

Construction of an Integrated Management and Control Platform Based on “Many-to-Many” Services (Postprint)

Authors: Zhang Jinsong

Date: 2023-10-08T00:00:00+00:00

Abstract

Establish a comprehensive system management and control platform, achieve service modularization, enable unified scheduling by the control platform, and construct a novel content management and publishing system. Transform the original “one-to-one” management model from backend to frontend into a “many-to-many” management paradigm, wherein through configuration of the control platform, multiple modular backend services simultaneously provide services to multiple clients and websites, thereby supporting the management and publishing for multiple clients and websites.

Full Text

Construction of a Comprehensive Management and Control Platform Based on “Many-to-Many” Services

Abstract: This paper discusses the construction of a comprehensive system management and control platform that achieves service modularization and unified scheduling, establishing a new content management and publishing system. The platform transforms the traditional “one-to-one” management model from backend to frontend into a “many-to-many” model. Through platform configuration, multiple modular services in the backend simultaneously serve multiple clients and websites, supporting their management and publishing.

Keywords: microservices; distributed deployment; multi-client publishing

Chinese Library Classification Number: TP316.8

Document Code: A

Article ID: 1671-0134(2017)12-052-02

DOI: 10.19483/j.cnki.11-4653/n.2017.12.014

Author: Zhang Jinsong

In recent years, the development of mobile internet and new media technologies has spawned numerous clients across media, commercial, government, and public service sectors. A newspaper group may operate multiple news clients and websites, often developed by different vendors. This leads to redundant system construction, integration challenges, and increased difficulty in managing business and data flows.

The primary objective of building the management and control platform is to create a flexible and efficient next-generation publishing architecture. It provides construction standards and interface specifications for client and website products, unifies system and business access management, and eliminates information silos and stovepipe systems. The platform supports third-party service integration, enabling service modules to serve both in-house and partner clients. Through client-server decoupling, modules can be independently upgraded and replaced. The platform offers excellent compatibility, allowing services to be freely upgraded, replaced, and plugged/unplugged. This enables dynamic, smooth, and convenient system upgrades and expansion without client updates. New features can be published to clients by combining various business services for rapid development. Concurrent user capacity is not limited by the application system itself; network bandwidth and server performance constraints can be addressed through CDN and distributed deployment.

The platform supports management of multiple publishing terminals and websites, which can be administered independently. Each terminal and website maintains its own domain name, columns, users, etc. Different terminals and websites can use distinct domain names for relatively independent management within the same backend, with support for invoking multiple backend modules. Different users can be assigned to each site, with separate administrators authorized to manage and publish content, meeting multi-level management needs for organizations of various scales. The platform also supports data sharing between different terminals.

Scalability and interconnectivity are thoroughly considered to meet current network and hardware resource requirements, support system expansion and secondary development, and accommodate potential future business needs. The platform features version control for access systems, displaying new version capabilities in an app store model with online upgrade and one-click update functions. Third-party systems develop version upgrades according to platform interface specifications. Each module maintains an independent version number with real-time query support, enabling parallel operation and switching between old and new versions.

1. Overall Architecture

The management platform architecture follows existing workflows while addressing specific requirements for simplicity and flexibility in system management and operations. Leveraging advanced mainstream technical architectures ensures

system robustness and upgradability. The overall system architecture design is shown in Figure 1 [Figure 1: see original paper].

1.1 Modularization, Distributed Deployment, and Version Control

The platform adopts modular, component-based, and object-oriented design, providing robust integration and expansion capabilities for embedded integration of diverse resources to meet future system management, upgrade, expansion, and access requirements. Distributed deployment allows module operational functions to be configured and deleted as plugins, with any module's functions distributable across server groups. Modules can be independently upgraded and replaced.

1.2 Microservices Architecture

Microservice Architecture is an architectural pattern that advocates dividing monolithic applications into small, coordinated services that deliver value to users. Each service runs in an independent process, communicating via lightweight mechanisms. Services are built around specific business capabilities and can be independently deployed to production environments. Supporting technological diversity, microservices provide elasticity, scalability, easy deployment, and convenient composition and optimization. The management platform has adopted microservices architecture throughout design, development, testing, deployment, and operations.

