了公平和正义,大大提高了政府公信力和群众满意度。



图 6 扶贫(惠民)项目绩效主要产出图

5.4 全民参与监督

为实现全民参与监督,系统将专项管理办法、项目信息、审批信息、补助资金发放信息等通过网站和手机 APP、微信小程序等进行全面有效的信息公开。为方便群众及时知道和下载微信小程序和手机 APP,通过短信、电视、二维码等多种途径向全省公众推送。为方便群众发现问题能及时反馈,系统通过接口与省纪委监委举报网站对接,实现一键举报。目前微信小程序和手机 APP 访问量超两千万,日均访问量超 10 万,网站、微信小程序和手机 APP 样例如图 7。



图 7 数据展示信息

5.5 精准扶贫试验田

本系统建设获得国务院和财政部的充分肯定,财政部指出该系统建起来后,有利于明晰责任,是财政管理的重大成果和制度创新,以后各方面资金都可以用来监管,因此系统带有“试验田”的作用,并以本系统为样板建设了全国财政扶贫资金动态监控平台,在全国推广应用。

6 系统建设经验

6.1 数据库选用

系统在完成微服务改造并部署到云上后,能否利用云计算优势,根据负载情况动态合理地为微服务分配系统资源,提高系统并发性和时效性,还有一个关键因素,即是否采用支持云化服务的数据库,这是很多人忽视的地方。在分别使用传统数据库和支持云化服务数据库时,并发量和系统性能之间会呈现一定关系。使用传统数据库时,系统在性能不影响的情况下,并发数最多只能达到 300 个,如果超过 300 个,系统响应速度急剧下降,系统出现不动或者白屏状态。但使用支持云化服务数据库时,系统在性能不影响的情况下,并发量可以达到 2000 个。目前政府机关建设大型信息化系统大部分使用 Oracle 数据库,但 Oracle12C 数据库之前的版本均不支持云化服务,因此如果要到达理想的并发性和时效性,需要使用 Oracle12C 及之后的版本。

6.2 微服务最佳颗粒度

目前微服务概念在国内兴起也好几年,但真正将微服务用好的大型政务系统,在国内几乎没有,大部分都处于尝试阶段,包括本系统在内。从目前已经尝试采用微服务架构的国内某些系统来看,效果不一定优于采用 SOA 系统架构的系统,主要原因是大家仅在概念上理解微服务,微服务到底微到什么程度最为合适,效果最好,目前没有明确说法,也很难判断。本系统在建设初期,为防止出现因微服务颗粒度选择不合适导致大量无用功情况的发生,在此先开展微服务颗粒度合适程度测试,并使用 4 种颗粒度分法。

第 1 种颗粒度分法,将每个细小的功能环节,比如预算指标下达,计划审核,公文起草等,都设计成微服务。按照这种分法,本系统需设计的微服务数量将达到几千个。

第 2 种颗粒度分法,变大微服务颗粒度,对每一个完整的功能服务,比如预算指标服务,对账服务,“小额

贷款”资金申报服务等设计成微服务,采用该种分法,微服务数量缩小到不到100个。

第3种颗粒度分法,继续变大微服务颗粒度,对每一个完整的管理服务,比如专项资金项目管理,专项资金分配管理,系统设置服务,监控预警和风险控制服务,外网信息公开服务,OA管理服务等设计成微服务。依照此种分法,微服务数量将进一步缩小到仅15个。

第4种颗粒度分法,综合第2和第3种颗粒度分法,将并发量大,访问频繁模块按照第2种颗粒度拆分,比如预算指标服务和对账服务等;对于并发量小,访问不是太频繁的功能模块按照第3种颗粒度拆分,比如系统设置服务和OA管理服务等。这种拆分方式,微服务数据量可以控制在50个以内。

图8为采用Oracle12C数据库,不同颗粒度微服务与并发量、系统性能(用系统响应时间代表系统性能)之间关系图,其中横坐标表示并发量,纵坐标为系统响应时间,曲线A~D分别代表第1~4种颗粒度。

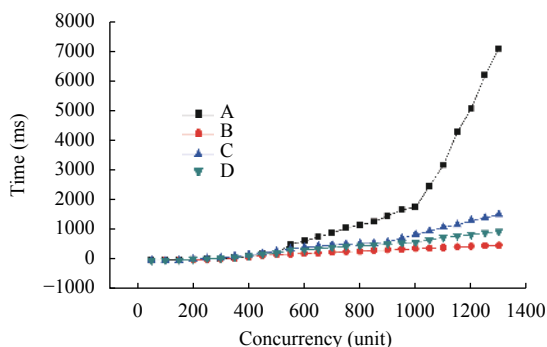


图8 颗粒度与并发量关系图

从图8可以看出,在并发量不超过500时,4种颗粒度的系统性能差距不大。但是超过500后,系统的性能和并发量并没有因为颗粒度越细而使系统性能越好;相反,采用第一种颗粒度,微服务最多,但是系统性能不仅没有明显提高,反而表现最差,在达到1000并发量后,系统性能迅速变差。采用第2、3、4种颗粒度的系统性能和稳定性明显高于第1种颗粒度,其中系统性能最好的是采用第2种颗粒度的,在并发量达到2000时,系统性能依然在1s范围内,其次是第4种,介于第2种与第3种之间。

随着颗粒度的增加,系统开发时间将会变长,同时部署难度也将随着增大。在综合考虑开发成本、时间效率以及后期系统在政务云平台部署难度等一些列因

素后,本系统最终决定采用第4种颗粒度分法。因此在选择微服务颗粒度大小时,要根据系统实际业务场景情况来划分,到底要拆的多细,绝不仅仅只是个技术问题,而是一个技术和业务理解相结合的问题,一定要选择适应自己的颗粒度,才能达到即节省成本和资源,又能充分发挥微服务优势的建设效果。

