

学号：

20330174027



中原科技学院

ZHONGYUAN INSTITUTE OF SCIENCE AND TECHNOLOGY

本科毕业论文

题目：混合云下的分布式 MOM 系统——

区块链、大数据和人工智能的综合应用

学 院：机电工程学院

专 业：机械设计制造及其自动化

年级班级：2020 级（二）班

姓 名：马庆奥

指导教师：边军

2024 年 05 月

中原科技学院毕业论文（设计）原创性声明

本人郑重声明：所呈交的毕业论文（设计），是本人在指导教师的指导下进行研究工作所取得的成果。除论文中已经注明引用的内容外，本论文没有抄袭、剽窃他人已经发表的研究成果。本声明的法律结果由本人承担。

毕业论文（设计）作者签名：_____ 日期：_____

关于毕业论文(设计)使用授权的说明

本人完全了解中原科技学院有关保留、使用毕业论文(设计)的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权中原科技学院可以将本论文（设计）的全部或部分内容编入有关数据库进行检索，可以上报全国本科毕业论文（设计）抽检信息平台进行抽检，可以采用影印、缩印或扫描等复制手段保存和汇编本论文(设计)。

本学位论文属于

保 密□，在_____年解密后适用本授权书。

不保密□。

毕业论文（设计）作者签名：_____ 日期：_____

指导教师签名：_____ 日期：_____

混合云下的分布式 MOM 系统——区块链、大数据和人工智能的综合应用

摘 要 本文设计的混合云下的分布式制造运营管理 (Manufacturing Operations Management) 系统是一种融合了区块链、大数据与人工智能技术的创新型解决方案。伴随着数字化转型步伐的加快及企业生产运营复杂性的上升, 该系统为企业提供了一条高效、可靠以及安全的途径, 用于管理其生产和运营流程。

基于 The Open Group Architecture Framework (TOGAF) 理念架构的系统体系结构设计、区块链技术下的生产数据溯源、大数据平台下的数据中心、Retrieval-Augmented Generation (RAG) 技术下的知识库建设等方法, 对企业的信息化改造探索了方向。

本文对 RAG 系统的召回精确度问题进行了一定程度上的研究, 提出了一种新的存储检索结构模型。本文提出了一种名为 Chunk-Tree 的设计, 这种检索召回结构极大地提高了增强检索召回精确度。通过分析和实验表明, Chunk-Tree 结构可以提高系统的召回率, 并能有效地降低 RAG 系统的错误率。

综上所述, 混合云下的综合运用区块链技术、大数据技术、人工智能技术的分布式制造运营管理系统, 为企业的制造运营管理提供了一套较为完善的解决方案。该系统将使企业的制造运营管理更加数字化、智能化, 对于相关企业的生产实践具有一定的指导价值。

关键字 制造运营管理、混合云、RAG

Distributed MOM System under Hybrid Cloud - Integrated Application of Blockchain, Big Data, and Artificial Intelligence

Abstract The Distributed Manufacturing Operations Management system designed in this article under a hybrid cloud is an innovative solution that integrates blockchain, big data, and artificial intelligence technologies. With the acceleration of digital transformation and the increasing complexity of enterprise production and operation, this system provides enterprises with an efficient, reliable, and safe way to manage their production and operation processes.

The system architecture design based on the Open Group Architecture Framework (TOGAF) concept architecture, production data traceability under blockchain technology, data centers under big data platforms, and knowledge base construction under Retrieval Augmented Generation (RAG) technology have explored the direction of enterprise information transformation.

This article conducts a certain degree of research on the recall accuracy of RAG systems and proposes a new storage retrieval structure model. This article proposes a design called Chunk Tree, which greatly enhances the accuracy of retrieval recall. Through analysis and experiments, it has been shown that the Chunk Tree structure can improve the recall rate of the system and effectively reduce the error rate of the RAG system.

In summary, the comprehensive application of blockchain technology, big data technology, and artificial intelligence technology in the distributed manufacturing operation management system under hybrid cloud provides a relatively complete solution for the manufacturing operation management of enterprises. This system will make the manufacturing operation management of enterprises more digital and intelligent, and has certain guiding value for the production practice of related enterprises.

Keywords manufacturing operations management, hybrid cloud, RAG

目 录

1 绪论	1
1.1 研究背景	1
1.2 研究意义	1
1.3 国内外现状	2
1.4 论文主要研究内容	3
2 MOM 系统相关技术研究	4
2.1 区块链技术分析	4
2.2 混合云技术分析	4
2.3 业务开发技术分析	5
2.4 本章小结	6
3 MOM 系统需求分析	7
3.1 业务需求分析	7
3.2 功能需求分析	8
3.3 非功能性需求分析	9
3.4 小结	11
4 系统设计	12
4.1 设计原则	12
4.2 架构设计	13
4.2.1 业务架构	13
4.2.2 数据架构	14
4.2.3 应用架构	15
4.2.4 技术架构	16
4.3 Mom 系统	18
4.3.1 业务架构设计	18
4.3.2 各模块数据库表设计	20
4.4 自动化运维系统设计	23
4.4.1 自动扫描发现资产模块	23
4.4.2 任务自动执行模块	24

4.5 企业知识库	25
4.5.1 架构设计.....	25
4.5.2 Embedding 算法分析	26
4.6 系统集成	27
4.7 本章小结	28
5 RAG 系统召回准确性问题研究	29
5.1 RAG 系统及研究问题简述	29
5.2 RAG 系统模型构建	31
5.3 Chunk-Tree 查找算法分析	32
5.3.1 B+树结构分析.....	32
5.3.2 Chunk-Tree 查找算法设计	32
5.4 本章小结	34
6 系统实现与测试	35
6.1 开发工具介绍.....	35
6.1.1 后端开发技术.....	35
6.1.2 前端开发技术.....	36
6.1.3 大数据开发技术.....	36
6.1.4 系统运维技术.....	38
6.1.5 RAGSys 技术	39
6.2 测试方法简述.....	40
6.3 系统整体功能实现与测试.....	41
6.3.1 MOM 系统	41
6.3.2 自动运维平台.....	46
6.3.3 大数据平台.....	48
6.3.4 知识库系统.....	49
6.4 系统非功能测试.....	50
6.5 本章小结	51
7 总结与展望	52
7.1 全文总结	52
7.2 展望	52

参考文献	53
附录	55
致谢	56

1 绪论

1.1 研究背景

随着世界经济的快速发展和科技的持续创新，跨国企业在全球化贸易浪潮中面临着分布全球的更加细颗粒度的制造运营管理困境。传统的制造运营管理方式存在着信息不对称、数据孤岛和效率低下的等诸多问题。这些问题导致跨国企业的生产成本上升，工厂生产的产品质量难以控制，工厂库存供应链存在难以管理的风险。为了应对这些问题，许多企业开始进行企业信息化建设和企业数字化转型，探索更加智能化的制造运营管理解决方案^[1]。

区块链、大数据和人工智是当下热门的前沿技术，并且在过往的十年里面，在各个行业展现出巨大的潜力和应用前景。区块链技术具有去中心化、不可篡改和可追溯的特点，可以解决企业生产经营过程中信息不对称和数据安全的问题。大数据技术可以促进企业从生产运营过程中产生的数据中发现规律和趋势，并且可以通过机器学习训练模型，提供决策支持和优化建议。人工智能技术可以通过机器学习和神经网络训练，提高企业生产运营管理的效率和智能化水平^[2]。

根据高德纳 (Gartner) 公司的预测报告，在 2022 年的报告中就包含有大量的关于生成式人工智能的预测，还有分布式企业、云原生平台等领域，在 2023 年和 2024 年的报告中，均有提及关于行业云平台的论述，以及诸多条关于人工智能领域的内容^[3]。本研究旨在探索一种将以上技术结合应用在企业生产中的解决方案，并为本领域提供新的见解，对于填补当前研究领域的知识空白、推动实践进步以及促进相关政策的制定具有一定的意义。

1.2 研究意义

研究综合应用区块链、大数据和人工智能技术的混合云的分布式制造运营管理系统中的综合应用，对企业提高产品质量、精细化管理生产活动、优化供应链具有重要意义。

提高生产运营的透明度和质量控制：通过采用区块链技术并将其与业务需求相结合，可以实时监控生产过程，并确保生产链的可追溯性。区块链技术的全面应用可以提高企业生产活动的透明度和可追溯性，大大降低质量问题和欺诈风险。

实现供应链的优化和风险管理：利用大数据技术，系统可以收集和分析海量的生产数据，帮助企业准确预测市场需求、优化供应链，并降低库存成本和交付风险^[4]。

提高生产效率，降低生产成本：降低成本，提高效率是当今企业热衷的管理理念。通

过使用人工智能技术，可以自动处理大量的数据和任务，达到降低劳动力成本和提高生产效率的目的。通过使用机器学习技术，企业还可以识别生产过程中的潜在瓶颈，优化解决方案，进一步降低生产成本。

推动企业数字化转型：混合云下的分布式制造运营管理系统为企业实现数字化转型提供了全面的解决方案，增强了企业在不断变化的全球市场中的核心竞争力。

基于混合云集成应用区块链、大数据和人工智能技术的分布式制造运营管理系统帮助制造商实现全球生产运营转型，为企业带来更加统一、灵活、可扩展、安全、智能的全球制造运营管理能力，促进企业向智能化转型，提高企业核心竞争力和创新能力。本项目对 MOM 系统的探索和研究，对促进经济可持续发展、推动社会进步具有一定的探索价值。

1.3 国内外现状

混合云下的综合应用区块链、大数据和人工智能技术的分布式制造运营管理系统是跨国企业数字化转型和智能化生产运营管理的重点领域。

在国外，许多企业已经开始采用制造运营管理系统。一些传统制造业巨头如通用电气、达索和西门子已经采用了这样的系统来管理其在全球供应链和工厂生产活动。制造运营管理系统利用区块链技术确保数据的安全性和可溯源，使用大数据技术进行生产运营过程中的数据收集和分析，利用人工智能领域技术进行产品缺陷检测和生产过程优化^[5]。

一些国外的新成立的创新型企业也在基于混合云的分布式制造运营管理系统领域取得了不错的成绩。例如，美国的电力汽车公司特斯拉使用区块链技术构建了一个去中心化的制造运营管理系统，实现了生产和销售过程的全链路可溯源，提高了生产效率，降低了质量控制成本^[6]。

在国内，制造运营管理系统也逐渐得到重视和推广。一些大型制造企业开始尝试利用区块链、大数据和人工智能技术来优化其制造运营管理。例如，沈阳一汽汽车制造公司利用区块链技术实现了生产链路的溯源管理，提高了汽车生产的质量和安全性^[7]。国内的一些互联网巨头也在这一领域进行了积极探索，例如阿里云、百度云、腾讯云、华为云。它们均有制造业信息化团队，通过利用大数据技术进行分析，优化供应链管理和生产排产计划，提高了企业的交付效率和客户满意度^[8]。

国内的一些传统企业也在制造运营管理系统方面取得了一些进展。三一重工、徐工集团和国家电网在企业数字化升级的过程中，均采购了大量的传感器和供应链软件，用于实

时监测和分析生产过程中的数据,这种方案为企业提供智能化的制造运营管理解决方案^[9]。

尽管在国内外都有诸多制造运营管理系统的成功案例。但是,根据高德纳的行业预测,基于混合云下的分布式制造运营管理系统仍处于起步阶段,未来仍属于行业云平台的蓝海。在实际生产经营活动中,企业仍然存在诸多的挑战,例如企业数据安全问题、系统应用数据集成问题和企业人才培养问题。

1.4 论文主要研究内容

本论文旨在探讨混合云下的分布式制造运营管理系统中区块链、大数据和人工智能的综合应用,并研究其对企业制造运营管理的影响和价值。具体而言,论文将重点研究以下几个方面:

混合云下的分布式制造运营管理系统架构:本文将结合实际生产需求,进行需求分析,并结合 TOGAF 理论进行系统架构设计,包括业务架构、数据架构、应用架构和技术架构,并在系统开发过程中使用区块链技术 and 大数据技术完善业务系统以达到企业对供应链溯源和产品质量控制的需求。

混合云下的分布式制造运营管理系统自动化运维管理:论文将分析混合云下的分布式架构,包括公有云和私有云的结合与混合云架构下的自动化运维管理,构建适用于制造运营管理的混合云架构和自动化运维体系。

人工智能技术在制造运营管理中的应用:论文将研究人工智能技术在制造运营管理中的应用,包括机器学习和自然语言处理等方面。通过构建 RAG 系统,研究人工智能技术结合企业内部培训系统对生产效率提升、质量控制、和风险管理等方面的价值和意义。

系统集成与测试分析:最后,论文将研究混合云下的分布式运营管理系统的集成与测试。通过测试,评估该系统效果和价值。同时,探讨实施该系统所面临的挑战和解决方案,为企业实施该系统提供一定程度的建议。

通过对基于混合云下的综合运用区块链、大数据和人工智能技术的分布式制造运营管理系统进行深入研究,本论文将为企业提供一种智能化的制造运营管理解决方案。论文的研究成果将为相关领域的学术研究和生产实践提供一定的参考和借鉴。

2 MOM 系统相关技术研究

2.1 区块链技术分析

区块链技术作为一种分布式、去中心化的数据存储和交互方式，正在混合云下的分布式制造运营管理系统中得到广泛应用。本论文将对区块链技术在该领域的应用进行分析。

区块链技术可以提供数据的安全性和可信度。传统的制造运营管理系统中，数据可能存在篡改和伪造的风险，导致信息的不准确和错误的决策。而区块链技术通过使用密码学算法和共识机制，确保数据的不可篡改性和可追溯性。每个区块都包含了前一个区块的哈希值，一旦有数据被篡改，整个链都会失效，从而保证了数据的安全性^[10]。

区块链技术可以实现供应链的溯源和透明化。在制造运营管理中，供应链的透明度和可追溯性对于质量控制和合规性非常重要。区块链技术可以记录和跟踪物料和产品的来源、生产流程和交易记录，确保供应链的可追溯性和真实性。通过区块链技术，企业可以实现对供应链各个环节的实时监控和溯源，提高质量控制和反欺诈能力^[11]。

区块链技术还可以提高合同管理和交易的效率。在制造运营管理中，涉及大量的合同和交易，传统的合同管理和交易流程通常需要大量的时间和资源。区块链技术可以通过智能合约将合约和交易信息编码为可执行规则，实现自动化的合约执行和交易验证。这可以简化流程，降低成本，并提高合同和交易的效率和可靠性。

目前区块链市场化的应用有各大联盟链，通过接入联盟链，可以实现多家公司之间的数据共享和数据信任，保证了对供应链全链路的深度监控^[12]。不仅可以提高企业管理能力，还有助于保障消费者权益。

综上所述，区块链技术在混合云下的分布式制造运营管理系统中具有重要的应用价值。它可以提供数据的安全性和可信度，实现供应链的可追溯性和透明度，并提高合同管理和交易的效率。然而，区块链技术在实际应用中仍面临一些挑战，如性能可扩展性、隐私保护和标准化。未来的研究和实践应着眼于这些挑战并寻求相应的解决方案，以促进区块链技术在制造业运营管理领域的进一步发展和应用。