APP server businesses are decomposed into independent services that actively register with the service directory center upon startup. Third-party services also register with the directory center after interface isolation through service proxies. The APP side accesses the authorization center via certificate keys to obtain tokens with different permissions. Using these tokens, it requests service addresses from the directory center (the service discovery process). After permission approval, the directory center returns the actual REST service address, which the APP side uses with the token to access corresponding services (Figure 2 [Figure 2: see original paper]).

1.3 Interface Specifications

The platform establishes interface interaction specifications that all third-party systems must follow. Specifications cover system configuration, administrator authentication, version control, data transmission, user authentication interaction, and log transmission. The interface provides enterprise-level support for external system access, ensuring secure and reliable access under high concurrency and large capacity while supporting dynamic resource expansion. It enables smooth system migration and rapid, convenient, accurate implementation during scaling or new business expansion.

The interface supports synchronous/asynchronous request/response, sessions, broadcast notifications, event subscriptions, and reliable message transmission.

Comprehensive information security and monitoring mechanisms ensure information protection and real-time operational monitoring, preventing massive access and resource occupation to guarantee system robustness.

Any module supports interface management with external forwarding and authentication. Interface routing functions enable configuration of protocols, request methods, data formats, parameter processing, and static caching. The platform supports direct HTTP interface calls for data display, online interface encapsulation, prevention of amplification attacks on external interfaces, and third-party system login authentication.

1.4 Unified User Management

The platform uniformly manages administrators and backend users of access systems, implementing single sign-on authentication and distributed authorization. Users access different system management functions through permission settings, with personalized management interfaces based on user permissions. Unified user authentication includes basic information management: account name, nickname, real name, ID card, bound accounts, personal photo, email, etc. Specific user information on different clients and activity platforms is managed by the corresponding platforms.

The platform manages user access permissions to service modules and functions, while permissions within specific modules are managed by the modules themselves. It supports user grouping and tiered management, distinguishing users registered through different clients and channels. Tiered design should comprehensively consider user behavior across various clients and websites.

The platform provides unified authentication management including user registration, multiple authentication levels, and registration information management. Standard user management interfaces enable unified user management for access systems, supporting at minimum mobile phone registration, third-party account registration (Weibo, WeChat, QQ), and ID card authentication. Open authentication interfaces provide user information interaction to third-party systems for comments, interactive activities, and services via SDK, API, URL, etc.

1.5 Visual Configuration

System administrators can perform visual configuration management of access systems through the platform. The system enables embedding and navigation of access modules, allowing backend users to conveniently select modules without secondary login and customize interfaces according to work needs. A complete functional module configuration management interface clearly displays access system modules, app management, user management, and third-party systems for convenient scheduling and configuration.

The platform provides intuitive lists of module access and version status, sup-

porting coexistence of different module versions. Administrators can achieve fast, smooth switching and replacement between modules through selection and configuration.

The Xinhua Press Media Group all-media client support cloud platform project adopted this management platform architecture to support Jiaohuidian news client and website management and publishing. Based on business requirements for government information services and Jiaohuidianhao 入驻, it built an open content 入驻 platform. Standardized interfaces enabled third-party service module access. Unified user authentication and information management facilitated unified service module scheduling. Service module access management supports redirection, takeover, and embedding, achieving true flexibility and control.

References

- [1] Guo Dong, Wang Wei, Zeng Guosun. A Novel Cloudware PaaS Platform Based on Microservices Architecture[J]. Information Network Security, 2015(2): 15-20.
- [2] Building Microservices by Sam Newman, Copyright 2015, Published by O'Reilly Media.
- [3] Microservices: From Design to Deployment by Chris Richardson with Floyd Smith.

(Author's affiliation: Xinhua Press Media Group)

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv — Machine translation. Verify with original.