Approaches to Middle Platform Construction in the Media Industry: Postprint

Authors: Liu Shen, Lin Songtao

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

Abstract

In the era of intelligent media, the construction of the "middle platform" has become a key focus of industry discussion in the media technology field in order to rapidly respond to business requirements. This paper proceeds from the pain points of middle platform construction in the media industry, plans around media business requirements, and aims to make a modest contribution to promoting media technology innovation and development by summarizing our own construction practices and reflections.

Full Text

Preamble

A Brief Discussion on Mid-Platform Construction Strategies for the Media Industry

Authors: Liu Shen, Lin Songtao
Affiliation: TRS Information Technology Co., Ltd., Beijing 100101, China

Abstract: In the era of intelligent media, the construction of "mid-platform" has become a focal point of industry discussion in media technology to rapidly respond to business demands. This paper addresses the pain points of mid-platform construction in the media industry, planning around media business requirements. By summarizing our practical experience and reflections on mid-platform development, we aim to contribute modestly to promoting media technology innovation and development.

Keywords: Mid-platform; Container PaaS Platform; Technology Mid-platform; Data Mid-platform; Industry Mid-platform; AI Mid-platform
Classification Number: TP391
Document Code: A
Citation Format: Liu Shen, Lin Songtao. A Brief Discussion on Mid-Platform

Construction Strategies for the Media Industry [J]. China Media Technology, 2021(02): 14-16+19.

1. What is a Mid-Platform?

In mid-2015, Alibaba' s management visited Supercell, then the world' s most successful mobile gaming company based in Finland. With fewer than 200 employees and less than 10 games, Supercell consistently topped global gaming revenue charts (before being acquired by Tencent the following year). A key factor behind Supercell' s success was its powerful gaming mid-platform that supported development teams averaging only 3-7 people each, enabling parallel development of 50 games and selecting one or two classics for market release through "internal horse racing."

By the end of 2015, Alibaba Group announced the full launch of its 2018 mid-platform strategy, establishing an organizational and business mechanism of "large mid-platform, small front-end" better suited to the DT era—one that is more innovative and flexible. In April 2017, Zhong Hua, Alibaba' s chief middleware architect, published *The Path to Enterprise IT Architecture Transformation: Alibaba' s Mid-Platform Strategic Thinking and Practical Implementation* [1], which detailed the origins, trial-and-error processes, and implementation methodologies of Alibaba' s mid-platform. The book became a classic, selling over 100,000 copies.

By mid-2019, "mid-platform" was included in the Baidu Index, bringing the term into the view of an increasing number of IT professionals [Figure 1: see original paper]. Although mid-platform has since become a hot topic, no unified definition exists, and understandings remain varied. A common online definition is Wang Jian from ThoughtWorks' s "enterprise-level capability reuse platform" [2].

As a B2B ISV serving enterprise-level clients, our understanding is relatively simple and practical. We focus not on capability reuse itself—since reuse has always been an inherent requirement of commercial software development (though not easily achieved in practice)—but rather on the ultimate goal: rapidly delivering software products that satisfy customer needs, creating win-win outcomes. Therefore, we understand mid-platform as facilities that help quickly deliver products that meet requirements. If these facilities are only needed in our R&D and production processes, their specific form matters little; we can let things develop naturally. However, if they are also needed in the customer' s production environment, they must be deliverable as products.

Since delivered software products must operate continuously and stably to satisfy various customer demands, what we deliver is not merely a static product medium but a dynamic, systematic combination including operations and maintenance services. In other words, delivering the product medium is only the beginning. Throughout the service period, requirement iterations, bug fixes, performance optimization, security enhancements, fault recovery, and environ-

ment adjustments—all within the scope of satisfying customer demands—require rapid delivery. Therefore, mid-platform facilities must function not only in the R&D and production process but also in the customer's production environment. Consequently, our mid-platform takes the form of a product.