2.2 混合云技术分析

混合云技术在混合云环境下的分布式制造运营管理系统中发挥着至关重要的作用。

混合云技术可以提供灵活性和可扩展性。私有云和公有云各有其独特的优势。私有云

可以提供更高的安全性和控制，而公共云提供更大的计算和存储资源。通过将私有云和公共云相结合，企业可以根据实际需求灵活调整资源使用，满足数据安全和隐私要求，并扩展计算和存储资源。

混合云技术可以实现数据的分布式存储和处理。在分布式制造运营管理系统中，涉及到大量的数据产生和处理，而私有云和公有云可以提供分布式的数据存储和处理能力^[13]。通过将数据分布在私有云和公有云中，可以降低单点故障的风险，并提高系统的可靠性和性能。

混合云技术还可以实现弹性和成本优化。私有云通常需要企业自己购买和维护硬件设备，而公有云则以按需付费的方式提供计算和存储资源^[14]。企业可以根据实际需求灵活调整资源的使用量，将部分服务程序迁移到公有云进行部署，避免资源浪费，降低成本。

综上所述，混合云技术在分布式制造运营管理系统中呈现出巨大的应用潜力，提供灵活性、可扩展性、分布式数据存储和处理能力，以及弹性和成本优化。尽管如此，实际挑战仍然存在，包括确保跨多个云平台的数据一致性和集成。未来的研究和实践应优先考虑解决这些障碍并设计适当的解决方案，以推动混合云技术在制造运营管理中的持续进步和利用。

2.3 业务开发技术分析

在混合云下的分布式制造运营管理系统中，业务开发技术起着关键作用。本论文将从 Java 语言的角度对业务开发技术在该领域的应用进行分析。

Java 语言具有广泛的应用性和跨平台特性。作为一种面向对象的编程语言，Java 语言可以应用于各种业务场景和系统平台。在混合云环境下，Java 语言可以同时运行于私有云和公有云，实现跨平台的一致性和可移植性。这使得 Java 成为开发混合云下分布式制造运营管理系统理想选择。

Java 语言具有丰富的开发框架和工具。Java 生态系统中有许多成熟的开发框架和工具，如 Spring、Hibernate、Apache Hadoop 等。这些框架和工具提供了丰富的功能和组件，可以加快业务开发的进程。Spring 框架可以提供依赖注入和面向方面编程等功能，从而简化系统开发和维护。Java 语言还支持与大数据处理和人工智能相关的库和工具，帮助开发人员处理大数据并实现机器学习算法。

Java 语言具有良好的安全性和稳定性。在分布式制造运营管理系统中，数据的安全性

和稳定性至关重要。Java 语言提供了丰富的安全机制和库，使用 Java 加密扩展（JCE）和 Java 安全套接字扩展（JSSE）技术，可以实现数据加密，保证传输安全。Java 语言的垃圾收集机制和异常处理机制可以提高系统的稳定性和容错性。

总之，Java 语言在混合云环境下的分布式制造运营管理系统中具有重要的应用价值。它具有广泛的适用性和跨平台特性，具有丰富的开发框架和工具，具有良好的安全性和稳定性。未来的研究和实践应侧重于 Java 语言在分布式系统开发中的创新和优化，以促进其在业务开发技术领域的进一步发展和应用。

2.4 本章小结

本章对制造运营管理 MOM 系统相关技术进行了简要分析，为后续系统设计和实施提供了重要的技术支持和指导。通过对区块链、混合云和业务开发技术的研究，可以更好地构建和应用适合企业需求的 MOM 系统，提升企业生产运营的效率 and 竞争力。

3 MOM 系统需求分析

3.1 业务需求分析

为支撑企业灯塔工厂项目建设，制造运营管理系统 MOM 作为智能制造不可或缺的管理平台，在全流程生产过程中具有承上启下的作用。因此，MOM 平台的成功建设实施，会将产业园相关的人、机、料、法、环、测等生产要素，进行有效整合与高效联动，实现生产信息全流通，生产进度透明等目标，为公司实现智能制造打下坚实基础^[15]。运用人工智能、5G+工业互联网、集成柔性自动化设备，打造具有领先行业水平的“灯塔”工厂，建设国际化、高端化、数字化、智慧化、特色化生产基地^[16]。

混合云下的分布式制造运营管理系统是一种融合区块链、大数据和人工智能的解决方案，可以提高制造运营管理的效率和可靠性。

在企业生产制造管理过程中涉及到大量的传感器数据、设备数据和生产过程数据，因此系统需要能够实时监控、采集、存储和分析这些数据^[17]。通过实时监控和数据采集，能够及时掌握生产现场的状态和变化。通过数据存储，为后续的分析 and 决策提供所需的分析数据基础。通过对采集到的数据进行数据分析和数据挖掘，发现潜在的规律、异常情况和优化机会^[18]。系统可以通过大数据技术对生产过程中的关键指标进行分析，实现生产效率的提升和生产成本的降低。系统还可以应用机器学习和深度学习，对数据进行分析预测和优化，提供更精准的决策支持^[19]。系统还需要支持多用户的协同工作，使企业内不同部门的人员能够共享数据和协同决策，提高生产运营管理过程的协同工作效率。

在企业的生产经营管理过程中，数据可能涉及商业秘密，系统需要确保数据的机密性和完整性。区块链技术可以有效解决这些问题，满足企业日常生产活动的需求。该系统可以通过集成区块链相关技术，实现去中心化的数据存储和防伪，增强数据的安全性和可信度。

综上所述，混合云下分布式制造运营管理系统的业务需求不仅包括基本的业务需求，还包括实现生产数据实时监控、数据分析与挖掘、数据安全、生产链追溯等功能。实现这些功能将使该系统能够提供高效、可靠和安全的制造运营管理服务。未来的研究和实践应围绕这些需求，提供相应的技术和解决方案，促进系统在实际应用中的进一步发展和应用。

通过对部分车辆生产制造企业进行调研，总结出了车企的一般生产流程为，对接订单、生产排期、物料配送、产线流转、质量管控、工艺优化等主要流程。流程与流程之间互相

交错配合，交叉验证，持续迭代。本系统设计实施的参考范围为某公司汽车生产产业园下料车间、结构件车间（焊装、机加、涂装）、装配车间（含精饰线）。智能化车间功能如图 3.1 所示。



图 3.1 智能化车间

虽然当前诸多公司的供应链管理和生产管理的智能化和自动化程度较高，但仍存在较多的管理盲区，例如链路管理缺失、数据安全不够重视、缺乏企业内部知识库等。针对这些问题，将结合当下较为火热的技术进行探索。

3.2 功能需求分析

混合云下的分布式制造运营管理系统是一个综合信息平台，旨在协调企业内部各种生产流程的数据流和信息管理，实现生产流程的高效管理和优化。以下是对系统各功能模块的功能要求的分析。

企业主数据微服务：该微服务应提供对企业主数据的统一管理，包括但不限于员工信息、供应商信息、客户信息等。其功能要求包括数据输入、修改、查询和删除，以及数据分类和归档。

生产执行微服务：该模块应负责生产计划的执行和监控，包括生产订单管理、工单管理、生产进度跟踪等功能。还需要满足用户的需求，根据实际生产情况及时调整生产计划，并能够实时监控和分析生产过程。

流程管理微服务：该微服务应支持企业的流程管理需求，包括流程路线管理、流程参数配置、流程版本控制等功能。还需要满足用户的需求，用户应该能够通过该模块灵活管理和调整流程。

质量管理微服务：该模块应负责监测和管理产品质量，包括制定质量检查计划、输入和分析质量检查结果，以及处理缺陷产品。

仓库物料微服务：该微服务应支持企业的仓库物料管理需求，包括物料入库、出库和盘点等功能。满足用户需求，通过该模块实现对物料库存的及时监控和管理。

设备管理微服务：该模块应负责企业设备的管理和维护，包括设备档案管理、设备维护计划制定、设备故障报告等功能。满足用户通过该模块实现设备状态实时监测和维护的需求，保证生产设备的正常运行。

自动化运维平台：该平台应支持企业自动化设备的运维管理需求，包括设备远程监控、故障诊断与预测、设备运行数据分析等功能。还需要满足用户的需求，通过该平台实现自动化设备的智能管理和优化。

RAG 知识库系统：该系统应为企业提供一个知识管理和共享平台，包括生产过程知识、质量管理经验、设备维护技术等方面的知识库建设和管理。使用户可以通过该系统实现企业知识资源的集中管理和共享。

大数据系统：该系统应支持对生产和运营过程中产生的大量数据的收集、存储、处理和分析，以提供数据驱动的决策支持。用户应该能够通过该系统监控和分析生产过程数据，识别潜在问题，并及时进行调整和优化。

综上所述，本文设计的分布式制造运营管理系统功能需求涵盖了企业生产经营过程中的主要环节，能够满足企业生产经营管理的需求。本文重点介绍了这些需求，并提供了相应的技术和解决方案。

3.3 非功能性需求分析

制造运营管理系统是企业生产管理的重要组成部分，为了保证系统的安全性、易用性和可靠性，需要满足一系列非功能性需求。以下是针对本系统的非功能性需求列表：

（1）安全性：

用户身份验证：系统应提供安全的用户身份验证机制，包括密码验证、双因素身份验证等，以确保只有授权用户才能访问系统。

数据加密：对用户密码、基本用户信息、公司信息等敏感数据进行加密，防止数据泄露。

访问控制：实施严格的访问控制策略，限制非法用户的访问，保护系统免受恶意攻击和未经授权的访问。

（2）易用性：

直观界面：设计简单直观的用户界面，降低用户操作的复杂性和学习成本，提高工作效率。

操作流程优化：优化操作流程，简化用户操作步骤，减少系统中用户交互次数，降低错误率。

用户培训支持：提供相关的用户培训和使用指南，帮助用户快速入门，提高系统可用性和用户满意度。

（3）可靠性：

高并发支持：系统需要同时支持数万用户的并发访问，确保系统在高负载条件下仍能正常运行。

实时响应：系统响应时间要快，保证生产节奏，避免生产工作延误。

数据库性能：数据库应具有高性能和可用性，能够承载大量数据并保持稳定运行，确保生产数据的准确性和完整性。

灾难恢复备份：建立灾难恢复备份机制，定期备份和恢复系统数据，确保系统数据的安全可靠。

（4）性能优化：

系统性能监控：系统架构设计需要实现对系统性能指标的实时监控，包括 CPU 利用率、内存使用率、网络带宽等。

缓存优化：在设计系统架构时，需要使用缓存技术来提高系统访问速度和性能，减轻数据库访问压力，提高系统响应速度。

系统调优：在系统架构设计期间，需要对系统进行评估，在系统开发和测试期间，需要进行性能优化，以提高系统的稳定性和可靠性，确保系统能够满足生产要求。

总之，制造运营管理系统的非功能性要求包括安全性、可用性、可靠性和性能。通过实施这些要求，制造运营管理系统可以稳定地支持企业制造运营管理工作的顺利进行，提高企业生产效率。

3.4 小结

本章对 MOM 系统的需求进行了全面而深入的分析，为后续系统设计和实施提供了重要的参考和指导。通过对业务需求、功能需求和非功能性需求的分析，可以更好地把握系统的定位和设计方向，确保系统能够满足用户的实际需求，并为企业的生产运营提供有效的支持和保障。

4 系统设计

4.1 设计原则

系统架构设计的目标是满足用户的需求并适应未来的变化，在系统设计之初，就需要考虑到系统的可维护性、可扩展性和安全性。为了达到这种目的，本文将采用以下系统设计原则：

（1）模块化设计

模块化设计的核心理念是将系统拆分为多个独立的模块，每个模块负责实现特定的功能，通过各个模块之间的相互调用，实现整个系统，模块化设计显著降低了系统开发和维护的复杂度。通过模块化设计，可以将复杂的系统拆解为易管理的小而美的形态。开发团队可以并行开发和维护每一个模块，每个模块都可以独立测试和部署。

（2）数据驱动设计

数据驱动设计以数据作为设计的核心，从数据流的角度出发分析需求，并设计相应的数据模型和处理流程。系统分析师和系统架构师通过深入分析数据的特性和用户需求，设计出满足用户需求的系统架构和相应的算法。

（3）安全性设计

随着数据安全法的颁布，涉及到企业数据的部分都是敏感功能，安全性设计是系统设计的关键，可以有效地降低安全风险并提高系统的稳定性和可信度。本系统设计将采用规范的安全性设计原则，参考网络安全等级保护标准规范进行设计，保护系统和用户的数据安全。

（4）可扩展性设计

可扩展性设计是为了满足系统能够适应未来业务增长和变化的需求，通过合理的设计和规划，可以使系统在面对未来的挑战时能够灵活扩展。系统采用松耦合的架构和 Restful 的接口设计，便于支持系统的快速扩展和集成第三方组件。

（5）用户体验设计

用户体验设计是确保系统能够提供简洁、直观的界面和操作方式，以提高用户的工作效率和满意度，在设计阶段多于客户沟通，可以通过原型开发的开发方法，边开发边设计，使系统设计一直符合用户的期望，可以给用户提供优秀的用户体验。在系统设计阶段，软件设计师需要注重用户的需求和反馈，不断优化系统的界面和交互设计。

综上所述，通过结合以上系统设计原则，将确保制造运营管理系统具备高度的可维护性、扩展性、安全性和用户体验，从而为用户提供高效、稳定和可靠的服务。

4.2 架构设计

4.2.1 业务架构

在分布式制造运营管理系统中，业务架构扮演着至关重要的角色，它是系统设计的基础，通过定义和描述各项业务活动、流程和功能，确保系统能够满足企业的核心业务需求^[20]。基于 TOGAF 理论和 Architecture Development Method(ADM)方法，将对企业主数据微服务、生产执行微服务、工艺管理微服务、质量管理微服务、仓储物料微服务、设备管理微服务、自动化运维平台、RAG 知识库系统和大数据系统等业务架构进行详细阐述。

企业主数据微服务是业务系统的枢纽，负责集中管理企业生产经营的主数据。企业主数据微服务统一管理这些数据，保证了数据的一致性、准确性和完整性，为其他微服务提供了一致的共享数据，实现了企业业务数据的共享和利用，提高了系统业务数据管理效率。

生产执行微服务则是负责监控和调度生产过程的关键组件，实时监控工厂车间生产线的运行情况，负责调度生产任务以及优化生产流程。生产执行微服务管理生产订单、工单、生产计划，保证高效的生产过程，提高企业资源利用率。

工艺管理微服务负责制定和优化生产工艺，管理工艺参数、工序流程数据。工艺管理微服务通过及时调整工艺版本管理和工艺路线，保证生产过程的稳定性和可控性，进一步提高产品品质和工业化水平。

质量管理微服务主要功能是监控和控制产品质量，实时采集和分析生产过程中的质量数据，及时发现并处理质量异常，保证产品达到预期的质量标准，通过使品质控制达到统一标准，提高客户满意度和企业的市场竞争力。

仓储物料微服务负责管理着企业的物流和库存，实时跟踪物料流动、优化仓储布局并且智能化管理库存。仓储物料微服务通过实现物流信息的实时可视化和智能化管理，进一步提高了企业内物流效率和工厂仓储利用率。

