

信息系统柔性重组技术研究与实现

焦萌萌 孙晨冉 代月

中国电子科技集团公司第十五研究所

DOI:10.12238/acair.v2i3.8611

[摘要] 随着计算机技术的高速发展,各行各业都实现了线上办公,软件的功能也是越来越多、越来越强大,不同的使用场景对软件功能的需求自然也就不同,所以软件的可重复利用值得大家研究。“框架+插件”的软件设计,能够实现软件柔性重组,最大化的重复利用,实现信息系统在不同使用场景下按需重组、灵活配置。

[关键词] 集成框架; 框架+插件; 柔性重组; 服务管理

中图分类号: TU378.4 **文献标识码:** A

Research and implementation of flexible recombination technology of information system

Mengmeng Jiao Chenran Sun Yue Dai

China Electronics Technology Group Corporation 15th Research Institute

[Abstract] With the rapid development of computer technology, all walks of life have realized online office, the function of software is more and more powerful, different use scenarios of the software function of the demand is naturally different, so the reuse of software is worth studying. The software design of "framework + plug-in" can realize the flexible recombination of software, maximize the reuse, and realize the recombination and flexible configuration of information system in different use scenarios.

[Key words] integration framework; Framework + plug-in; Flexible recombination; Service management

软件构架是软件系统的核心,它对于整个软件系统的设计开发和使用具有深远的影响。软件构架并不会随着某个软件系统生命期的结束而终结。目前,大多数的软件设计者一直接受的是完全按照系统的技术需求表述进行系统设计,而忽略了软件构架对于整个软件的作用。本文对软件构架进一步描述,目的在于引起软件开发人员对于软件构架的重视。

1 设计原则

系统软件设计贯彻“1+X”能力建设思路,主要考虑以下方面:

(1) 1套平台,多类型应用支持。系统软件为办公信息系统提供任务策划与分发、导调控制及监视等功能,应具备使用人员不同软件功能也不同的能力。从软件系统架构、业务系统构建、业务支撑环境等方面进行全面考虑和精心设计,需要构建能够支撑多类型应用的软件平台,包括消息总线、数据访问等基础服务、服务管理和插件集成等,平台需具有高度的灵活性和高效扩充性、扩展性,并且随着各种资源(模型、模板、服务、流程、数据等)的积累,不断提升平台能力,实现平台持续演化。另一方面,制定相应的集成规范、数据交换规范等,使得各类应用能够基于统一的软件平台进行规范化开发,易于集成,灵活运用。

(2) 1套系统,多任务并行支持。系统软件面向多人员、多职

责的需求,采用一体化设计,针对任务制定与分发、综合监控、数据采集、分析评估等功能,统筹共性、适配差异,进行客户端和服务功能的综合设计,形成办公信息系统。根据任务需要,按需组织、灵活编配,从系统中快速构建出定制化的软件系统,有效实现1套系统支持多任务并行的能力。

(3) 1套系统,多系统适配。按照不同的系统应用需求,统筹考虑全部需求,进行全集功能开发,按照组件化封装、服务集成技术实现功能重组、业务系统重组和系统优化配置,满足信息系统在不同使用场景下按需重组、灵活配置需求。注重面向服务,实现易扩展和提升。信息系统设计过程中通过服务编排、服务重组等技术,实现功能流程可重组、服务构件按需无缝插接和调用,同时对信息系统提出的功能升级、业务扩展和能力提升等要求预留接口。

2 软件技术架构设计

为了提升软件开发效率,同时提升系统的稳定性等,系统软件总体架构采用微服务架构,通过分解大业务处理逻辑,将复杂处理拆分到多个服务中实现。

又考虑到办公软件功能柔性重组和扩展需求,需要使功能组件化,并能方便地实现功能扩展。因此在前端桌面应用采用“框架+插件”技术,搭建统一的前端集成框架,并开发基于框架

的各应用插件,通过插拔组装插件形成功能不同的办公系统专用软件;前端Web应用采用“框架+业务页面片段”技术,提高界面渲染效率,提升界面显示速度,同时通过灵活组合页面片段,实现界面功能的多元化展示。

综上所述,系统软件技术架构模式定义为“微服务+集成框架”。前端应用开发可采用QT,基于统一的集成框架,开发功能插件;Web应用采用基于Java的SpringMVC框架开发,基于统一的Web展现框架,开发业务片段页面;服务采用基于Java的SpringCloud框架开发。