In Alibaba' s mid-platform strategic practice, the company evolved six mid-platforms based on actual conditions: business mid-platform, data mid-platform, algorithm mid-platform, technology mid-platform, R&D mid-platform, and organizational mid-platform, later adding a mobile mid-platform. Some enterprises have emulated this approach, planning these six or seven mid-platforms without considering their own needs and actual circumstances, making it difficult to achieve ideal results. We believe mid-platform planning must start from one' s own reality and deliver actual value to both ourselves and our customers—not pursuing mid-platform for its own sake. Given the challenges and uncertainties, planning can only establish general directions and overarching principles first, with specific guidelines gradually emerging through continuous exploration and trial-and-error in practice.

Our overarching principle is: create actual value as early as possible. This principle helps guide many decisions. For instance, regarding industry mid-platforms, a necessary prerequisite is that we can only plan R&D for industries whose business we thoroughly understand. Only by mastering the business can we develop industry mid-platforms that rapidly respond to business needs. Otherwise, heavily invested mid-platforms will not only fail to deliver value but also become a hindrance. Technology mid-platforms and data mid-platforms, particularly the former, are generally less affected by industry-specific business. Therefore, our mid-platform R&D begins with the technology mid-platform and data mid-platform.

Based on our R&D philosophy and actual circumstances, we prioritized the development of the technology mid-platform and media industry mid-platform. Theoretically, the media industry mid-platform should utilize the technology mid-platform, data mid-platform, and AI mid-platform. However, TRS' s deep collaboration with media industry customers since the early stages of media convergence has gradually formed reusable platforms for media capabilities such as central kitchens and media big data. Following the principle of creating actual value early, these can be directly 下沉 (sink down) to form the media industry mid-platform without waiting for the other mid-platforms to be rebuilt.

3.1 Container PaaS Platform

The core is a Docker + K8S resource scheduling platform that implements containerized middleware resource pools, dynamic resource allocation and scheduling, and middleware clusters. Container/cloud-native has long become a trend. Cloud-native refers not only to where applications reside but more importantly to how applications are built and deployed. Cloud-native technologies enable organizations to build and run scalable applications in dynamic environments such

as public, private, and hybrid clouds. These technologies facilitate the construction of loosely coupled systems with good fault tolerance, easy management, and convenient observability. Combined with reliable automation, cloud-native technologies allow engineers to make frequent and predictable major changes to systems easily. The Cloud Native Computing Foundation (CNCF) defines cloud computing as containerized packaging + automated management + microservices orientation. In 2018, CNCF updated the cloud-native definition to include service mesh and declarative APIs [3]. Specifically, container technology, service mesh, microservices, and serverless are core cloud-native technologies, while DevOps is a key practice for integrating these technologies.

Key advantages of cloud-native include: (1) accelerating product delivery and iteration through agile and DevOps processes compared to traditional monolithic applications; (2) automatically and incrementally improving cloud-native applications under microservices architecture to continuously add new features or enhance existing ones; (3) enabling non-invasive improvements without causing downtime or service interruptions that create poor user experiences; (4) supporting easy elastic scaling for cloud-native applications under different access load pressures; and (5) enabling cloud-native development processes to better adapt to the speed and innovation required by today' s business environment.

3.2 One-Stop DevOps Platform + API Gateway

The core of DevOps is achieving continuous integration and continuous delivery capabilities while implementing critical testing and operations capabilities such as test management and automation, environment management, configuration management, monitoring and alerting, and elastic scaling of application node scale. DevOps is essentially a philosophy of "development-operations integration" that breaks down barriers between Development and Operations teams, emphasizing communication and collaboration between developers and operations personnel. Based on CI/CD (Continuous Integration/Continuous Deployment), it optimizes all aspects of development, testing, and system operations. Simply put, the focus shifts from just development to also encompass deployment, upgrades, monitoring, and operations. Compared to agile development, which has been popular for over a decade, DevOps is essentially an advanced version that adds the operations phase, thereby covering the entire product development and usage process.

In concrete implementation, DevOps emphasizes CI/CD because a long-standing industry problem is that IT changes cannot keep pace with business changes—IT changes simply take too long, requiring development, testing, deployment, and other stages while comprehensively considering the impact of each change. For example, when a company launches feature H, it must ensure that existing features ABCDEFG remain unaffected. Only compliant continuous integration/continuous deployment can ensure IT keeps up with market changes. DevOps also includes monitoring and alerting, a key focus for improving operations capabilities. The core is achieving monitoring of

resources such as containers, middleware, and applications, service and service chain monitoring, and problem detection with automated operations processes. Issues discovered through monitoring enter the operations workflow, and user feedback can also enter this workflow.