设备管理微服务的主要功能是监控和维护生产设备，实时监测设备运行状态、预防性维护和故障诊断，管理着设备信息、维护记录，保障设备的平稳运行，减少了生产中断和损失。设备管理微服务还负责维护提醒维护班进行定期维护和清点工作，避免企业资产流失。

自动化运维平台为企业提供了便捷的运维管理平台，采用了智能化的自动化工具和算法，进行设备监控、故障预测、维护计划优化，实现运维数据的集中管理和智能分析，提高系统运维效率和可靠性。

RAG 知识库系统通过上传企业内部丰富的专业技术资料，为员工提供学习和参考资源，促进技术创新和知识共享，提高员工学习和工作效率，增强企业核心竞争力。

大数据系统为企业提供了数据分析和决策支持，通过采集、存储和分析企业制造运营过程中产生的海量数据，通过对数据的多维度分析与挖掘，为企业提供了精准的业务洞察和决策建议，助力企业实现智能化管理和持续发展^[21]。

综上所述，制造运营管理系统通过各个业务微服务的协同处理，实现了对企业制造运营过程的全面管理和优化，显著提高了企业的生产效率和竞争力。业务架构如图 4.1 所示。

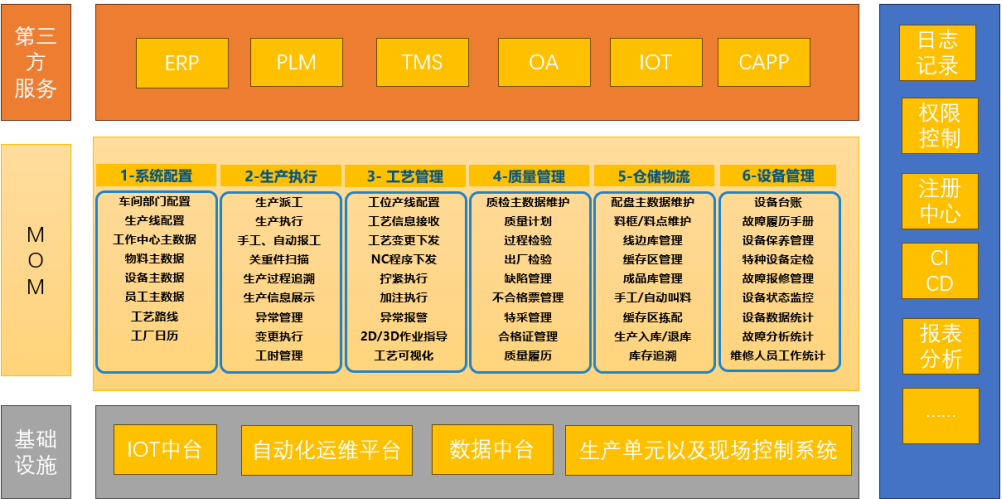


图 4.1 业务架构图

4.2.2 数据架构

在分布式制造运营管理系统中，数据架构是整个系统的核心组成部分，负责管理和组织各类数据，以支持企业的生产运营活动。基于 TOGAF 理论和 ADM 方法，对系统中的主要数据组织和管理进行设计和规划，主要涉及企业主数据微服务、生产执行微服务、工艺管理微服务、质量管理微服务、仓储物料微服务、设备管理微服务、自动化运维平台、RAG 知识库系统以及大数据系统等模块。

在数据架构设计中，通过数据架构师制定统一的数据标准和规范，确保各个微服务和应用系统之间的数据交互和共享具有一致性和可靠性^[22]。企业主数据微服务作为系统数据交汇的核心微服务，负责管理企业级的基础数据，通过建立统一的数据模型和标识体系，

实现对这些数据的集中管理和统一调用，保证企业核心数据的一致性。生产执行微服务、工艺管理微服务以及质量管理微服务等模块，负责生产过程中涉及的各类数据进行管理。通过与企业主数据微服务进行关联，实现对生产过程中所涉及的数据的准确记录和实时更新，支持开展生产计划的执行和质量管控的工作。仓储物料微服务和设备管理微服务负责管理物料和设备等资源。通过与生产执行微服务和工艺管理微服务等模块的数据交互，实现对物料和设备的有效调度和管理，确保生产过程的顺利进行。

自动化运维系统作为基础设施系统，负责支撑整个系统的正常运行，通过监控和管理生产设备的运行状态，实施自动化的设备维护和故障处理。自动化运维系统通过与设备管理微服务进行数据交互，实现了对计算机设备状态和维护记录的实时更新和管理，以提高系统的可靠性和稳定性。

RAG 知识库系统负责管理和分享企业的专业知识和经验，支持生产运营活动的持续优化和员工培养工作。RAG 知识库系统通过建立企业内部专业知识库，将企业内部的经验和教训进行归档和分享，为生产运营提供可靠的参考建议。

大数据系统作为数据中心的重要组成部分，主要负责收集、存储和分析各种生产活动产生的数据，还需要提供决策分析。大数据系统建立数据湖和数据仓库，管理和分析海量数据，分析和预测企业的生产经营，从而实现生产经营的智能化和优化。

综上所述，该系统利用 TOGAF 理论和 ADM 方法，结合分布式制造运营管理系统的实际用户需求，建立了统一的数据架构，将各种微服务和应用系统有机集成，促进数据共享和流通，支持企业制造运营活动的持续改进和优化。

4.2.3 应用架构

在分布式制造运营管理系统中，应用架构是系统的核心组成部分之一，它承担着实现系统核心功能和支持企业生产运营活动的重要责任。基于 TOGAF 理论和 ADM 方法，将应用架构划分为生产运营主系统、自动化运维系统、RAG 知识库系统和大数据中台系统四个部分，每个部分都包含了多个微服务和平台，共同构成了系统的应用架构。

生产运营主系统是分布式制造运营管理系统的核心，它包括企业主数据微服务、生产执行微服务、工艺管理微服务、质量管理微服务、仓储物料微服务和设备管理微服务等多个微服务。企业主数据微服务负责管理企业的核心数据，如产品信息、客户信息和供应商信息等；生产执行微服务负责执行生产计划、监控生产过程和协调资源调度；工艺管理微

服务负责管理生产工艺和工艺参数；质量管理微服务负责监控生产质量和执行质量管理流程；仓储物料微服务负责管理物料的入库、出库和库存；设备管理微服务负责监控设备状态和执行设备维护计划。这些微服务通过 Restful API 相互连接，共同构成了生产运营主系统的核心功能。

自动化操作和维护系统旨在提高生产和运营效率，降低手动操作和维护成本。自动化运维系统实时监测设备运行状态，预测设备故障，执行设备维护任务，实现设备的智能管理和自动化维护管理。

RAG 知识库系统旨在提高员工在制造和运营活动中的管理水平。RAG 知识库系统负责收集、组织和共享企业生产经营知识和经验，为生产经营决策提供参考和支持。并通过帮助员工不断学习，提升企业核心竞争力，响应不断变化的市场需求，帮助员工与企业共同成长。

大数据中台系统是为了实现数据共享和数据应用而设计的，主要包括数据存储和数据分析两个部分。数据存储的主要功能是收集、存储和管理企业的生产数据和业务数据。数据分析的主要功能是分析生产数据和业务数据，挖掘数据的价值。这两个部分通过数据湖和数据分析平台相互连接，共同构成大数据中台系统。

总之，基于 TOGAF 理论和 ADM 方法。分布式制造运营管理系统的应用架构设计包括四个核心子系统：生产运营主系统、自动化运维系统、RAG 知识库系统和大数据中端系统。MOM 业务主系统还包含多个子微服务。实现系统核心功能，支持企业日常生产经营活动。

4.2.4 技术架构

在构建分布式制造运营管理系统的技术架构中，充分遵循了 TOGAF 理论和 ADM 方法，以确保系统的稳定性、可扩展性和安全性。技术架构涵盖了企业主数据微服务、生产执行微服务、工艺管理微服务、质量管理微服务、仓储物料微服务、设备管理微服务、自动化运维平台、RAG 知识库系统和大数据系统等多个方面，为企业提供了完善的制造运营管理解决方案。

企业主数据微服务扮演着整个系统的基石角色。它通过集中管理企业核心数据，如客户信息、产品信息、供应商信息等，确保了数据的一致性和准确性，为其他微服务提供了可靠的数据基础。

生产执行微服务负责管理生产过程中的各项任务和资源分配，通过实时监控生产线情况、调度生产任务以及优化生产流程等功能，提高生产效率和产品质量，实现了对生产过程的智能化管理。

工艺管理微服务负责制定和优化生产工艺的重要任务，通过管理工艺参数、工序流程，保障了生产过程的稳定性和可控性，提高了产品的品质。

质量管理微服务负责监控和控制产品质量，通过实时采集和分析生产过程中的质量数据，及时发现和处理质量异常，保障产品达到预期的质量标准。

仓储物料微服务的主要功能是管理企业的物流和库存。通过实时跟踪物流、优化仓储布局 and 智能管理库存，提高仓储物流和仓储利用效率，降低企业运营成本。

设备管理微服务的主要功能是监控和维护生产设备。通过对设备运行状态的实时监测、预防性维护和故障诊断，确保设备的稳定运行，减少生产活动中断的发生和由此造成的损失。

自动化运维平台为企业提供了一个智能化的运维管理平台。通过自动化工具和智能算法，可以实现对信息设备的监控、故障预测和维护计划的优化，提高运维人员的工作效率。

RAG 知识库系统创建了一个企业级的技术信息和经验共享平台。RAG 知识库系统收集和整理公司内部的核心技术文件和案例研究，为公司员工提供学习和参考资源，促进公司的技术创新。

大数据系统为企业提供了数据分析和决策支持。通过收集、存储和分析业务系统生成的海量业务数据，可以发现数据之间的相关性和模式，为企业提供智能的业务见解和决策建议，帮助企业实现数字化和智能化管理。

综上所述，在分布式制造运营管理系统的技术架构过程中，本文充分考虑了微服务架构、自动化运维技术、大数据分析和人工智能等先进技术，为企业提供高效智能的解决方案，可靠的制造运营管理系统解决方案，帮助企业实现数字化转型。技术架构图如图 4.2 所示。

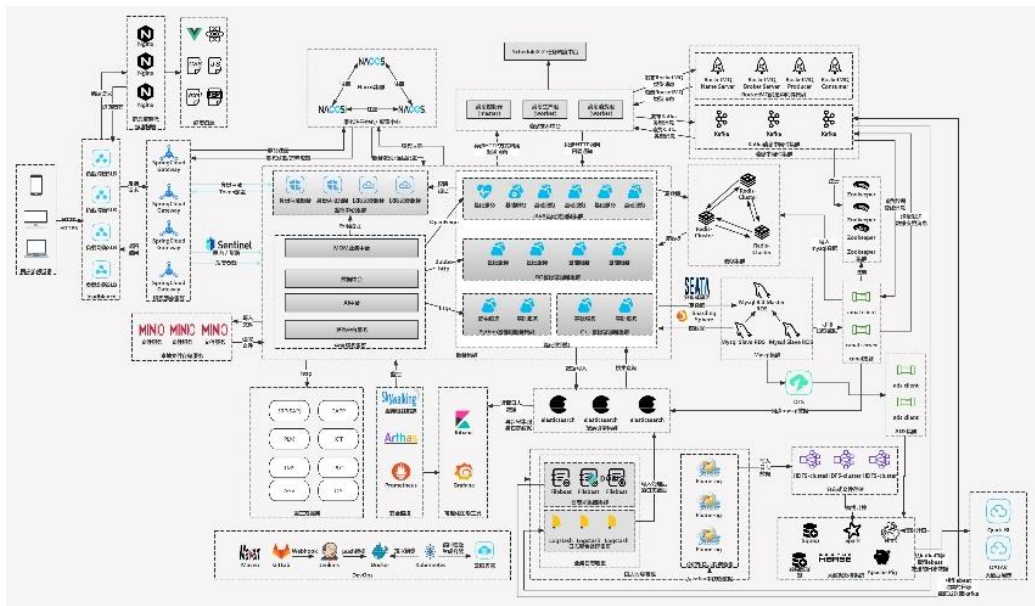


图 4.2 技术架构图

4.3 MOM 业务系统

4.3.1 业务架构设计

在构建企业级系统时，满足需求并考虑业务细节的功能设计至关重要。这样的系统不仅需要满足不同部门的具体需求，还可以保证不同部门之间的信息共享和数据流通，提高整个企业的协同生产运营效率和协同管理水平。以下是一个综合系统设计，集成了角色访问控制、应用程序功能、数据管理、技术架构和用户交互。

(1) 角色与访问控制设计

在一个大型系统中，对数据资源和功能的访问控制是确保数据安全和业务流程顺畅的关键。目前通常使用的是基于角色的访问控制方法进行权限体系的设计。系统将用户分为不同的角色，如生产部门员工、供应链管理部门员工、质量管理部门员工和销售部门员工等。每个角色根据其职责和需求被赋予相应的权限。例如，生产部门的员工可以查看和更新生产计划与进度，而供应链管理部门可以管理供应商信息和采购订单。销售部门可以管理销售订单和客户信息。通过精细的权限设置，确保了数据的安全性和业务流程的高效性。

(2) 应用功能设计

系统的核心应用功能设计围绕企业的核心业务流程展开，具体包括如下的几个核心业务功能模块。

生产计划管理应用模块：该应用使生产部门能够制定和调整生产计划，管理订单，进

行产能规划，确保生产活动与市场需求相匹配。

供应链管理应用模块：供应链管理部门通过此应用管理供应商信息，处理采购订单，监控库存水平，以保障物料的及时供应。

质量管理应用模块：质量管理部门利用该模块管理产品质量数据，进行质量检测，记录不良品信息，以持续提升产品品质。

销售管理应用模块：销售部门通过此应用管理销售订单，维护客户信息，分析销售数据，以提高销售效率和客户满意度。

（3）数据资产管理

数据是企业的核心资产，系统的数据库设计必须确保数据的完整性、准确性和可访问性。MOM 业务系统包含以下类型的数据：

生产数据：记录生产计划、进度、成本等关键信息，为生产决策提供数据支持。

供应链数据：包括供应商信息、采购订单和库存状态，以优化供应链管理。

质量数据：收集质量测试结果和缺陷产品记录，用于质量控制和改进。

销售数据：涵盖销售订单、客户信息、销售等，为制定销售策略提供依据。

（4）数据库与开发工具选择

数据库管理系统（DBMS）：选择可靠的数据库系统，如 MySQL、Oracle 或 MongoDB 来存储和管理企业数据。

应用程序开发工具：使用 Spring Boot、Spring Cloud Alibaba 等方便的开源开发工具和框架，方便系统开发和部署，确保系统稳定性。

前端框架：使用 React 和 Antd 等流行的前端框架，提高用户界面的交互性和用户体验。

（5）微服务架构设计

在微服务架构中，系统被拆分成多个独立的服务，每个服务负责特定的业务功能：

企业主数据微服务：管理企业的核心数据，如员工信息、产品信息等。

生产执行微服务：负责生产过程的执行和监控。

工艺管理微服务：管理产品的生产工艺和流程。

质量管理微服务：负责产品质量的检测和管理。

仓储物料微服务：管理仓库中的物料和库存。

设备管理微服务：监控和管理生产设备的状态和维护。

通过这种微服务架构，系统能够实现更高的灵活性和可维护性，同时允许各个服务独

立部署和扩展，以适应不断变化的业务需求。

（6）用户交互设计

用户交互直接影响着用户的工作效率和体验，一个好的人机交互设计应该满足三大黄金原则。系统支持以下交互活动：

生产部门可以根据实时订单和市场变化灵活制定和调整生产计划，及时了解市场需求的变化，并相应调整生产策略，从而提高生产效率和产品按时交付。供应链管理部门可以与供应商进行有效沟通，及时处理采购订单，以确保材料供应的连续性，优化供应链流程，降低库存成本，确保生产不受材料短缺的影响。质量管理部门可以对生产过程中的质量进行监控，及时发现和解决问题，促进质量的持续改进。通过及时的质量监控和问题解决，我们确保产品符合标准，提高客户满意度和市场竞争力。销售部门可以有效地管理销售订单，确保订单及时交付并满足客户需求。销售部门的工作效率直接影响客户满意度和企业绩效，因此系统设计应优先提高其操作的便利性和准确性。

