Journal of Electronic Research and Application, 2025, Volume 9, Issue 1
http://ojs.bbwpublisher.com/index.php/JERA
ISSN Online: 2208-3510
ISSN Print: 2208-3502

Design of Universal Platform Architecture for Complex Discrete Storage System

Yangyang Zhang, Haiting Xu, Dong Wang*, Bo Xu

Tianjin Institute of Aerospace Mechanical and Electrical Equipment, Tianjin 300000, China

***Corresponding author:** Dong Wang, zyy199200@126.com

Abstract: The flexible satellite batch production line is a complex discrete production system with multiple cross-disciplinary fields and mixed serial parallel tasks. As the source of the satellite batch production line process, the warehousing system has urgent needs such as uncertain production scale and rapid iteration and optimization of business processes. Therefore, the requirements and architecture of complex discrete warehousing systems such as flexible satellite batch production lines are studied. The physical system of intelligent equipment is abstracted as a digital model to form the underlying module, and a digital fusion framework of "business domain + middleware platform + intelligent equipment information model" is constructed. The granularity of microservice splitting is calculated based on the dynamic correlation relationship between user access instances and database table structures. The general warehousing functions of the platform are divided to achieve module customization, addition, and configuration. An open discrete warehousing system based on microservices is designed. Software architecture and design develop complex discrete warehousing systems based on the SpringCloud framework. This architecture achieves the decoupling of business logic and physical hardware, enhances the maintainability and scalability of the system, and greatly improves the system's adaptability to different complex discrete warehousing business scenarios.

Keywords: Satellite batch production line; Complex discrete production; Warehousing system; Architecture design; Flexibility; Microservices

Online publication: February 14, 2025

1. Introduction

In recent years, with the intensification of competition among aerospace powers, satellite mass production lines have become a key focus of research on high-speed satellite networking and production. The flexible satellite batch production line is a complex discrete production system that integrates multiple professional technical fields and involves a mixture of serial and parallel tasks. As the source of the production process, the warehousing system has urgent needs such as uncertain production scale, rapid iteration and optimization of business processes, high degree of customization, and short development cycle. It is urgent to design and package flexible warehousing systems with business needs as the core, to achieve on-demand use and on-demand adaptation [1,2].

In response to the shortcomings of existing methods, this article studies the requirements and architecture of complex discrete warehousing systems such as flexible satellite batch production lines. Based on the idea of digital systems, the physical system of intelligent equipment is abstracted as a digital model to form the underlying module. The platform's warehousing general functions are segmented to achieve module customization, addition, and configuration. An open software architecture for an intelligent warehousing system based on microservices is designed. This architecture fully utilizes the flexibility and scalability of microservices, enabling the system to automatically configure according to changes in business requirements and enhance the generalization capability of the warehousing platform [3,4].

2. Architecture design

According to the organization and deployment structure of the system, the evolution process of software architecture can be roughly divided into the following stages: monolithic architecture, Service Oriented Architecture (SOA), and microservice architecture [6], as shown in **Table 1** for comparison. The monolithic architecture integrates all business logic and control logic into one program, which is suitable for all simple applications. As business becomes increasingly complex, a module failure can cause the entire system to crash. The vertical splitting of monolithic architecture has evolved into Service Oriented Architecture (SOA). The microservice architecture emphasizes componentization and servitization, with services that can interact and integrate with each other.

The flexible satellite batch production line is a highly integrated complex discrete production system, specialized in producing high-quality and high-precision complex products such as satellites. As the source of the satellite batch production line process, the warehousing system has challenges such as multiple varieties, small batches, and flexible production processes. It is required that the warehouse system architecture has good scalability and reconfigurability to flexibly respond to various demand changes. The modular, independent deployment and easy scalability advantages of microservice architecture can meet the requirements and become the best choice for flexible warehousing.