6.3 微服务容器使用

在微服务实际部署过程中,为了能够真正实现随着需求的变化自动调配微服务数量,更好的发挥微服务效果,还必须使用微服务容器对微服务进行管理和监控。

目前普遍使用的是Docker微服务容器。使用微服务容器不仅可以实现自动调配微服务数量的目的,还可以通过管理界面直观看到每个微服务数量和运行健康状态,容器的使用大大提高了微服务的管理效率。微服务容器以集群的方式部署,让系统服务部署变得简单、高效。如果不使用微服务容器,则需要在每台服务器上安装运行环境,如果需求的服务器数量庞大,在每台服务器上安装运行环境将是一项无比繁重的工作,一旦运行环境发生改变,就不得不重新配置。而使用容器技术,微服务是以镜像的形式运行在容器中,只要将所需的基础镜像和微服务生成一个新的镜像,将这个最终镜像部署在容器中。在此创建一个镜像仓库用来存放所有的基础镜像以及生成的最终交付镜像,在镜像仓库中对所有镜像进行管理。

7 小结

实现全国精准扶贫和精准脱贫,是完成中央“两个一百年”奋斗目标的前提条件。如何解决扶贫对象精准、项目安排精准、资金使用精准等一些列精准问题,及时发现违规违纪行为,确保扶贫资金精准、及时发放,成为政府部门亟需解决的问题。福建省积极响应中央精神,利用大数据、云计算、微服务等前沿先进技术,在全国率先研究开发福建省扶贫(惠民)资金在线监管系统,实现对扶贫资金的全流程精准监控,确保扶贫资金精准及时发放。系统建设成果得到国务院和财政部的充分肯定,并获得第二届数字中国建设峰会数字福建电子政务十佳案例成果奖,同时在全国推广使用,为我国早日实现精准脱贫奠定了基础。本文详细描述了该系统的体系架构和关键技术,对实现效果和建设经验进行了分享,该系统建设思路可以为其他政府部门解决其他领域建设问题提供参考和借鉴。

参考文献

- 1 陆嘉恒. Hadoop 实战. 北京: 机械工业出版社, 2011.
- 2 Taunk AK, Parmar AK, Srivastav R. The Hadoop distributed file system. *International Journal of Computer (IJC)*, 2013, 8(1): 8–15.
- 3 冷芳玲, 鲍玉斌, 于戈, 等. 基于 Map Reduce 的封闭数据立方. *计算机研究与发展*, 2011, 48(S1): 610–616.
- 4 陈吉荣, 乐嘉锦. SingleMapReduce: 单一输出 HDFS 文件的 MapReduce 编程模型. *华南理工大学学报 (自然科学版)*, 2014, 42(5): 135–142. [doi: [10.3969/j.issn.1000-565X.2014.05.021](https://doi.org/10.3969/j.issn.1000-565X.2014.05.021)]
- 5 王海荣, 刘珂. 基于 Hadoop 的海量数据存储系统设计. *科技通报*, 2014, 30(9): 127–130. [doi: [10.3969/j.issn.1001-7119.2014.09.031](https://doi.org/10.3969/j.issn.1001-7119.2014.09.031)]
- 6 张榆, 马友忠, 孟小峰. 一种基于 HBase 的高效空间关键字查询策略. *小型微型计算机系统*, 2012, 33(10): 2141–2146. [doi: [10.3969/j.issn.1000-1220.2012.10.005](https://doi.org/10.3969/j.issn.1000-1220.2012.10.005)]
- 7 贾少攀. 基于 Hadoop 的 XBRL 数据分析系统设计与实现 [硕士学位论文]. 大连: 大连理工大学, 2013.
- 8 邓祥. 基于 Hadoop 的海量日志数据处理研究与应用 [硕士学位论文]. 厦门: 厦门大学, 2013.
- 9 Thusoo A, Sarma JS, Jain N, *et al.* Hive: A warehousing solution over a map-reduce framework. *Proceedings of the VLDB Endowment*, 2009, 2(2): 1626–1629. [doi: [10.14778/1687553.1687609](https://doi.org/10.14778/1687553.1687609)]
- 10 Djafri L, Bensaber DA, Adjoudj R. Big data analytics for prediction: Parallel processing of the big learning base with the possibility of improving the final result of the prediction. *Information Discovery and Delivery*, 2018, 46(3): 147–160. [doi: [10.1108/IDD-02-2018-0002](https://doi.org/10.1108/IDD-02-2018-0002)]
- 11 Xing ZC, Li GM. Intelligent classification method of remote sensing image based on big data in spark environment. *International Journal of Wireless Information Networks*, 2019, 26(3): 183–192. [doi: [10.1007/s10776-019-00440-z](https://doi.org/10.1007/s10776-019-00440-z)]
- 12 赵军, 陈子晗, 高子航. 基于微服务架构的一体化科研管理平台设计与实现. *无线电工程*, 2019, 49(5): 436–441. [doi: [10.3969/j.issn.1003-3106.2019.05.16](https://doi.org/10.3969/j.issn.1003-3106.2019.05.16)]
- 13 辛园园, 钮俊, 谢志军, 等. 微服务体系结构实现框架综述. *计算机工程与应用*, 2018, 54(19): 10–17. [doi: [10.3778/j.issn.1002-8331.1808-0014](https://doi.org/10.3778/j.issn.1002-8331.1808-0014)]
- 14 刘劭. 基于微服务的教学支持平台服务端的设计与实现 [硕士学位论文]. 南京: 南京大学, 2018.
- 15 徐康明. 基于微服务架构的服务发现与服务可靠性的研究 [硕士学位论文]. 北京: 北京邮电大学, 2018.