在人机交互设计过程中，我们遵循将其置于用户控制之下、减少用户记忆负担、保持界面一致性的原则。通过精心设计的用户交互过程，该系统将提高用户的便利性，减少错误和延迟，从而全面提高整体工作效率和用户体验。

在 MOM 业务系统设计过程中，考虑到了从角色访问控制到技术实现的各个方面，通过高度集成和模块化的设计，提供了一个稳定、灵活且用户友好的解决方案，满足企业的持续发展的需求。

4.3.2 各模块数据库表设计

数据库表设计是建立数据库的基础，直接关系到数据的存储、管理和检索效率。在进行数据库表设计时，需要考虑到数据的特性、业务需求以及系统的扩展性和性能优化。

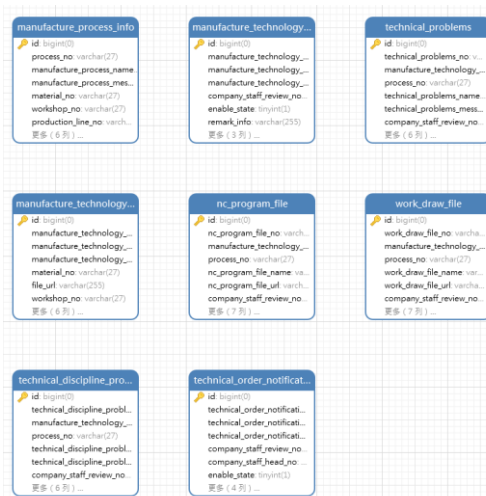
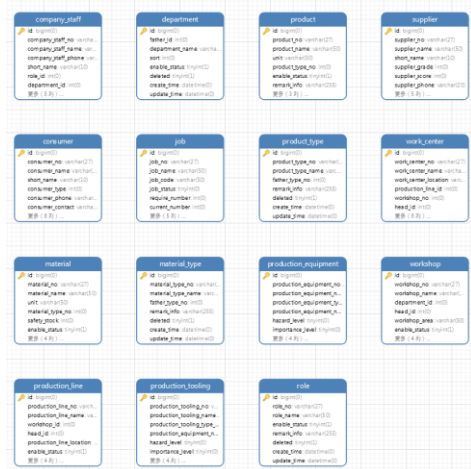
在进行数据库设计的时候需要明确每个表的名称和用途，确保表名能够清晰地反映表所存储的数据内容，方便开发人员和管理员理解和管理。对于每个表中的字段，需要明确定义字段的名称、数据类型、长度以及约束条件，确保数据的完整性和一致性。主键和外键的设计则是确保数据的唯一性和表之间关联性的关键，通过主键和外键建立表之间的关系，实现数据的引用完整性约束。由于当前较为新颖的软件开发实践中，出于易于维护的目的，基本上都不在数据库层面维护表与表之间的关系，转而使用代码逻辑进行控制。本章节设计的方法采用代码控制的方式进行控制。

在表设计中还需要考虑索引的创建。通过在经常需要查询操作的数据字段上创建索引，可以提高查询性能并加快数据检索。在设计数据表时，您需要遵循数据库设计原则，选择合适的数据库范式，并确保数据库架构设计在结构上合理、高效且易于维护。

综上所述，通过以上数据库设计方法，可以设计出一套高质量的数据库表。合理的表设计可以提高数据查询效率、系统性能和可维护性。如表 4.1 是系统数据库表设计的一些细节。

表 4.1 数据库设计表

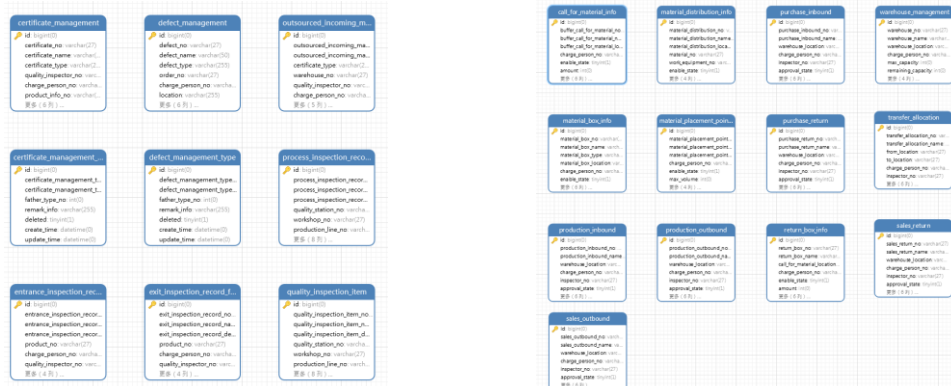
(2) 工艺管理微服务



(4) 生产管理微服务



(6) 仓储物流管理微服务



4.4 自动化运维系统设计

需求概述：自动发现主机，自动接入，自动运维部署，DevOps 平台采用现有的云效，提供自建 DevOps 的接口保留。目前实现自动化运维，MOM 系统服务的自动化运维，资产管理，大数据系统的自动化运维。

结合 4A 架构理论和 TOGAF 理论，以下是详细的应用架构设计：

（1）持续集成（CI）和持续交付（CD）

使用自动化工具（如 Jenkins、GitLab CI 等）实现持续集成和持续交付，确保代码质量和交付速度。实施自动化测试和部署流程，提高开发和发布效率。

（2）自动化部署和容器化

使用 Docker 容器技术将应用程序与其依赖项打包，实现环境一致性和快速部署。结合 Kubernetes 等容器编排工具，实现自动化部署、水平扩展和故障恢复。

（3）监控和日志管理

在日常系统运维项目中，系统监控和日志监控是系统运维的重要手段。通过集成监测工具（如普罗米修斯和格拉法纳）实时监测系统性能和健康状况，可以及时发现系统异常。通过配置日志管理系统（如 ELK Stack）、记录系统日志并分析问题原因，可以防止故障并提高未来的故障排除效率。

通过以上的自动运维系统架构设计，可以实现制造运营管理系统的快速开发、稳定运行和持续交付，可以满足多方需求并保持系统的可维护性和可扩展性。

4.4.1 自动扫描发现资产模块

资产发现是自动化运维的首要任务之一，它包括发现网络中的所有主机、设备、服务和应用程序，并对其进行识别、分类和管理。

传统的手动资产发现方法存在效率低下、容易出错等问题，因此自动化资产发现技术成为了网络安全领域的热门话题。Ansible 是一种流行的自动化工具，它可以通过 SSH 协议远程管理和配置多台计算机。结合 Python 编程语言，可以编写灵活的脚本来实现自动化资产发现。

使用 Ansible 编写 Playbook，定义资产发现任务和规则。用 Python 编写自定义模块或脚本，与 Ansible 结合使用，实现资产发现的具体逻辑。在网络中部署 Ansible 控制节点，通过 SSH 协议连接到目标主机，并执行资产发现任务。Python 脚本在目标主机上执行各种

命令或查询，收集主机信息并上传到控制节点。控制节点接收并处理所有主机的信息，进行资产识别、分类和管理。

自动化资产发现模块适用于各种网络环境和各种规模的组织，特别是种大型企业。通过自动化资产发现，可以自动发现和管理云端资源、物联网设备、容器、虚拟机等各种类型的资产。通过定期执行资产发现任务，可以及时更新资产清单，提高管理颗粒度。

4.4.2 任务自动执行模块

任务管理和自动执行是许多组织和企业中重要的工作流程，它可以提高效率、减少错误，并确保任务按时完成。在信息技术领域，Ansible 和 Python 是两个强大的工具，结合它们可以实现灵活、可靠的任务管理和自动执行方案。

以下是基于 Ansible 和 Python 的任务管理和自动执行流程。

运维工程师先在管理主机上安装 Ansible，并设置好必要的配置文件，如 hosts 文件和 ansible.cfg 文件。使用 YAML 语言编写 Playbook，定义任务的执行步骤、主机组、变量和条件等。Playbook 可以包含多个任务，每个任务可以调用不同的模块来完成特定的操作。根据用户需求，使用 Python 编写自定义模块，以满足特定任务的需求。自定义模块可以调用底层的 API 和库，实现更高级的功能。通过 Web 可视化界面执行 Ansible Playbook，Ansible 会将任务分发到目标主机，并按照定义的步骤进行自动化执行。在执行过程中，Ansible 会调用相关的模块和自定义模块，完成各种操作。

Ansible 还可以提供任务执行的实时监控和详细报告，帮助运维工程师了解任务的执行状态和结果。同时，它可以与 Python 的数据处理模块相结合，根据需要生成定制的报告。

基于 Ansible 和 Python 的任务管理和自动执行模块广泛应用于系统管理、应用程序部署、配置管理和数据处理等系统运维活动。在服务器管理中，可以使用 Ansible 和 Python 自动执行常见的管理任务，如用户管理、软件安装、服务启动和关闭等。在部署集群应用程序时，可以利用 Ansible 自动化应用程序部署过程，减少手动干预，提高一致性。

任务自动化模块可以帮助组织和企业提高效率，减少错误，并确保任务按时完成。Ansible 提供了强大的任务编排和执行能力，Python 具有良好的可扩展性。使用这两个工具，可以构建一个强大的自动化运维系统解决方案，以满足各种规模的自动化运维管理需求。

4.5 企业知识库

4.5.1 架构设计

近年来，生成式模型的迅猛发展为自然语言处理领域带来了突破性进展。尤其是大型语言模型（Large Language Models LLM）领域的发展，检索增强生成模型（RAG）的出现，为构建企业内部知识库系统提供了全新的思路和方法^[23]。RAG 系统通过检索与用户提出问题相关的数据来增强生成文本的能力，其基本工作流程包括检索、解码和融合。在企业内部知识库系统的存储、查询和更新的环节中，RAG 系统发挥着重要的作用，通过应用 RAG 理论设计企业内部知识库系统，可以增强查询能力，提高回答的正确率和准确度，对于专业领域，可以使得大模型不经过微调也可以具有很好的效果。

基于 RAG 理论设计企业内部知识库系统，可以帮助员工快速获得所需信息，从而显著提高员工的工作效率，提高整体工作效率。它也有助于增强企业的核心竞争力，通过积累和沉淀知识来提高企业的核心竞争能力。还可以降低企业的运营成本，降低手工知识管理的成本，降低企业的整体运营成本。基于 RAG 理论设计企业内部知识库系统是提高企业知识管理水平、增强企业核心竞争力的重要途径。通过充分利用 RAG 系统的优势，企业可以更有效地积累企业知识，更方便地培训企业员工，在激烈的市场竞争中占据领先地位，实现可持续发展的目标。

RAG 系统是一种检索增强生成模型，它通过检索与输入文本相关的文档来增强生成文本的能力。RAG 系统架构主要分为三部分，其中检索模块负责检索与输入文本相关的文档。解码模块负责对检索到的文本进行 Prompt 工程。融合模块负责调用 LLM 融合检索结果和生成结果，生成最终的输出。

检索模块负责检索与输入文本相关的文档。常用的检索方法包括：基于关键词的检索：根据输入文本中的关键词进行检索。基于语义的检索：根据输入文本的语义进行检索。

编码模块负责对检索到的文档进行编码。在进行 Prompt 工程的过程中，编写 Prompt 提示词是较为新的领域，目前学术界对于此问题还处于探索阶段。

融合模块负责融合检索结果和生成结果，生成最终的输出。这一阶段主要是考察模型本身的生成能力，目前的 LLM 通过测评推理能力可以近似小学生的水平，结合庞大的知识库，生成的结果具有一定的参考价值。

LangChain 是一个开源的自然语言处理框架，可以用于构建各种自然语言处理应用，

LangChain 架构图如图 4.3 所示。基于 LangChain 框架，可以构建一套 RAG 系统。



图 4.3 LangChain 架构图

综上所述，RAG 系统是一种先进的自然语言处理技术，可以应用于各种场景。基于 LangChain 开源框架，可以构建一套 RAG 系统，以满足企业的知识管理需求。

4.5.2 Embedding 算法分析

Embedding 算法是一种将离散符号（如词语、句子等）转换为张量的技术。Embedding 算法可以有效地保留符号之间的语义关系，并将其表示为张量形式，按照在张量空间的距离表示相似程度，方便后续的机器学习任务训练。

常用的 Embedding 算法包括：

- （1）Word2Vec：使用 Skip-Gram 模型或 CBOW 模型训练词向量。
- （2）GloVe：使用全局矩阵分解的方法训练词向量。
- （3）FastText：使用哈希技巧提高训练速度的词向量模型。
- （4）BERT：使用 Transformer 模型训练的词向量模型。

在 RAG 系统中，Embedding 算法主要用于以下两个方面：

- （1）检索：将输入文本和文档进行 Embedding，并使用相似度计算方法进行检索。
- （2）生成：将检索到的文档进行 Embedding，并使用注意力机制将其融入到生成过程中。

选取一个优秀的 Embedding 系统对于 RAG 系统的性能至关重要。一个优秀的 Embedding 系统应该具备以下特点：

- （1）语义表达能力强：能够有效地保留符号之间的语义关系。
- （2）向量空间稠密：能够充分利用向量空间的表达能力。

(3) 训练速度快：能够满足大规模数据训练的需求。

在实际应用中，可以根据具体的应用场景和需求来选择合适的 Embedding 算法。目前比较常见的是使用 Transformer 模型训练的词向量模型，本文采用智谱清言提供的词向量接口，也支持使用 Hugging Face 上开源的本地部署词向量模型。

综上所述，Embedding 算法是 RAG 系统的重要组成部分，选取一个优秀的 Embedding 系统对于 RAG 系统的性能至关重要。

4.6 系统集成

企业应用程序集成是将不同的软件应用程序、系统和技术集成到一个统一的环境中，使它们能够相互通信、共享数据，并实现应用程序系统和业务流程的集成。这种整合可以帮助企业提高效率、降低成本，重塑企业的战略模式和竞争优势。