2.1. Standard information model

To cope with the continuous evolution of business processes and build a flexible warehousing system, there is an urgent need for an open and highly scalable business domain model. The objects that the current business model can describe are fixed and difficult to flexibly expand [5]. Therefore, a solution of "business domain + middleware platform + intelligent equipment information model" is proposed, as shown in **Figure 1**. At the same time, it can flexibly adjust and optimize the deployment of microservice architecture according to the actual scale and business complexity of the production line. The middleware platform is a bridge for information exchange between business domains and intelligent equipment. It digitizes the information model of intelligent equipment and interacts with business data to ensure unified and accurate data standards, providing support for subsequent traceability analysis and visual display, building a reusable and standardized intelligent equipment information model, solving the problems of diverse, non-open or non-standard communication protocols, and achieve interconnection and intercommunication of on-site devices [5,6].

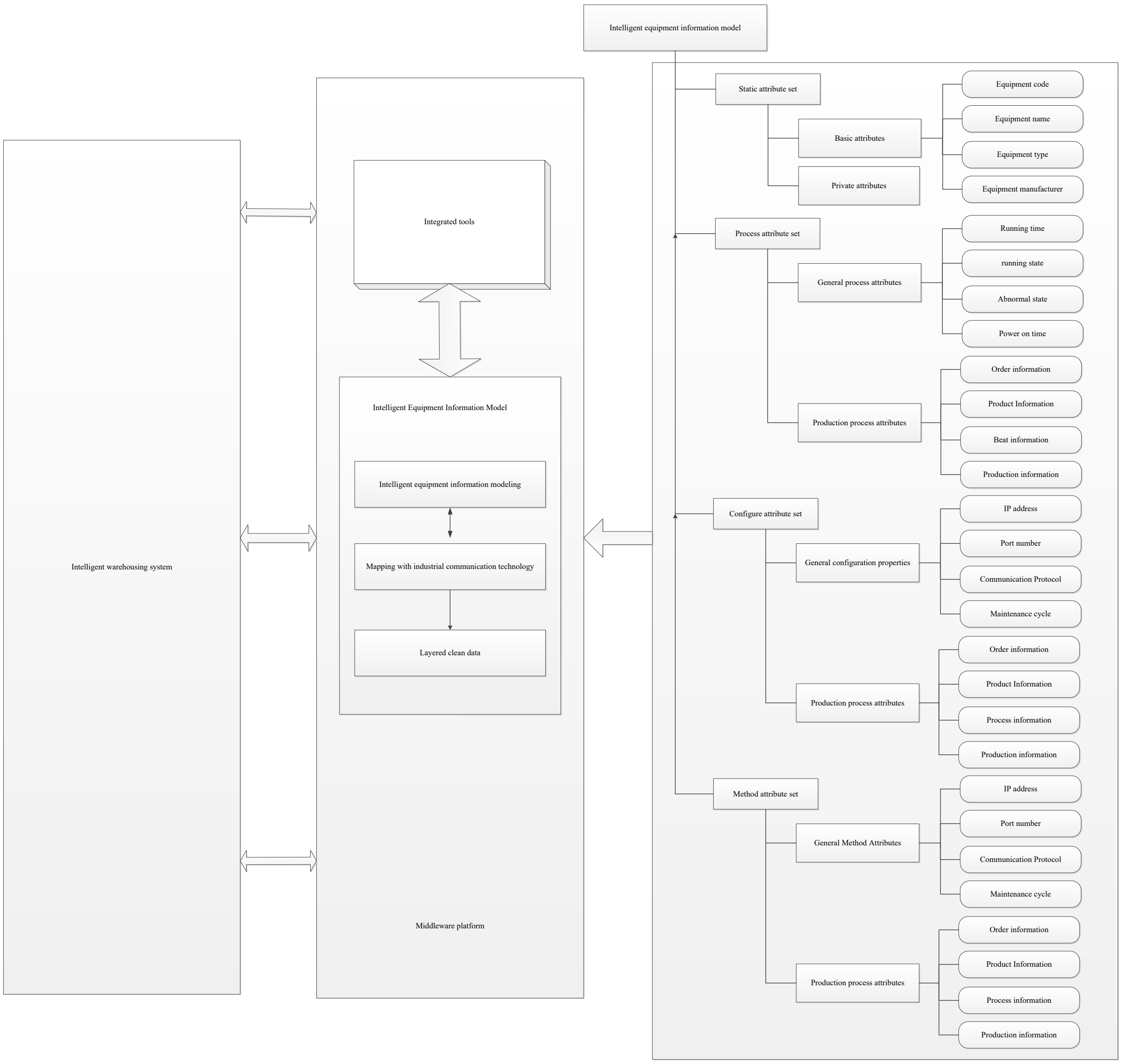

Figure 1. System standard model

2.2. Functional architecture design

The general platform for complex discrete warehousing systems realizes the management of inbound and outbound inventory for different warehouses, and can also adapt to changes in complex discrete scenarios. The functional architecture is shown in the following **Figure 2**. This platform has functions such as warehouse management, inventory management, report management, kanban management, basic information management, permission management, system configuration, and many more [7].

It includes the equipment layer, data layer, business layer, interface layer, and upper layer applications. The equipment layer is composed of multiple intelligent devices, each of which is independent of the other. The device control layer mainly controls the actions of each device through protocols. The business layer consists of a middleware platform, a Warehouse Management System (WMS), and a monitoring display module. The middleware platform mainly completes the reception, parsing, decomposition, and process control of upstream system instructions, and distributes the decomposed instructions to various intelligent devices in sequence according to the process. At the same time, it completes functions such as different interfaces, data standardization, and data collection, monitoring, and analysis for each intelligent device. The system configuration function of WMS includes a rule engine and process configuration. The rule engine includes coding rules, shelving rules, and picking rules, and supports custom configuration and extension. Process configuration can be used to configure processes such as receiving, acceptance, and shelving. The interface layer is responsible for integrated communication with upper-layer applications.