Regarding API service gateways: a mid-platform is an independent organization responsible for serving multiple front-end businesses, thus requiring standard service interfaces, mature service governance capabilities, and efficient agile R&D technologies. In the current technical environment, adopting universally familiar REST-style synchronous APIs and message queue asynchronous communication as standard service interface technologies, and using API gateways as standard service governance is the most appropriate choice. The gateway subsystem integrates key capabilities such as service registration centers, service chain monitoring, and service rate limiting and circuit breaking. API gateways can also cooperate with DevOps for more flexible grayscale publishing and elegant degradation functions.

3.3 Basic Applications

Basic applications are first industry-independent and must provide simple, easy-to-use APIs or even lighter and faster access methods, enabling other applications (primarily business applications) to gain relevant capabilities with minimal or even no coding. For example:

Audio/video processing: Many business systems have audio/video file processing needs, such as CMS needing to upload and publish videos or knowledge management systems needing to disseminate video-based knowledge. With diverse audio/video file formats, if each business system developed its own solution, it would involve significant redundant development and long cycles. Moreover, audio/video files scattered across different business systems create numerous problems for storage management, network transmission, file circulation, and user interaction. The audio/video processing application thus serves as a mid-platform, providing component-based upload and playback pages for external embedding by other systems. By following integration specifications and developing minimal pages, other systems can add audio/video functionality, allowing users to seamlessly access these features within existing business processes through browser interactions. This greatly reduces development workload while enabling centralized management of audio/video files and optimizing storage and other aspects. The overall effect is that audio/video upload, transcoding, playback, and other processing are completely encapsulated within the audio/video processing application system, requiring other systems to understand few details and thus minimizing development work.

Unified authentication: Similarly, each business system only needs to add a class according to integration specifications without modifying existing code to obtain unified user provisioning from the unified authentication system and gain single sign-on capability.

4. Media Industry Mid-Platform Construction

In the mobile internet era, user needs are paramount, and platform-based approaches can multiply our effectiveness. Given that traditional "front-end + back-end" platform architectures have gradually failed to meet users' rapidly changing demands, our media products adopt the "mid-platform" concept—a platform truly born for the front-end—to better serve front-end scaled innovation and respond to service needs, achieving rapid and continuous alignment between our capabilities and user requirements.

By constructing business mid-platform, technology mid-platform, data mid-platform, and AI mid-platform, we build a cross-business-line media industry mid-platform. This creates a large convergence and large control platform capability layer to realize a capability reuse platform. Our media industry mid-platform facilitates easy acquisition of capabilities by various business lines through service reuse, 沉淀 (precipitating) capabilities and data to foster innovation and resolve the contradiction between "rapid business changes" and "slow technology system iteration."

Under our media industry mid-platform system, container technology provides stable runtime environments and agile development foundations for PaaS mid-platform service capabilities, technical components, and middleware. Microservices architecture enables faster implementation of PaaS mid-platform service capabilities for upper-layer media applications. DevOps accelerates the iteration speed of PaaS mid-platform service implementation. Through the data mid-platform, media-owned data assets and external data resources are centrally managed. Relying on big data and AI technologies, data is tagged and archived, establishing unified data service standards and interfaces for upper-layer media services to provide standardized data services for different media application scenarios. AI services provide media intelligent engine services, covering intelligent speech, intelligent vision, and natural language processing in both content perception and cognition directions. The AI mid-platform, based on machine learning and deep learning and combined with our years of accumulation in the media industry, independently customizes service corpora and learning samples to make AI implementation in media scenarios more concrete.

Our media industry mid-platform can be disassembled according to user project scale and actual requirements. For overall projects adopting the large mid-platform concept, we create unified mid-platform capability services. Alternatively, we can independently deploy technology mid-platform, data mid-platform, or AI mid-platform services to achieve mid-platform capability layer construction oriented toward different requirement directions, better matching user realities and completing the upgrade and transformation from technology platform to mid-platform services.

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