基于区块链、大数据和人工智能技术的混合云综合应用的分布式制造运营管理系统可以实现高效的数据共享、智能分析和自动化运维，使业务系统高效运行。该系统分为四个主要部分：企业业务应用系统、大数据系统、问答系统和自动化运维系统，并利用企业应用集成实现它们之间的紧密集成。

企业应用系统 MOM 系统是公司日常运营的核心系统。通过企业应用集成，将外围系统与 MOM 系统集成，实现数据的实时同步共享。企业应用系统生成的数据可以传输到大数据系统进行存储和分析，自动化运维系统实时监控业务系统的稳定性和正常运行。

大数据系统负责存储、处理和分析制造运营过程中产生的大量数据，为企业提供商业见解和决策支持。通过与分布式 MOM 系统的集成，可以实现数据的实时采集和流式处理。大数据系统可以使用机器学习技术进行数据挖掘和数据分析，发现数据中的模式，并提供准确的预测分析。

RAG 系统可以通过 LLM+RAG 技术理解用户的查询，并从公司内部知识库中提取相关信息进行回答。通过与 MOM 业务系统的集成，RAG 系统可以实时获取最新数据，并将生成的结果返回给用户。

自动化操作和维护系统使用人工智能和自动化技术来监控、管理和优化企业的 IT 基础设施和应用程序。自动化运维系统通过与 MOM 业务系统集成，可以实时获取各个子系统和计算机系统的监控数据和报警信息，并能快速响应，确保企业信息系统的稳定性和可靠性。

综上所述，将 RAG 系统、大数据系统和自动化运维系统与 MOM 业务系统相集成，可以为企业提供更加智能化的解决方案。通过企业应用集成，使得这些系统高效协同工作，满足企业的数字化转型和业务优化需求。

4.7 本章小结

本章深入探讨了在混合云环境下构建分布式制造运营管理（MOM）系统，设计及相关子系统，包括 MOM 业务系统，自动化运维系统、大数据和企业知识库。通过明确设计原则，确保系统具备可扩展性和高效性。MOM 业务系统的设计注重业务功能实现，应对复杂的生产实际需求，自动化运维保障系统稳定性，大数据系统提供业务数据处理和业务数据分析，知识库提供员工团队建设和企业内部知识共享。系统集成确保各子系统协同工作，提升整体效率。

5 RAG 系统召回准确性问题研究

5.1 RAG 系统及研究问题简述

Question Answering (QA)系统是一种基于机器学习技术的智能问答系统，可以正确回答用户提的问题。QA 系统通过自然语言处理技术和机器学习算法，数据中提取有用信息，生成正确的答案。本文设计的 RAG 系统也是 QA 系统的一种，可以通过分析用户的问题，理解用户的意图，然后在企业知识库中检索出相关的数据，提供即时服务和准确的答案。如今的 QA 系统通常使用预训练的语言模型，例如 GPT、Bert、LLAMA 等生成式模型，和专门的检索召回算法，用于进行高效的信息检索和答案生成。

RAG 系统结合了信息检索和自动生成的技术，从大规模文本数据中检索相关数据，利用这些数据生成答案。RAG 系统使用了向量检索模型，来从企业知识库或其他数据中检索相关信息，并将检索结果，通过 Prompt 工程后，发送给生成模型，生成答案。这种结合检索和生成的方法使得 RAG 系统能够生成更加详细、准确的答案，同时充分利用了大规模文本数据。RAG 系统概念架构图如图 5.1 所示。

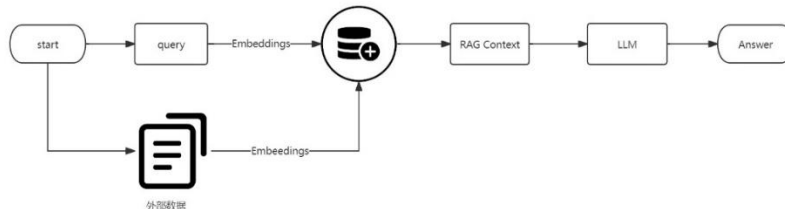


图 5.1 RAG 系统

企业知识库是一个企业内部的业务信息资源库，包含了企业的各种文档、报告和数据信息。这些信息通常经过编排和分类和信息化处理，使得企业员工能够方便地访问和使用。企业知识库是私有域的 QA 系统和 RAG 系统的重要数据源，提供了丰富的数据信息，可以用于回答员工的问题和生成具有专业领域背景的回答。企业知识库也是企业内部知识管理的重要工具，能够帮助员工更好地共享和利用组织内部的知识和经验。

综合来看，QA 系统、RAG 系统和企业知识库三者之间存在着紧密的三角关系。QA 系统可以基于 RAG 系统进行增强，并结合企业私有的知识库，利用企业知识库中的信息来回答员工的专业问题。企业知识库也可以通过这些系统在企业内部，员工群体中实现更加高效的知识管理和共享。通过 RAG 系统可以使得企业内部提高信息利用效率，加强内部

沟通与协作，推动业务发展和创新。

RAG 系统潜力巨大，但在实际的企业部署应用过程中，仍然存在着以下几方面问题。

(1) RAG 系统在检索阶段可能会有精准度的问题。精准度问题是指从数据集中检索召回到的 chunk 不是全部都与用户查询相关，存在 chunk 内部分内容与用户查询相比毫无关联甚至语义相反。这种情况下，生成的答案可能会失去准确性，进一步增强 LLM 的幻觉现象，导致用户认为系统效果不佳，无法提供有效的信息^[24]。

(2) RAG 系统可能存在低召回率的问题。即使所有相关的块都被检索到，但是信息还是不足。导致 LLM 缺乏足够的上下文信息，使得 LLM 在生成答案时表现不佳^[25]。

(3) RAG 系统可能受到过时信息的影响。知识库假设没有持续更新或由于数据冗余，检索模型检索到的内容可能不再具有价值，从而影响到生成的答案的质量。

在答案生成阶段，模型可能会产生幻觉，编造出不在上下文中的答案，或者生成与问题无关的答案，需要对模型进行调整或增加适当的约束条件，降低模型生成的不确定性，使生成的答案符合预期。模型生成的答案还可能会带有有害或令人反感的内容，这将严重影响用户体验，并造成不良的社会影响。在 RAG 系统答案生成过程中，需要加强模型的道德规范，对数据的筛选，来减少这种问题的发生。模型输出的质量也可能受到上下文信息不足、格式错误或不完整等因素的影响。为了提高生成答案的质量，需要在系统设计和实施中更加重视上下文的传递和数据处理的准确性，以及对输出格式的规范和完整性的管理。

另外，在 Retrieval 阶段也可能会遇到无法处理更大数据量和速率限制错误的问题。数据规模的局限性会影响系统的整体效率，随着企业的发展和业务规模的规模不断增长，系统内会存储着越来越多的数据。Retrieval 阶段可能无法快速地处理这些大规模数据集，这将导致 RAG 系统检索性能不佳。在 Retrieval 阶段，系统也可能受到计算机系统性能和检索算法的限制，导致检索速度较慢或甚至出现错误。

综上所述，RAG 系统在实际应用中可能面临检索阶段低精度、低召回率，以及过时信息等问题。在生成阶段可能出现幻觉问题，Retrieval 阶段存在难以处理大规模数据和检索算法速率限制的问题。

确保源数据质量是构建高效 RAG 系统的重要一环。如果数据质量很差，即使 RAG 系统设计再好，也无法从这些低质量数据中检索到准确的结果。因此，清理数据是至关重要的。清理数据的常见策略包括消除噪音和不相关的信息，这涉及删除特殊字符、停用词和 HTML 标记等方法。删除重复数据也是一个重要的方法，这些重复记录或类似记录可能会

干扰检索过程，降低系统的性能。为了构建高效 RAG 系统可以通过清理噪音数据、删除重复数据等策略提升数据质量。

为解决以上问题，本文提出了一种 **Chunk-Tree** 的新型向量存储检索结构。该结构基于 B+树思想对向量数据进行存储，并结合高效的近似搜索技术，实现了在大规模文本数据上快速高效的信息检索。

5.2 RAG 系统模型构建

RAG（检索增强生成）系统的架构设计包括两个主要组件：索引和检索/生成。

（1）索引组件的工作流程

加载（Document Loaders）：在这一阶段，系统通过文档加载器读取原始数据，从数据源中获取数据并将其加载到系统中。

拆分（Text Splitter）：使用文本拆分器，将较大的文档分割成更小的段落或块，可以提高检索效率和适应模型的有限上下文窗口，便于索引和后续处理。

存储（Store）：将拆分后的文本块存储到向量数据库中，后续可以通过查询向量数据库，检索它们并反向召回原始文本。

（2）检索和生成组件的工作流程

检索（Retriever）：给定用户的查询，使用检索器从存储中检索相关的文本块。这些文本块包含可能与用户查询相关的信息。

生成（Chat Model/LLM）：利用包含问题和检索到的数据的提示，生成答案。这个过程由聊天模型或者 LLM 完成，根据检索到的信息和用户提出的问题来生成答案。

（3）提示更新和上下文化问题

提示更新：为了支持历史消息作为输入，需要更新提示。在与 LLM 进行交互时，将历史数据通过 Prompt 工程全量或压缩传输给 LLM，让 LLM 在生成答案时考虑到用户之前的交互历史。

上下文化问题：使用 LangChain 添加一个子链，重新表述最新的用户问题，并在 Prompt 工程中将其加入提示词，使系统能够理解并正确地回答带有上下文的问题。

RAG 系统实现架构图如图 5.2 所示。

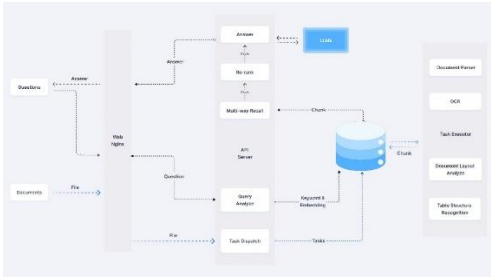


图 5.2 RAG 系统实现架构图

5.3 Chunk-Tree 查找算法分析

5.3.1 B+树结构分析

B+树是一种常用的数据结构，用于在计算机科学中实现有序数据的存储和快速检索。它是 B 树的变体，在 B 树的基础上进行了优化和改进。

B+树是一种平衡多路搜索树，从根节点到叶子节点的路径长度都是相等的。平衡性使得在 B+树中进行查找、插入和删除等操作时的性能稳定，不容易出现性能退化的现象。B+树的节点分为非聚簇节点和聚簇节点两种类型。非聚簇节点用于存储索引信息，指向子节点或聚簇节点。聚簇节点存储实际数据和指向相邻聚簇节点的指针。与 B 树不同，B+树的叶子节点包含了全部的关键字和对应的数据指针，而非仅仅是部分关键字。这样可以减少对数据的访问次数，提高了查询的效率。

B+树的叶子节点是一个有序链表，可以实现快速的范围查询和顺序访问，非常适合进行范围查询。B+树采用分层索引的方式组织数据，在存储大规模数据时仍能保持较低的高度，提高了查询效率。

综上所述，B+树的设计使得它在处理大规模数据时具有较高的性能和效率，适用于需要频繁范围查询和顺序访问的应用场景。

5.3.2 Chunk-Tree 查找算法设计

在 RAG 系统中，Memory 层当下的策略是将用户文本进行 chunk 切分，针对语义问题，采用了滑动窗口来解决这种问题。在 CRAG 论文中，提出了一种新的考虑，chunk 的切分如果较大，就会存在错误或无关信息误导 LLM 的情况，虽然 LLM 有一定的能力去自主解决，但是解决这种问题将会提高 RAG 系统的质量。论文中提出了一种思想，二次 Chunk 切分，取出相关的 Chunk 后，再次进行 embedding，检索出较高相关度的内容，给大模型

进行询问。

基于 B+树的思想，结合 Embedding 算法，设计了 Chunk-Tree 结构。对每个文本创建一个 Chunk-Tree，对文本进行切分和词嵌入计算，将 embedding 后得到的向量作为 B+树的主键，进行存储。Chunk-Tree 的结构如图 5.3。

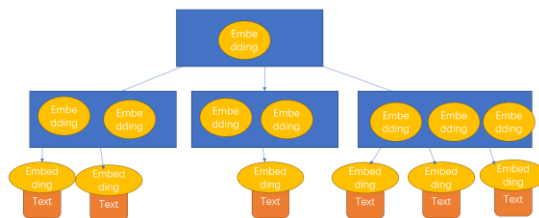


图 5.3 Chunk-Tree

对于查找算法的改动有，每次进行查询之前，首先比对根的词向量，快速确认文档是否跟文本有关，如果有关，然后对每一层进行查询，如果无关将不再对这个分支进行查询，如果有关，将继续查询到叶子节点，取出数据。这种结构，有效提高了查询速度和准确度，减少了无关内容对 LLM 的误导。

针对 Chunk-Tree 结构进行性能分析：

(1) 查找操作：Chunk-Tree 的查找操作时间复杂度为 $O(\log n)$ ，其中 n 是 Chunk-Tree 中叶子节点的数量。

(2) 插入和删除操作：Chunk-Tree 的插入和删除操作时间复杂度为 $O(\log n)$ 。虽然插入和删除可能需要进行节点的分裂和合并，但这些操作的代价较小，且不会对整棵树的平衡性产生太大影响。

(3) 范围查询：在范围查询方面 Chunk-Tree 由于所有数据都存储在叶子节点上，因此可以通过根据父节点范围来遍历叶子节点实现范围查询，不需要遍历整棵树，极大减少了时间复杂度。

(4) 顺序访问：Chunk-Tree 的叶子节点按链表结构，顺序链接，支持高效的顺序访问。

Chunk-Tree 时空复杂度综合分析：

(1) 时间复杂度：Chunk-Tree 的查找、插入和删除操作的平均时间复杂度都是 $O(\log n)$ ，其中 n 是树中关键字的数量。Chunk-Tree 的节点具有较大的容量，树的高度较小，一般不超过 5 层，各种操作性能都比较稳定。

(2) 空间复杂度：Chunk-Tree 的空间复杂度取决于数据量。节点的大小通常受到文本内容大小的影响，而树的高度受到关键字数量和节点大小的影响。Chunk-Tree 的空间复杂度较低，且随着数据量的增加增长较慢。

5.4 本章小结

本章围绕 RAG 系统召回准确性问题展开研究，首先介绍了 RAG 系统及研究问题，然后详细阐述了 RAG 系统模型构建过程，以及 Chunk-Tree 查找算法的分析和设计。通过对 B+树结构和 Chunk-Tree 查找算法的深入探讨，为提升 RAG 系统的召回准确性提供了一定的理论支持和技术方案。

6 系统实现与测试

6.1 开发工具介绍

6.1.1 后端开发技术

本文在后端开发过程中基于 Java、Spring Boot 和 Spring Cloud Alibaba 进行后端开发，以下简要地解释一下它们的作用和关系。

Java: Java 是一种基于 JVM 的跨平台编程语言，被广泛应用在后端开发领域。Java 具有丰富的生态和强大的开源社区支持，常常被用于构建各种服务器应用。