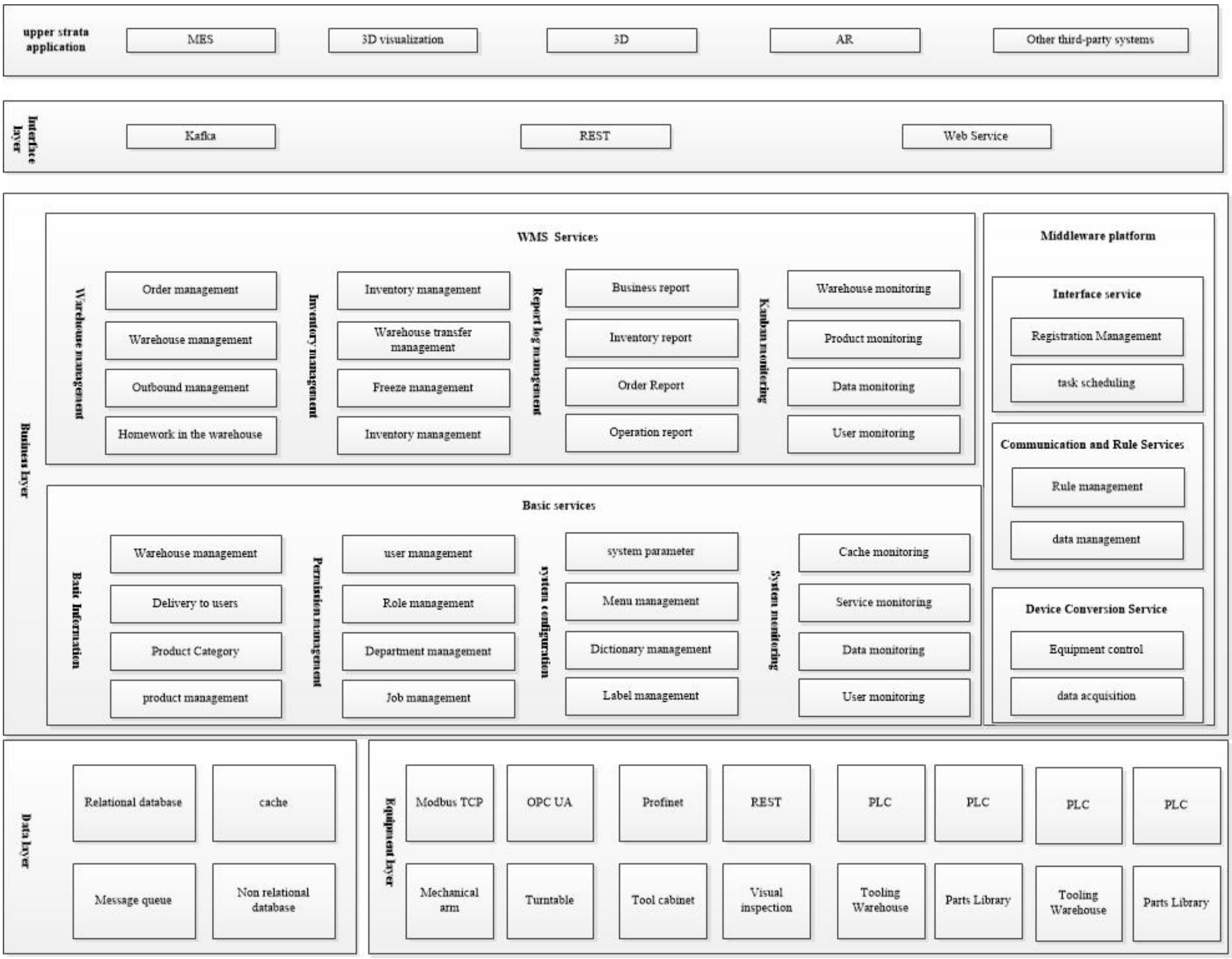

Figure 2. Function architecture

2.3. Microservice segmentation

According to the "business domain + middleware platform + intelligent equipment information model," the middleware platform organizes services into four categories based on system scalability: interface services, device protocol conversion services, communication services, and rule services. Based on system reusability, services are divided into basic services and query services. Furthermore, based on business functions, services are categorized as inbound services, outbound services, and inventory services, as shown in **Figure 3**.

The interface service provides Representational State Transfer Application Programming Interface (REST API) and Kafka queue communication methods to communicate with upper-level application systems, and call device protocol conversion modules and business management modules according to the specific types of tasks [8]. The basic service is the fundamental data part of the middleware platform, and all operational data of the business is based on the current module. Business data processing is carried out based on the data in this module. The device protocol conversion service includes a protocol adaptation and conversion engine, as well as a data unification engine. The protocol adaptation and conversion engine is based on the communication connection of multi-source heterogeneous devices. Simultaneously, the protocol can be converted into more universal Application Programming Interface (API) or Software Development Kit (SDK) interfaces, enabling compatibility with the scheduling and control of multi-source heterogeneous devices. The data unification engine processes JSON (JavaScript Object Notation) data parsed from the intelligent equipment information model and standardizes it into a unified format before pushing it to the message queue. Communication and rule services are the core components of middleware platforms. These services handle subscribing to incoming data from message queues and performing various processing and analysis tasks, including communication management, rule management, data analysis, alarm management, device predictive maintenance, and other critical functions.