系统软件按照界面、服务和数据独立设计、集成融合的技术架构,分离界面、服务和数据的硬关联,降低界面、服务与数据的耦合性,减少数据、服务和界面关联带来的系统臃肿、冗余、效率不高和操作展现复杂等问题,面向最终用户构建精简、高效、可按需配置的信息系统软件。

(1)数据的独立设计,可向各业务功能域提供一致的数据信息,避免各业务要素对数据的理解不一致性;提供统一的数据访问接口,隔离不同数据库的差异,数据访问层采用主流开源技术MyBatis进行数据访问层的开发,该技术能够采用配置化的方式对SQL进行管理,又具备ORM(对象与关系模型映射)的能力,既有Hibernate等ORM框架的易用性,又能对SQL进行优化配置获得较高的性能。

(2)服务的独立设计,可实现业务逻辑、模型算法的精细化构建,按照各业务信息流程和逻辑,精确化组装信息系统;通过统一的微服务治理框架实现对微服务集群的统一管理,实现业务服务的统一注册、统一监控和服务的订阅;通过服务接口调用隔离客户端对业务逻辑的依赖,实现轻量的客户端实现,将复杂的业务隔离到服务端;通过消息总线接口实现消息的异步发送,同时实现异构系统之间的信息交互;通过对不同业务服务的组合部署,实现典型的应用系统部署模式,满足不同的应用场景;通过对业务服务的水平扩容可以实现系统的性能提升。

(3)界面的独立设计,可向用户提供界面展现的深度定制、个性化定制能力,实现功能的动态调整和柔性重组,有利于向用户提供符合实际使用和操作需求的界面展现,提高系统操作效率。针对桌面和Web两类应用,分别通过客户端集成框架和Web展现框架提供的注册加载、界面展现、事件调用等接口,集成业务插件和业务展示页面,构建完整的桌面应用软件和Web应用软件。

3 前端集成框架技术设计

3.1 作用与用途

前端业务柔性重组,是提升软件信息系统使用灵活性的重要手段。由于信息系统定位不同,针对不同使用场景,各个软件的功能有所侧重,因此不能完全固化软件功能,而是应该具备一定的柔性重组能力,根据使用场景要求,灵活配置信息系统的软件功能。

集成框架技术是实现业务柔性重组的技术手段,同时也更有利于系统模块化开发和系统综合集成。因此信息系统的前端

业务应用采用集成框架技术进行开发,按照应用形态分为桌面应用和Web应用两类。

采用“即插即用”软件开发技术,把前端业务应用软件分成框架和插件两个部分。框架与插件能够互相通信,而且在框架程序不变的情况下,可以通过增减或修改插件来调整软件功能。

框架所完成的功能为系统的核心和基础,这些基本功能既可为用户使用,也可为插件使用,因此把框架基本功能分为两个部分,内核功能和插件处理功能。框架的内核功能是整个系统的重要功能,提供基础的通用功能。框架的插件处理功能用于扩展框架和管理插件,为插件操纵框架和与插件通信提供标准框架扩展接口。

框架定义了两个标准接口,一个为由框架所实现的框架接口,一个为插件所实现的插件接口。框架接口完全由框架实现,插件只是调用和使用,插件接口完全由插件实现,框架只是调用和使用。框架接口实现插件向框架方向的单向通信,插件通过框架接口可获取框架的各种资源和数据,插件接口为框架向插件方向的单向通信,框架通过插件接口调用插件所实现的功能。

3.2 前端集成框架功能模块组成和描述

(1)插件管理。整体管理所有插件,负责所有插件的载入、验证、初始化、释放、插件命令执行等。其中,插件命令执行功能,能将菜单、状态栏等触发动作与插件中定义的执行命令相关联,执行特定的功能。

(2)消息管理。软件集成框架通过插件管理接口,支持插件与插件之间、插件与框架之间的数据和事件传递。软件集成框架提供插件与插件之间传递数据接口,支持插件向其他插件传递内存数据,数据格式由两个插件之间约定。软件集成框架提供事件接口,支持插件向软件集成框架、获取菜单状态、控制界面显示隐藏、鼠标左右键消息、键盘按键消息等功能。

(3)插件组装。由插件定制和插件组装两部分组成。插件定制服务于插件组装,提供单个插件的配置功能,生成单个插件的配置文件。插件组装基于插件配置,能够将多个插件组装成应用软件,为应用软件的定制界面生成配置文件。