Spring Boot: Spring Boot 是 Spring 框架的一个子项，用于简化和加速基于 Spring 的应用程序的开发和部署。它提供了自动配置、快速开发和零配置等特性，使得开发者能够更加专注于业务逻辑的实现而不必花费太多精力在配置上。

Spring Cloud Alibaba: Spring Cloud Alibaba 是阿里巴巴基于 Spring 框架开发的一套分布式开发开源框架。它提供了一整套开箱即用的微服务组件，包括服务注册与发现、配置管理、负载均衡、分布式网关。开发者可以基于 Spring Cloud Alibaba 快速搭建自己的分布式系统。

这三个技术在 Java 生态中占有举足轻重的地位，使用它们可以快速构建起一个稳健、高效的后端系统。使用 Spring Boot 大幅简化了应用程序的开发复杂度。Spring Cloud Alibaba 提供了完整的分布式系统解决方案和微服务架构支持。开发工程师能够基于这些技术框架更加便捷地构建和管理复杂的分布式系统。

结合区块链开发技术时，主要需要选择合适的合约和加密算法。

智能合约: 智能合约是在区块链上执行的自动化合约，可以实现去中心化应用(DApp)的核心业务逻辑，并且具有自动执行、不可篡改、去信任等特点。

加密算法: 加密算法在区块链中起着至关重要的作用，包括数据加密、数字签名、哈希算法等，用于保护交易的隐私性和安全性。

在区块链开发技术和后端开发结合的场景中，通常会将智能合约作为区块链系统的核心业务逻辑部分，后端开发工程师则负责与智能合约进行交互，并提供 Web 接口供前端或其他服务调用。后端开发过程中还需要考虑如何与分布式账本技术进行交互，包括查询区块链数据、监听区块链事件等。安全性也是非常重要的一点，需要在后端开发中实现对

用户身份和数据的有效保护。

综上所述，将区块链开发技术和后端开发结合，需要软件开发工程师熟悉区块链技术的基本概念和智能合约的编写，还需要具备后端开发的技能，熟悉后端开发框架、数据库操作、API 设计等。通过无缝集成两者的业务系统，将为企业制造运营管理的数字化转型牢固基石。

6.1.2 前端开发技术

本文在后端开发过程中基于 TypeScript、React 和 Ant Design 进行前端开发，以下简要地解释一下它们的作用和关系。

TypeScript: TypeScript 是一种静态类型检查的编程语言，它是 JavaScript 的一个超集，基于 JavaScript 演化而来，所有的 JavaScript 代码都可以在 TypeScript 中运行。TypeScript 的代码将更加可靠和易于维护，能够在编译时发现潜在的错误，弥补了 JavaScript 在前端开发的不足。

React: React 是由一个由 Facebook 公司开源的，用于 Web 开发的 JavaScript 组件库，采用了组件化的架构设计。在 Web 开发工程师开发 Web 界面的过程中，将页面拆分成多个独立的组件，每个组件负责管理自己的状态和 UI。React 很受欢迎，具有强大的生态和活跃的开源社区支持。

Ant Design (Antd): Ant Design 是由蚂蚁金服开发的一套企业级的开源的 UI 库。Antd 提供了一系列精美的 UI 组件和模板，Web 开发工程师可以使用 Antd 快速构建高质量的前端界面。Antd 遵循良好的开发设计规范，支持国际化和响应式布局，被广泛应用于各种前端项目中，是当前流行的前端开发工具。

基于 TypeScript、React 和 Antd 可以构建起一个美观前端应用程序。TypeScript 提供了类型安全和更好的编译期检查，程序员可以开发出健壮的 Web 程序，使用 React 可以进行高效的组件化开发，另外使用 Antd 提供的丰富的 UI 组件，使得开发者能够快捷地构建出符合用户期待的人机交互界面。

6.1.3 大数据开发技术

大数据技术是利用先进的数据处理和分析方法，从海量数据中提取有用信息的过程。大数据的流程包括数据收集、清洗、存储、处理和分析，通过数据分析与挖掘，发掘数据

的价值，为企业决策提供数据支撑。如今大数据技术已经被应用于各个领域，例如市场营销、金融、生产管理等行业。大数据技术可以帮助企业发现商机、优化运营、降低风险，实现更加智能化的管理。大数据技术发展 20 余年，如今的现代大数据系统架构在 Apache Hadoop 生态之上。Hadoop 生态体系的相关组件如图 6.1 所示。

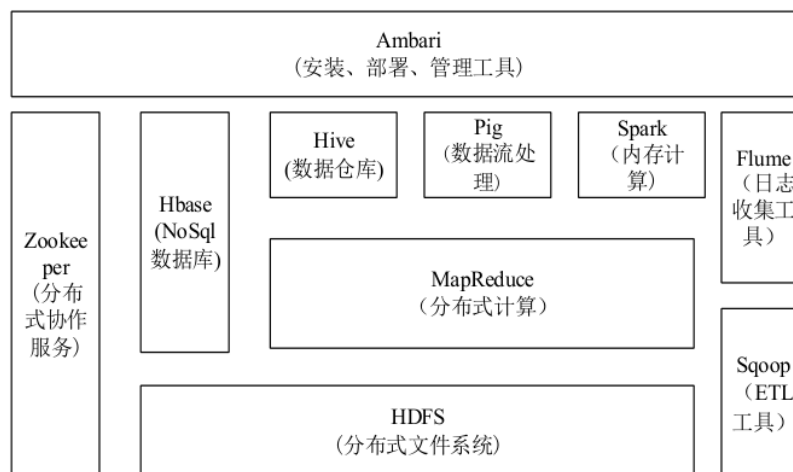


图 6.1 Hadoop 生态

当谈到大数据生态系统时，Hadoop 生态已经是通用标准，现代的大数据大厦建立在 Hadoop 生态之上。以下简要分析一下它们的作用：

HDFS (Hadoop Distributed File System): Hadoop 分布式文件系统，用于存储大规模数据集的分布式文件系统。它将数据分成小块，分布在 Hadoop 集群的多个节点上，并提供高可靠性和容错性。

MapReduce (MR): Hadoop 中的一种编程模型和处理框架，用于在 HDFS 上进行分布式数据处理。MapReduce 将任务分成 Map 和 Reduce 两个阶段，分别处理数据的映射和归约。作为 Google “三大马车”的一员，是如今大数据体系的理念基石，后续众多软件基于此理念进行架构设计。

Spark: Apache Spark 是一个开源的专为大规模数据处理而设计的分布式计算系统，提供了比 MapReduce 更丰富的功能和更高的性能。支持实时数据流处理和批处理。Spark 拥有丰富的 API 支持各种编程语言，还提供了易于使用的库，简化大数据处理的复杂性。

Flink: Flink 是一个热门的开源实时流处理框架，相较于 Spark 有更低的延迟和更高的吞吐量，常被用于辅助构建数据湖。Flink 适用于实时分析、事件驱动应用和复杂的流式数据处理场景，也为多种编程语言提供了易于使用的 API。

Hive: 一个基于 Hadoop 的构建数据仓库基础工具，也常常被用于构建数据湖。Hive 定义了一种类似于 SQL 的查询语言 (HiveQL)，基于 MapReduce 理念设计，方便用户在大

数据集上进行查询和分析。

Zookeeper: 一个分布式应用程序协调服务, 采用 ZAB 协议, 保证集群的分布式一致性, 用于管理和维护分布式系统中的配置信息、命名服务、分布式同步和组服务。

Ambari: 一个用于管理、监控和配置 Hadoop 集群的开源管理工具。Ambari 提供了一个用户友好的 Web 界面, 相较于采用脚本进行管理, Ambari 使管理员可以轻松地管理 Hadoop 集群的各个方面。

Kafka: 一个高性能的分布式消息队列系统, 用于处理大量实时数据流。Kafka 通过将数据分成多个分区并分布在多个服务器上来实现高吞吐量和可伸缩性。

这些技术通常一起使用, 构建起一个完整的大数据处理和管理解决方案。例如, 可以使用 HDFS 存储数据, 通过 MapReduce 或 Spark 进行批处理, 使用 Flink 进行实时流处理, 通过 Hive 进行交互式查询, 使用 Zookeeper 进行协调和配置管理, 使用 Ambari 管理 Hadoop 集群, 同时使用 Kafka 来处理实时数据流。本系统也将采用这种方式去搭建湖仓一体, 流批一体架构。

大数据技术在基于混合云的分布式制造运营管理系统中发挥着重要的作用。在制造运营管理过程中, 涉及到大量的传感器数据、设备数据和生产过程数据, 传统的关系型数据库系统即使使用了分库分表技术也无法管理如此大规模的数据量。但是使用大数据技术可以提供分布式的数据存储和处理能力, 满足企业对于海量数据的需求。在生产过程中产生的数据通常包含着丰富的信息和潜在的规律, 通过对这些数据进行实时分析和挖掘, 可以获取有价值的洞察和决策支持。通过对供应链数据和市场数据进行整合和分析, 可以实现供应链的实时优化和需求的精准预测。大数据技术可以帮助企业识别供应链中的风险, 通过对市场数据的分析和模型建立, 企业可以更准确地预测产品的需求和销售趋势, 从而进行合理的生产规划和供应链库存管理。

综上所述, 大数据技术在基于混合云的分布式制造运营管理系统中具有重要的应用价值。大数据技术可以帮助企业收集和存储海量的生产数据, 通过进一步的数据分析和挖掘, 应用于供应链优化和需求预测。。

6.1.4 系统运维技术

本文在后端开发过程中基于 Python、Ansible、Django 和 Kubernetes 进行自动化运维平台开发, 以下简要地解释一下它们的作用和关系。

Python: Python 是一种高级编程语言，由于其“胶水语言”的特性，在诸多领域都被广泛使用。Python 可以用于编写脚本、自动化任务、开发 Web 应用，完美契合了运维平台开发的需求。Python 的生态环境较为活跃，多年占据热门编程语言榜首，丰富的第三方库也让运维开发变得更加高效和便捷。

Ansible: Ansible 是一个开源的配置管理、应用部署和任务执行系统。采用 SSH 进行主机之间的通信，用户无需在目标服务器上安装代理，支持推模式和拉模式。Ansible 使用 YAML 格式的剧本（playbooks）来描述配置和任务，能够高效地自动化管理主机资源，被广泛用于各种自动化 IT 基础设施的配置、部署和维护。

Django: Django 是一个 Python 开发的开源的 Web 应用开发框架，用于快速开发高质量的 Web 应用。知名的 Instagram 平台后端用的就是 Django 框架。Django 具有强大的 ORM（对象关系映射）系统、内置的认证和授权机制、URL 路由等功能，帮助开发者能够更加专注于业务逻辑的实现，而不必过多关注底层实现细节。

Kubernetes (K8s): k8s 是一个可移植、可扩展的开源平台，用于管理容器化的工作负载和服务，可促进声明式配置和自动化。K8s 拥有一个庞大且快速增长的生态，其服务、支持和工具的使用范围相当广泛。K8s 提供了强大的容器编排和管理能力，运维工程师使用 K8s 可以很便捷地管理大规模的容器化应用，实现高可用性、弹性伸缩、故障恢复等目标。

目前蓬勃发展的各种云平台通常一起使用以上技术，构建稳健、高效的运维平台，例如阿里云、腾讯云、华为云等诸多企业云。通过 Ansible 进行自动化配置和任务编排，使用 K8s 进行容器编排和自动化部署去管理容器，再基于 Django 进行快速开发 Web 应用的框架和工具，使得运维开发人员能够便捷地管理和维护复杂的系统环境。

6.1.5 RAGSys 技术

LangChain、RAG、LLM 是当前搜索增强生成领域的核心技术。

LangChain: LangChain 是一个用于构建 LLM 应用程序的开源框架，简化了开发人员集成 LLM 的工作，通过链式设计和准确的 Prompt 工程，可以显著提高 LLM 的推理能力。LangChain 通过提供一套模块化组件，简化了整个应用生命周期，从数据预处理到信息检索、再到最终调用 LLM 的整个链路。LangChain 还提供了一些现成的链（Chains），快速开发这些特定的高级任务，开发人员也可以构建自定义链以实现具体的业务需求。

RAG: RAG 是一种利用外部知识源来提高生成式人工智能模型准确性和可靠性的搜索增强生成技术。它结合了语言模型和检索模型的优势,通过在生成式模型之前引入检索模块,从检索到的相关文本中提取有效信息,经过 Prompt 工程后,传输给生成模型进行结果生成。这种方法既保留了生成模型的流畅性和灵活性,又利用了检索模型的准确性和相关性,从而提高了生成结果的质量和效率。

LLM: LLM 在近几年蓬勃发展,如 GPT、BERT、LLAMA 等模型。这些模型通过 Zero-shot/Short-shot Learning 技术进行预训练,具有强大的语义理解、推理、计划和生成能力。在 RAG 系统中,LLM 可以用于理解用户需求、生成结果、评估文档相关性等任务,可以显著提高用户体验。

LangChain、LLM 在 RAG 系统的开发中发挥着关键作用。Prompt 工程师开发 RAG 系统,可以改善 LLM 的知识局限于训练集的困境,通过提高 LLM 对用户查询的更准确理解、对相关文档的更有效检索以及对搜索结果的更精准生成,全面提升 LLM 生成结果的质量。

6.2 测试方法简述

系统测试技术是软件开发过程中一个重要环节,旨在验证所开发的系统是否符合需求规格说明书中需求用例,确保系统在各种情况下都可以正常运行。在测试过程中,测试工程师还需要根据需求选择合适的测试技术和工具,并确保测试过程有效性。本文针对功能测试和非功能性测试分别对生产运营管理系统进行测试。

(1) 功能测试:

功能测试主要针对软件的功能性需求进行测试验证,包括用户界面、数据输入和输出、功能操作和业务实现等。测试工程师根据需求文档编写测试用例,并编写测试程序,通过这些测试用例来验证系统是否能够按照预期进行正确运行。功能测试通常需要使用测试管理工具和自动化测试工具来管理测试用例、执行测试程序和生成测试报告,目前已经有较为完备的测试流程和测试平台可以参考。

(2) 非功能性测试:

非功能测试种类众多,如兼容性测试、安全性测试、可靠性测试、易用性测试、可维护性测试等。本文主要选择用性能测试评估制造运营管理系统在不同负载条件下的性能表现,度量指标选取为响应时间、吞吐量和资源利用率。常见的性能测试类型包括负载测试(模拟不同用户并发访问)、压力测试(逐渐增加负载至系统崩溃点)、并发测试(测试系

统同时处理多个请求)。性能测试通常需要使用性能测试工具, 本文主要采用 JMeter 来模拟用户行为、收集性能数据并生成测试报告。

通过这些系统测试技术之间相互配合, 可以准确地评估软件系统的质量、性能和功能, 确保系统可以正确运行。

6.3 系统整体功能实现与测试

在测试基于分布式的制造运营管理系统(MOM)的功能方面, 通过功能测试, 可以验证系统是否能够按照预期执行其各项功能, 并且确保这些功能符合用户的需求和期望。本文将详细描述测试过程, 包括测试范围和测试结果分析。通过这些测试, 评估系统的稳定性、可靠性和性能, 确保系统在实际生产环境中可以正确运行。