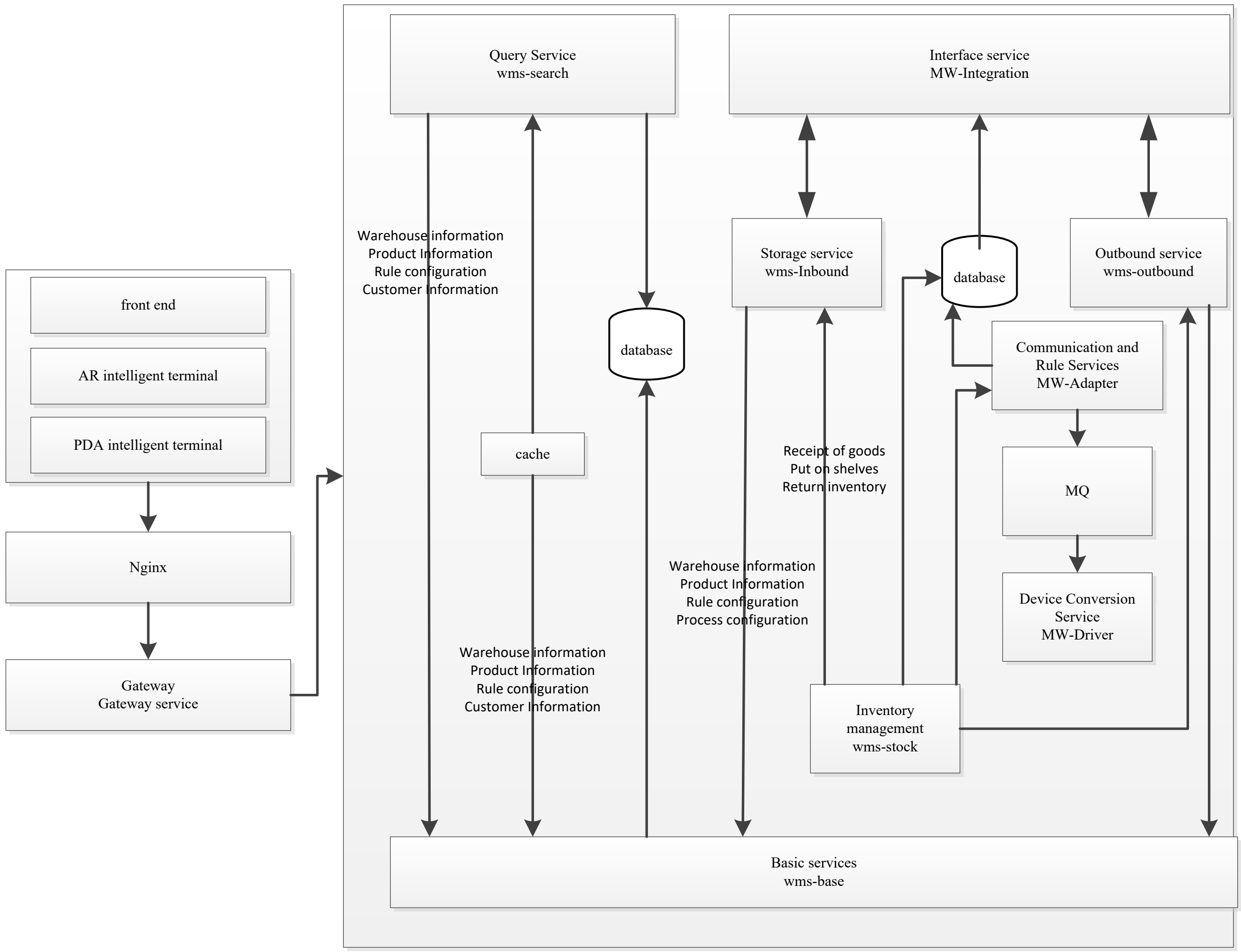

Figure 3. Microservice segmentation

3. System software design

3.1. Warehouse management

As a key link in the warehousing system, inventory management covers the entire process from generating inbound orders to updating inventory, ensuring efficient product warehousing, and real-time monitoring of inventory status. **Figure 4** shows the operational process of the system's inbound business.

(1) Inbound order management includes generating inbound orders, printing inbound orders, and storing inbound orders. The incoming order is imported into the system by the process personnel, and the incoming order is automatically generated, with detailed records of the incoming order number.