(4)部署升级。提供软件部署升级功能。提供软件部署、配置功能,实现无人值守状态下的软件插件升级替换和不打扰用户操作情况下的软件在线更新。

(5)界面布局管理。软件集成框架启动时,读取插件的界面描述文件,根据配置文件提供的详细的配置属性,进行统一布局。

(6)服务调用。提供统一的服务接口和消息总线服务接口,支持功能插件与服务网关进行通信,获取服务实例;功能插件与服务、功能插件之间订阅/发布异步消息传递机制。

4 Web展现框架设计

Web框架采用主流的MVC架构的SpringMVC框架进行开发,搭配SpringBoot自动装配功能(采用习惯优于配置和自动配置的策略),实现Web应用的快速开发。同时,基于SpringBoot的开发框架,支持轻量级和嵌入式容器部署,方便部署管理。在设计上,

采用基于用户和角色的方式实现界面的动态组合,通过授权管理实现不同用户的不同权限管理。在界面展示方面,采用开发界面展现集成框架加页面片段的方式进行开发,以提高界面渲染效率,提升界面显示速度。

Web应用的集成采用开发界面展现集成框架加页面片段的方式进行集成,即在总体结构上把界面分为界面展现框架(包含标题栏、菜单导航栏、状态栏和业务显示区)和业务界面片段两大部分,框架负责加载公共部分并提供业务界面片段集成区域,业务界面片段负责嵌入界面展示框架并显示业务视图,通过菜单导航栏来进行不同业务视图的展示。菜单导航栏可以通过导航栏配置界面进行配置,来实现不同业务的界面集成,通过用户和角色的限制来实现不同业务的灵活组合。

5 服务管理框架设计

5.1 作用与用途

微服务管理框架管理所有服务注册和发布;消息总线负责服务之间、服务和应用之间消息通信;基础服务为业务服务和前端应用提供基础通用功能支持,如数据汇集、数据分发、数据信息处理、通信录、信息分发服务等;业务服务是支撑前端应用的业务处理和计算等,业务服务即根据不同的应用场景开发设计不同的服务。

信息系统服务集成采用基于容器编排的方式进行集成,通过docker Compose file来定义业务所需的服务集合,在编排文件中定义每一个业务服务的名称、服务镜像的地址、服务部署策略(实例部署个数、重启策略、部署机器限制等)、服务暴露的端口、服务使用的网络等,同时还可设置服务运行时的环境变量、服务依赖的服务等。

5.2 服务管理框架模块的功能组成和描述

(1)服务注册发现。微服务架构是由一系列职责单一的细粒度服务构成的网状结构,服务之间通过轻量机制进行通信,服务的提供方要注册报告服务地址,服务的调用方要能发现目标服务。服务的注册发现功能就是提供对服务信息的注册与发现,

对于已发布的服务进行健康监测,更新注册列表。

(2)服务配置管理。微服务有很多依赖配置,某些配置参数在服务运行期间可能还要动态修改,传统的使用xml、yaml等配置文件打包在应用内的方式已经无法满足微服务架构模式的需求,因此需要一个动态的配置中心来统一管理微服务的配置信息。服务配置管理的功能是提供动态的配置中心来统一管理微服务的配置管理信息。

(3)服务访问管理。服务的访问管理功能提供统一的服务访问接口,具备服务动态路由、限流容错、身份认证和安全控制等功能,使得微服务集群提供统一的访问入口,隔离客户端与内部系统之间的关联。

(4)服务监控管理。服务监控管理的功能主要是提供对服务状态监控、服务调用链监控、运行日志监控和健康检查和告警等。

(5)服务部署管理。服务部署管理的功能主要是提供对服务集群的统一编排和部署,提供典型应用的编排模板。

6 结束语

按照集成框架+微服务的方式进行软件设计,可以有效避免软件设计师的重复设计与开发,框架重复利用,也增加了软件的可靠性和稳定性,插件以及微服务的个性化定制就能实现软件功能的重组。

[参考文献]

- [1]钟声,软件构件框架技术与软件系统集成研究[D].海南省,海南大学,2010-04-27.
- [2]许坤,沈伟,邓洲.基于软件总线技术的管控软件框架设计[J].舰船电子对抗,2019,42(06):95-98.
- [3]陈武,汪骥宇,程植.基于框架设计开发微服务项目的迭代与重构研究[J].电脑知识与技术,2023,19(16):32-34.

作者简介:

焦萌萌(1991--),女,汉族,河北衡水人,本科,工程师,研究方向:计算机软件。