功能性测试采用黑盒测试方法, 测试整个系统的实现功能是否正确实现。

本系统通过 Web 浏览器进行访问, 系统测试的软硬件配置如表 6.1 测试环境配置表所示。

表 6.1 测试环境配置表

名称	配置
操作系统	Windows 10
CPU	Intel 酷睿 i7-10875H
硬盘	1 TB SSD
浏览器	Chrome

6.3.1 MOM 系统

(1) 系统首页

混合云下的分布式 MOM 系统结合了区块链、大数据和人工智能技术, 旨在提供一种智能化的企业制造运营管理解决方案。以下是 MOM 系统首页具备的功能, 首页可以展示生产过程的实时状态, 包括各个节点的运行情况、生产工单的堆积情况以及系统资源利用率等信息。系统首页如图 6.2 所示。

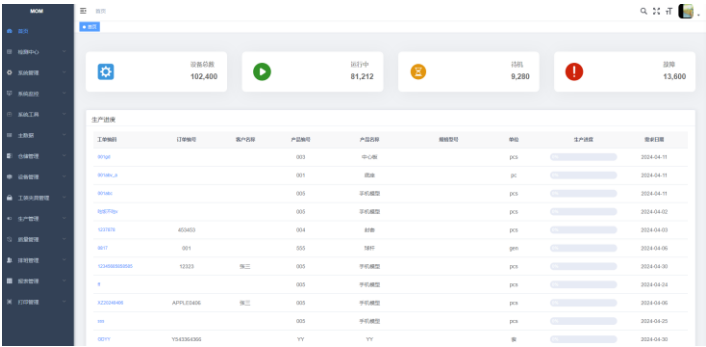


图 6.2 系统首页

系统首页的功能测试主要围绕正常展示测试进行。测试结果如表 6.2 所示。

表 6.2 系统首页功能测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	数据实时性	修改统计数据	自动修改	达到预期
2	数据准确性	修改统计数据	准确修改	达到预期
3	报表可用性	打开首页	正常显示	达到预期
4	跨平台兼容性	切换设备访问	正常显示	达到预期

(2) 用户个人中心

该模块的主要功能是查看用户信息、修改密码和查看与修改用户个人资料，个人中心页面如图 6.3 所示。测试结果如表 6.3 所示。

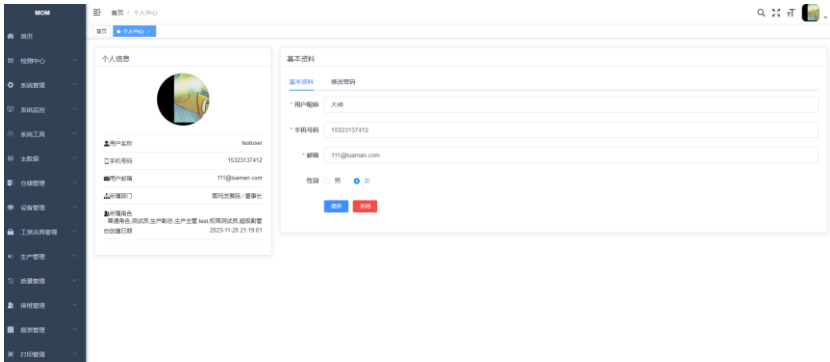


图 6.3 个人中心页面

表 6.3 个人中心测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	页面测试	访问页面	正确显示	达到预期
2	修改测试	修改个人信息	提示修改结果	达到预期
3	修改密码	修改密码	提示修改结果	达到预期

(3) 系统登陆页

测试系统的登录与注册功能、能否正确跳转。系统登录页实现如图 6.4 所示，测试结果如表 6.4 所示。



图 6.4 系统登录页

表 6.4 系统登陆测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	主页测试	输入域名	正常显示	达到预期
2	登陆测试	输入账号密码	验证跳转	达到预期
3	注册登录	输入账号密码	校验跳转	达到预期

(4) 质量管理模块

质量管理模块主要功能是设置与管理质量计划、生成质量数据报表、质量控制项设置。质量管理模块如图 6.5 所示，质量管理测试结果如表 6.5 所示。

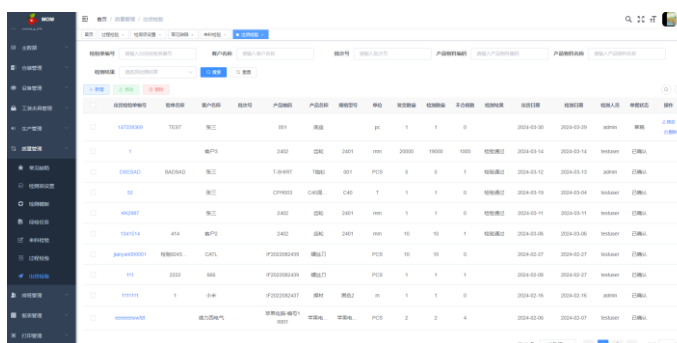


图 6.5 质量管理模块

表 6.5 质管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	质量计划	设置质量计划	正确提示	达到预期
2	质量报表	质量数据	正确展示	达到预期
3	质量项设置	质量项数据	正确提示	达到预期

(5) 设备管理模块

设备管理模块的主要功能是管理设备台账、设备维护管理、设备点检管理、设备备品件管理和统计报表。设备管理模块如图 6.6 所示，设备管理测试结果如表 6.6 所示。



图 6.6 设备管理模块

表 6.6 设备管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	管理设备台账	设备台账信息	正确提示	达到预期
2	设备维护管理	设备维护信息	正确提示	达到预期
3	设备点检管理	设备点检信息	正确提示	达到预期
4	备品件管理	备品件信息	正确提示	达到预期
5	设备统计报表	设备统计信息	正确展示	达到预期

(6) 仓储管理模块

仓储管理模块的主要功能是仓库信息管理、库存管理、出入库管理、可疑品管理和仓储报表。仓储管理模块如图 6.7 所示，仓储管理模块测试结果如表 6.7 所示。

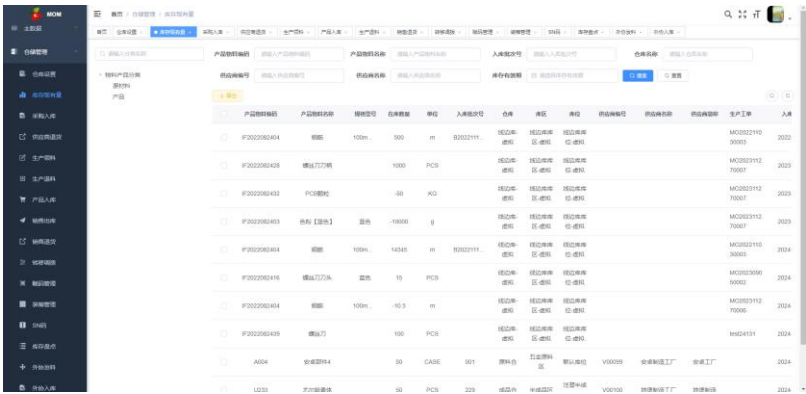


图 6.7 仓储管理模块

表 6.7 仓储管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	存储信息管理	仓储信息	正确提示	达到预期
2	库存管理	库存变动信息	正确提示	达到预期
3	出入库管理	出入库信息	正确提示	达到预期
4	可疑品管理	可疑品信息	正确提示	达到预期
5	仓储报表	仓储相关信息	正确显示	达到预期

(7) 生产管理模块

生产管理模块的主要功能是生产工单管理、工序管理、工艺流程管理、生产报工管理和生产报表。生产管理模块如图 6.8 所示，生产管理模块测试结果如表 6.8 所示。

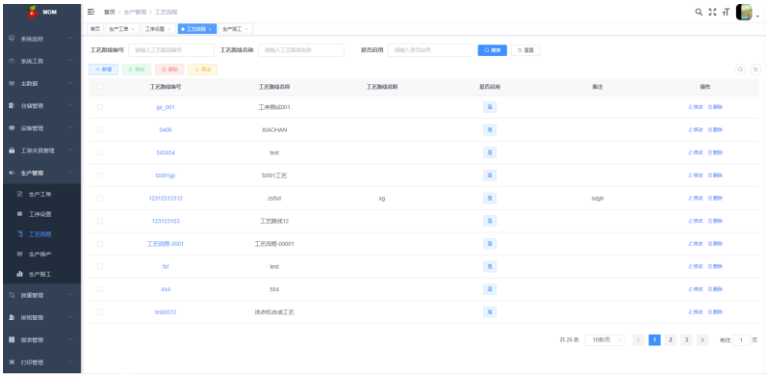


图 6.8 生产管理模块

表 6.8 生产管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	生产工单管理	工单信息	正确提示	达到预期
2	工序管理	工序信息	正确提示	达到预期
3	工艺流程管理	工艺流程信息	正确提示	达到预期
4	生产报工管理	报工信息	正确提示	达到预期
5	生产报表	生产相关信息	正确显示	达到预期

(8) 元数据管理

元数据块的主要功能是产品信息管理、物料信息管理、分类管理、计量单位管理、工厂信息管理、客户信息管理和供应商信息管理。元数据块如图 6.9 所示，元数据块测试结果如表 6.9 示。

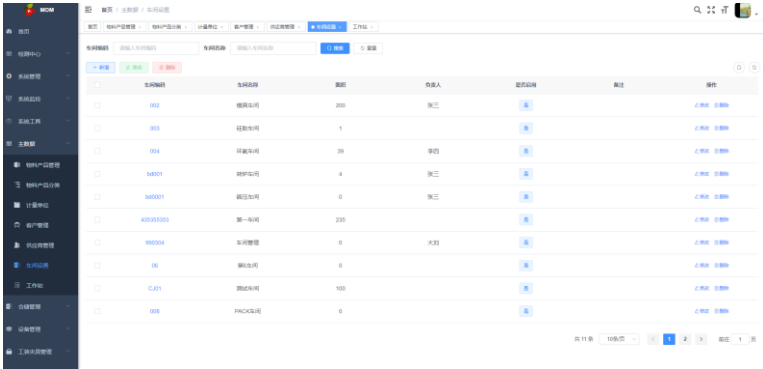


图 6.9 元数据管理模块

表 6.9 元数据管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	产品信息管理	产品信息	正确提示	达到预期
2	物料信息管理	物料信息	正确提示	达到预期
3	分类管理	分类信息	正确提示	达到预期
4	计量单位管理	计量信息	正确提示	达到预期
5	工厂信息管理	工厂信息	正确提示	达到预期
6	客户信息管理	客户信息	正确提示	达到预期
7	供应商信息管理	供应商信息	正确提示	达到预期

6.3.2 自动运维平台

(1) 自动运维平台首页

分布式 MOM 系统的自动运维平台首页主要功能是运维系统的控制主页、展示系统的节点状态和操作日志。自动运维平台的首页实现如图 6.10 所示，自动运维平台功能测试结果如图 6.10 所示。

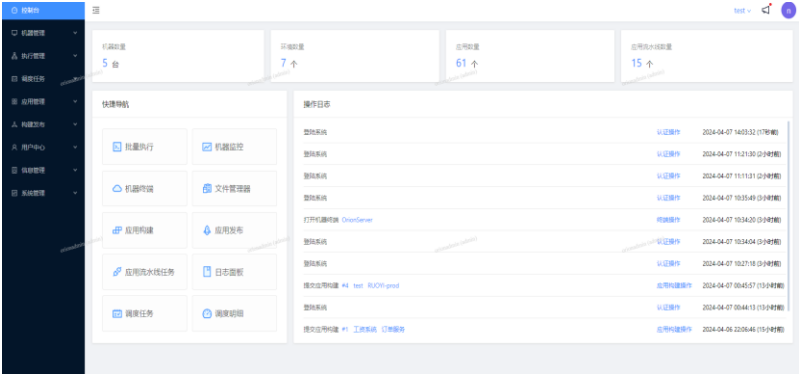


图 6.10 自动化运维平台首页

表 6.10 自动化运维平台首页测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	主页展示	访问主页	正确展示	达到预期
2	节点状态展示	访问主页	正确展示	达到预期
3	操作日志展示	访问主页	正确展示	达到预期

(2) 机器管理

机器管理的主要功能是管理运行系统的机器节点和 Kubernetes 的节点信息、管理机器列表、机器登录密钥、监控机器运行信息、监控机器日志和机器堡垒机代理。机器管理的实现如图 6.11 所示，机器管理测试结果如表 6.11 所示。

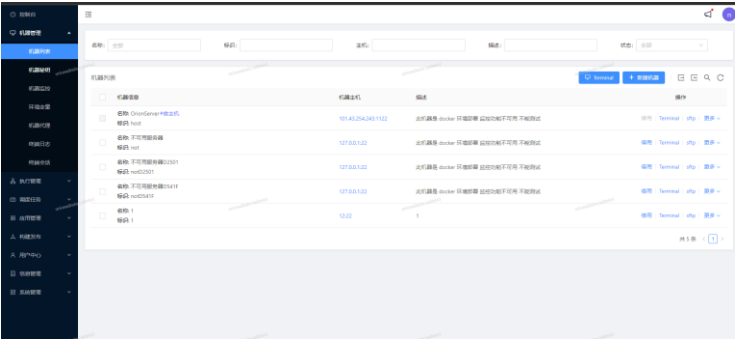


图 6.11 机器管理

表 6.11 机器管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	机器列表	修改机器信息	正确提示	达到预期
2	节点信息	修改节点信息	正确提示	达到预期
3	机器密钥	使用密钥访问机器	正确提示	达到预期
4	机器监控	查看机器监控信息	正确显示	达到预期
5	机器日志	查看机器日志信息	正确显示	达到预期
6	机器代理	通过代理访问机器	正确访问	达到预期

(3) 任务管理

任务管理的主要功能是任务列表、执行记录、任务执行管理、PlayBook 批量上传、日志面板、执行统计和构建发布管理。任务管理实现如图 6.12 所示，任务管理测试结果如表 6.12 所示。

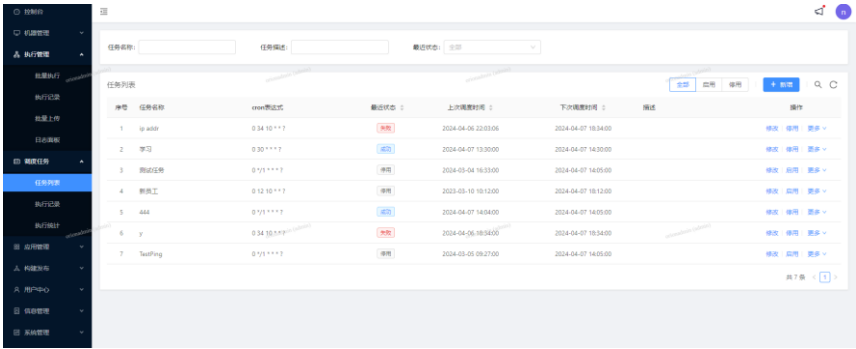


图 6.12 任务管理

表 6.12 任务管理测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	任务列表	修改任务信息	正确提示	达到预期
2	执行记录	查看执行记录	正确显示	达到预期
3	任务执行列表	修改任务计划执行信息	正确提示	达到预期
4	PlayBook 上传	上传本地 PlayBook	正确提示	达到预期
5	日志面板	查看任务日志信息	正确显示	达到预期
6	执行统计	查看任务执行统计	正确显示	达到预期
7	构建发布管理	修改发布管理信息	正确提示	达到预期