(2) Inspection management conducts receiving and preliminary product inspections based on the inspection tasks generated from the incoming orders to prevent the storage of non-conforming products.

(3) Shelf management will safely and efficiently place products that have passed inspection in designated storage locations, and update inventory information in real-time. The system intelligently recommends or manually allocates shelf locations, and the recommended shelf locations are allocated according to the preset shelf strategy.

Figure 4. Inventory process

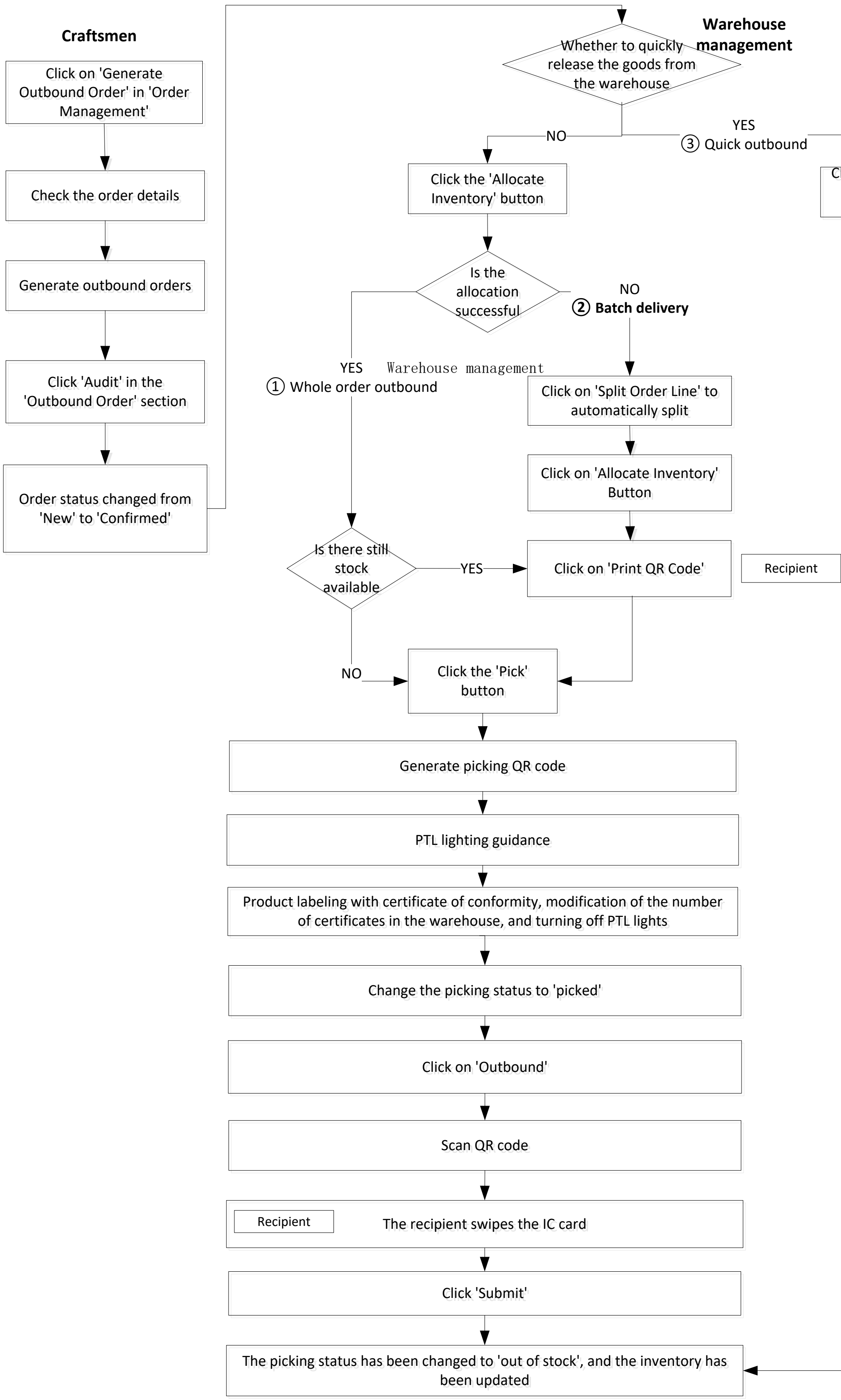

Figure 5. Outbound process

3.2. Outbound management

Outbound management mainly includes outbound order management, inventory turnover rule setting, picking management, and outbound record query, ensuring accurate and efficient product outbound process. When the product is released from the warehouse, a release order is generated in the system, inventory turnover rules are set, release is allocated, release picking tasks are generated, and automatic release is executed. After the release is completed, a release record is automatically generated to update the product inventory. **Figure 5** shows the operation process of outbound business.

(1) Outbound order management, where process personnel generate outbound orders in the system based on the outbound plan. Outbound orders include key information such as outbound order number, task number, batch number, part drawing number, part name, outbound quantity, and whether mixed batch outbound is required. After the outbound order is generated, it needs to go through an audit process to ensure the accuracy and legality of the order information. Before executing the outbound operation, the outbound order can be modified or canceled to meet the corresponding requirements.

(2) Picking management allows the system to remove and locate products that need to be shipped based on outbound orders and preset inventory turnover rules. The warehouse management personnel carry out picking operations according to the designated storage location and quantity indicated by the system, ensuring accurate and error-free picking and shipping of products.

4. Conclusion