6.3.3 大数据平台

(1) 任务管理

任务管理页的主要功能是新建大数据任务、查询任务信息、运行任务和任务监控。任务管理如图 6.13 所示，任务管理测试结果如表 6.13 所示。

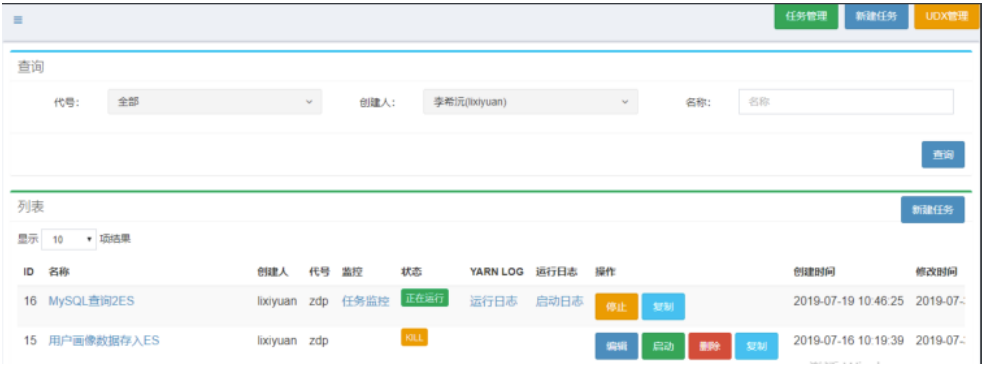


图 6.13 大数据任务列表

表 6.13 任务列表测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	新建任务	任务信息	正确提示	达到预期
2	查询任务信息	查询任务列表	正确显示	达到预期
3	执行任务	选择任务执行	正确提示	达到预期
4	任务监控	查看任务执行信息	正确显示	达到预期

(2) 任务调度

任务调度的主要功能是任务配置、调度配置和更新记录。任务调度如图 6.14 所示，任务调度测试结果如表 6.14 所示。

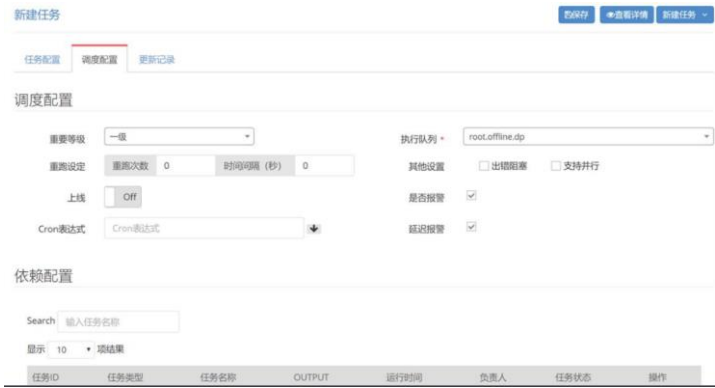


图 6.14 任务调度

表 6.14 任务调度测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	任务配置	任务配置信息	正确提示	达到预期
2	调度配置	调度配置信息	正确提示	达到预期
3	更新记录	访问更新记录页	正确显示	达到预期

6.3.4 知识库系统

(1) 问答页

RAG 系统问答页主要功能是接受用户输入，返回对话结果。问答页如图 6.15 所示，问答页功能测试结果如表 6.15 所示。

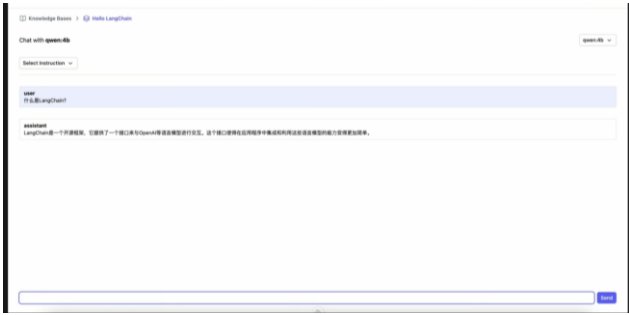


图 6.15 RAG 系统问答页

表 6.15 RAG 习题问答测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	接受用户输入	用户输入	正确显示	达到预期
2	返回对话结果		正确显示	达到预期

(2) 知识库知识输入页

RAG 系统知识输入页主要功能是上传用户知识库数据。知识输入页如图 6.16 所示，知识输入页功能测试结果如表 6.16 所示。

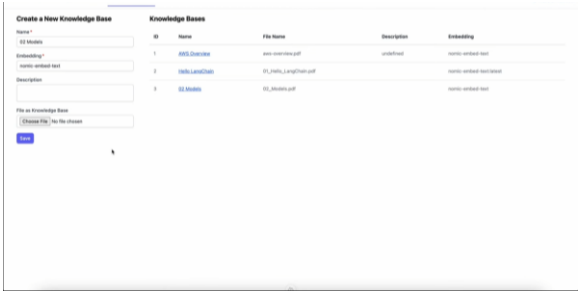


图 6.16 知识输入页

表 6.16 知识输入测试结果

序号	测试场景	输入内容	预期效果	测试结果
1	输入 File 格式	知识库文件	正确提取	达到预期
2	输入文本	知识库文本	正确上传	达到预期
3	输入图片格式	知识库截图	正确识别	达到预期

6.4 系统非功能测试

非功能性测试是针对整个系统的测试，主要是对系统性能、反应时间、数据传输时间等的测试。具体测试如表 6.17 所示。

表 6.17 非功能测试表

序号	测试说明	数据量	要求	结果
1	登录反应时间	正常	小于 100ms	达到预期
2	操作反应时间	正常	小于 300ms	达到预期
3	数据查询时间	正常	小于 300ms	达到预期
4	系统数据交互	复杂	小于 500ms	达到预期
5	文件传输时间	复杂	小于 60s	达到预期
7	RAG 召回准确性	复杂	大于 80%	达到预期

6.5 本章小结

本章首先介绍了系统开发工具和系统测试方法，然后通过黑盒测试法对系统功能进行了相关测试并展示了系统实现，采用白盒测试法对系统的性能进行了测试。测试结果显示，两者都通过测试，系统基本符合要求。

7 总结与展望

7.1 全文总结

本文基于混合云环境下的分布式制造运营管理系统，结合区块链、大数据和人工智能的综合应用，进行了系统化研究与设计。各章节内容围绕着系统的构建和应用展开，从绪论到总结，系统地探讨了该系统在企业应用集成、大数据分析、智能运维和检索增强生产系统等方面的重要性和应用前景。

(1) 本文分析了混合云环境下 MOM 系统的背景和意义，以及区块链、大数据和人工智能在其中的作用和价值。系统地回顾了现有的相关技术和研究成果，为后续的系统设计和实现提供了理论基础和参考。分析阐述了该系统的功能需求和性能需求，包括数字化、智能化和自动化等方面的要求。

(2) 采用 TOGAF 理论进行了系统架构、模块设计和技术实现，以及区块链、大数据和人工智能在 MOM 系统中的应用方式。综合运用多种技术，开发实现 MOM 系统。并通过白盒与黑盒的测试方法，对系统的功能性测试和非功能性测试，验证了系统设计的有效性和可行性。

(3) 探讨了 RAG 系统召回准确性问题中可能存在的问题和挑战，并提出了基于 Chunk-Tree 的解决方案。

7.2 展望

但由于个人知识水平、学术水平和时间有限，本系统还存在一定的不足，研究内容还有待进一步改善：

(1) 系统和人工智能结合仅采用了 RAG 打造企业知识库系统，对于 LLMs 的推理有待进一步挖掘，融入到企业决策中，例如：通过 Agent 帮助员工进行规划工作计划、通过 Code Generation 辅助员工进行工作等。

(2) 系统仅仅实现了一部分的功能 Demo，对于应对复杂的企业真实生产场景还有待进一步的持续开发和迭代，提示系统性能，完善系统功能。

(3) 本系统为制造业融合人工智能等领域技术进行了探索，未来可以进一步探索区块链、大数据和人工智能等前沿技术在其他领域的融合应用，如金融、医疗、物联网等，以解决实际应用中的复杂问题，并推动跨领域的技术创新和发展。

参考文献

- [1]肖土盛,孙瑞琦,袁淳,等.企业数字化转型、人力资本结构调整与劳动收入份额[J].管理世界,2022,38(12):220-237.DOI:10.19744/j.cnki.11-1235/f.2022.0174.
- [2]孙伟增,毛宁,兰峰,等.政策赋能、数字生态与企业数字化转型——基于国家大数据综合试验区的准自然实验[J].中国工业经济,2023,(09):117-135.
- [3]张隽昊.X公司供应链数字化建设存在的问题及优化对策研究[D].河南大学,2022.
- [4]巫强,姚雨秀.企业数字化转型与供应链配置:集中化还是多元化[J].中国工业经济,2023,(08):99-117.DOI:10.19581/j.cnki.ciejournal.2023.08.005.
- [5]张晓暄.大数据视角下公司成本信息化管理模式研究[D].内蒙古大学,2022.
- [6]沈仲华.复杂场景下基于区块链的分布式溯源架构设计与仿真研究[D].浙江工商大学,2023.
- [7]吴晋.一汽大众长春公司数字化转型战略研究[D].吉林大学,2023.
- [8]景文治.人工智能赋能下关键核心技术突破与制造业智能化升级实证研究[D].辽宁大学,2021.
- [9]周游,钱科军,景栋盛.加快能源工业互联网融合发展,助力企业数字化转型[C]//中国电力企业管理创新实践(2021年).国网江苏省电力有限公司苏州供电分公司;,2023:3.
- [10]陈姿芊,牛科迪,姚中原,等.适用于物联网的区块链轻量化技术综述[J/OL].计算机应用,1-13[2024-04-22].
- [11]吕广美,胡祥培,吉清凯.零售竞争下农产品供应链的定价与区块链采用博弈模型[J/OL].工程管理科技前沿,1-10[2024-04-22].
- [12]邹宇鹏,朱健,张问银.放疗联盟链——基于区块链的放疗数据跨平台安全共享方案[J].中华肿瘤防治杂志,2024,31(06):325-332.
- [13]张士双,葛英勇,樊婷,等.物联网和工业云在选矿过程监控中的应用研究[J].自动化技术与应用,2024,43(03):1-4+26.
- [14]宋坤,刘吉宁.5G赋能工业互联网的应用研究与实践[J].电信工程技术与标准化,2024,37(03):75-82+87.
- [15]孔祥君,高云鹏,吕海洋.离散制造业 MOM 工业软件的发展难点和趋势[J].新型工业化,2023,13(09):53-60.
- [16]刘赞.如何有效构建工业软件 MOM(制造运营管理)平台?[C]//中国通信学会工业互联网委员会.2022 工业互联网学术大会论文集.新华三集团工业互联网产业研究院;,2023:5.
- [17]李金财,李彦德,刘宝新,等.煤矿企业生产中安全监控信息管理系统的设计与实现[J].科技创新,2023,(10):97-100.
- [18]邓小芳,高锐.机械自动化技术在汽车生产中的应用研究[J].内燃机与配件,2024,(06):112-114.
- [19]姚新.基于 MES 系统的海上平台机械发电机的管理方法探究[J].石化技术,2024,31(03):246-248.
- [20]陈志平,汪传亮,黄超亮,等.铜盘管企业 MES 的构建与实现[J].制造业自动化,2024,46(02):209-213+220.
- [21]陈亮,王孝春.MES 系统在企业数据化转型中的应用分析[J].数字技术与应用,2024,42(01):18-20.
- [22]徐健.基于微服务和中台技术架构的移动警务生态系统建设研究[J].科技与创新,2024,(07):102-104+107.
- [23]Corrective Retrieval Augmented Generation Shi-Qi Yan1, Jia-Chen Gu2, Yun Zhu, Zhen-Hua Ling.arXiv: arXiv:2401.15884 [cs.CL].<https://arxiv.org/abs/2401.15884>.

- [24]Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela.arXiv:2005.11401 [cs.CL].<https://arxiv.org/abs/2005.11401>.
- [25]Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, Hannaneh Hajishirzi.arXiv:2310.11511 [cs.CL].<https://arxiv.org/abs/2310.11511>.

附录 攻读学位期间取得的成果

本科期间获奖情况：

河南省大学生机器人竞赛省赛一等奖；亚太数模三等奖；蓝桥杯河南省赛一等奖，国赛二等奖；机械创新设计大赛河南省赛三等奖；高教杯大学生先进成图技术与产品信息建模创新大赛省团体特等等奖，省个人机械类一等奖，国赛个人三等奖，国赛团体三等奖，国赛团体 3D 打印类二等奖。软著一份。

本科期间考取职称资格：

软件设计师（中级职称）。

致 谢

大学时光荏苒，这段旅程即将结束，一路走来，在大家的帮助下，我不仅在学习方面取得显著的进步，更在个人综合素养上提升颇多。在大学这个充满活力与挑战的环境中，培养了解决问题的能力，也锻炼了团队协作能力和领导力。

首先，我要衷心感谢我的导师边军老师。边老师待人和善，为人谦虚，学术态度严谨，每当我遇到难题时，边老师总能以极具启发性的思想耐心讲解问题的思考方向和解决方法。君子之学也，说义必称师以论道，听从必尽力以光明。感谢边老师在我整个论文写作过程中的悉心指导和鼓励，使得我树立了正确的学术理念，构建了严谨的学术思维。

其次是感谢大学生涯对我悉心指导的授课老师们，你们辛苦了。学者必有师，所以通其业、成就其道德者也。其中，我由衷的感谢杨俊红老师，在我初入大学时便为我指引方向，在我大学求学生涯中，指导我颇多，帮我解决诸多问题。同时也很感谢赵建国老师和马伟伟老师在工程制图方面对我的指导与培养，赵老师和马老师上课风格幽默风趣，讲述的知识浅显易懂，不仅授予我知识，还教授我很多人生道理。也很感谢辅导员老师们在学习之外的方面对我的诸多帮助。

感谢我的同学和朋友们，这段记忆将成为我宝贵的财富。感谢学长学姐们的关怀与帮助，感谢与我同届的同学们的互帮互助，感谢其他届同学在学习生活中的陪伴。这些经历将成为我们人生旅途中宝贵的财富，激励我们在未来的道路上不断前行，勇于探索，追求卓越。

感谢朋友家人在背后的默默支持和鼓励，在我遇到困难与挫折时，关怀我帮助我，给予了我精神上的鼓励。同时，我要感谢自己。学之广在于不信，不倦在于固志。感谢自己在学业路上的坚持和努力，让我不断进步和成长。学问有未惬适处，只求诸心，若反身而诚，清通和乐之象见，即是自得处，更望勉力，以此而已。这段求学之旅将成为我人生中宝贵的经历，我将继续努力，追求更高的目标。

最后感谢论文评审以及答辩过程中的各位评审以及专家老师对我毕业论文的指导与建议。

马庆奥

2024 年 05 月于中原科技学院