The general platform for complex discrete warehousing systems based on microservices has been successfully applied to multiple complex discrete scenario projects such as aerospace. The system adopts a microservice architecture, with each service deployed and upgraded independently, achieving a high degree of modularity and loose coupling, significantly improving the maintainability and flexibility of the system. The digital integration framework of "business domain + middleware platform + intelligent equipment information model" has seamlessly integrated business logic and underlying devices, greatly improving job accuracy and efficiency. The system has functions such as equipment management, inbound and outbound management, and inventory management, which can quickly adapt to different complex discrete warehousing business scenarios, promote the automation and intelligence of business processes, and ensure the precision and efficiency of warehousing operations.

Disclosure statement

The authors declare no conflict of interest.

References

[1] Wang J, Du Y, 2019, Design and Implementation of Intelligent Warehouse Management System. Computer Technology and Development, 29(12): 189–193.

[2] Zhen L, Haolin LI, 2022, A Literature Review of Smart Warehouse Operations Management. Frontiers of Engineering Management, 9(1): 25.

[3] Zhao Q, Liu M, Zou Y, et al., 2023, Research and Application of Special Loading Warehouse Storage Management

System. Aerospace Industry Management, 2023(09): 64–68.

[4] Revillot-Narvaez D, Perez-Galarce F, Alvarez-Miranda E, 2020, Optimizing the Storage Assignment and Order-Picking for the Compact Drive-In Storage System. International Journal of Production Research, 58(22): 6949–6969.

[5] Sharma R, 2021, Intelligent Warehousing Based on the Internet of Things Technology. Computing Reviews, 2021(4): 162–163.

[6] Ye S, 2023, Research on Dynamic Adjustment and Quality Optimization of IoT Perceived Service Composition, thesis, North China University of Technology.

[7] Leng JW, Yan DX, Liu Q, et al., 2019, Digital Twin-Driven Joint Optimization of Packing and Storage Assignment in Large-Scale Automated High-Rise Warehouse Product-Service System. International Journal of Computer Integrated Manufacturing, 09(29): 1–18.

[8] Leung E, Lee C, Ouyang Z, 2022, From Traditional Warehouses to Physical Internet Hubs: A Digital Twin-Based Inbound Synchronization Framework for Pi-order Management. International Journal of Production Economics, 2022: 244–259.