

基于区块链的食品溯源系统

摘要 本篇论文旨在探究基于区块链技术的食品溯源系统，该系统使用区块链技术实现供应链中各个参与方的溯源数据的可靠记录和共享，以提高供应链管理的透明度和效率。具体而言，本论文分析了现有的供应链溯源技术和区块链技术，并结合供应链溯源系统的实际需求，提出了一种基于区块链的供应链溯源系统的设计与实现方案，并通过实验验证了该方案的可行性和有效性。

本文的主要贡献在于提出了一种新的基于区块链技术的供应链溯源系统的设计方案，并通过实验验证了该方案的可行性和有效性。与传统的供应链溯源技术相比，本系统具有更高的安全性和可靠性，可以有效保护供应链中各参与方的数据隐私和安全，同时提高了供应链管理的效率和透明度。

关键词 区块链技术；食品溯源；数据共享；供应链管理；

ABSTRACT This paper aims to explore a blockchain-based food traceability system that uses blockchain technology to reliably record and share traceability data for each participant in the supply chain to improve transparency and efficiency of supply chain management. Specifically, this paper analyses existing supply chain traceability technologies and blockchain technology, and proposes a design and implementation plan for a blockchain-based supply chain traceability system based on the actual needs of supply chain traceability, and validates the feasibility and effectiveness of this plan through experiments.

The main contribution of this paper is to propose a new design solution for a blockchain-based supply chain traceability system, and to validate the feasibility and effectiveness of this solution through experiments. Compared with traditional supply chain traceability technologies, this system has higher security and reliability, can effectively protect the data privacy and security of each participant in the supply chain, and improves the efficiency and transparency of supply chain management.

KEYWORDS blockchain technology; food traceability; data sharing; supply chain management

目录

1 前 言	1
1.1 选题背景与研究意义	1
1.2 与选题相关的国内外研究现状	1
1.3 主要研究内容	2
1.3.1 食品溯源系统的设计和实现	2
1.3.2 食品溯源系统的应用场景分析	2
1.3.3 食品溯源系统的数据标准化与合规性	2
1.4 本文组织结构	3
2 关键技术介绍	4
2.1 GS1 标准	4
2.2 区块链技术	5
2.3 Rust 语言	6
2.4 Substrate 框架	7
3 存储结构设计	10
3.1 简述	10
3.2 食品持有项	10
3.3 溯源痕迹链	12
3.4 食品原料	13
3.5 账户到 GLN 映射	15
3.6 GLN 到生产许可证映射	15
4 交易方法实现	17
4.1 简述	17
4.2 生产食品	17
4.3 交易食品	19
4.4 存储食品 & 运输食品	20
4.5 设置账户到 GLN 映射 & 设置 GLN 到生产许可证映射	22

5 RPC 溯源查询方法实现.....	25
6 系统前端设计	27
6.1 简述.....	27
6.2 生产时追加原料 交互界面展示.....	28
6.3 选择参与方位置 交互界面展示.....	29
6.4 简单输入检查 交互界面展示.....	31
6.5 测试生成溯源码.....	31
7 结论	34
参考文献	36
致谢	38

1 前言

1.1 选题背景与研究意义

随着全球化的不断推进和国际贸易的日益频繁，食品安全问题越来越受到人们的关注。食品安全问题不仅影响着人们的身体健康，还直接关系到国家和地区的经济利益和国际形象。而互联网技术的发展，使得越来越多的消费者开始关注食品的安全问题，并且要求了解食品的来源和生产过程，以便更好地保护自己的健康^[3]。

在食品安全问题的解决过程中，食品溯源技术被广泛应用。食品溯源技术是指通过对食品生产、流通和销售等环节信息的记录和追踪，实现对食品的溯源和追溯。食品溯源技术可以提高食品行业的透明度和可信度，增强消费者的信任和满意度。但是，传统的食品溯源技术存在一些问题，例如信息不对称、数据造假、易被篡改等等，这些问题导致了食品溯源的可靠性和有效性存在瓶颈^[4]。

为了解决这些问题，近年来，区块链技术被广泛应用于食品行业的溯源系统中。区块链技术的分布式、去中心化、不可篡改等特点，使得其成为食品溯源技术的一种有效解决方案。然而目前已有的区块链溯源系统还存在效率低下、扩展性差等问题，限制了系统的实用性和推广应用。因此，有必要研究开发一种新型的基于区块链技术的食品溯源系统作为研究对象，旨在解决现有系统存在的问题，并提高食品行业的数字化水平。

1.2 与选题相关的国内外研究现状

当前，国内外许多研究人员和企业已经开始研究使用区块链技术来建立食品溯源系统。国外研究主要集中在美国、欧洲和澳大利亚等地，这些国家和地区已经开始大规模使用区块链技术来管理食品供应链。例如，美国的 Walmart 公司已经与 IBM 公司合作使用区块链技术来建立自己的食品溯源系统^[2]。欧洲的一些农产品公司也开始使用区块链技术来跟踪和记录产品的生产过程和交易流程。澳大利亚也在不断探索如何使用区块链技术来解决食品溯源和食品安全问题。

国内方面，许多企业和研究机构也在积极探索使用区块链技术来建立食品溯源系统。例如，阿里巴巴集团的菜鸟网络公司已经推出了基于区块链技术的“菜鸟智慧物流溯源平台”，该平台可以确保食品的生产、流通和销售环节的透明性和可追溯性^[15]。另外，一些食品企业也开始尝试使用区块链技术来改进其自身的食品溯源系统。随着区块链技术的不断发展和完善，越来越多的企业和研究机构开始将其应用于食品行业中。

然而，目前国内外使用区块链技术建立食品溯源系统的研究还处于探索阶段，仍存在一些问题和挑战，例如如何保证溯源过程中记录的信息的真实性和准确性、如何确保食品生产链条中商业机密不泄露等等。因此，需要进一步深入研究和探索，以确保区块链技术在食品行业中的有效应用和推广。

1.3 主要研究内容

1.3.1 食品溯源系统的设计和实现

本研究将设计并实现一套基于区块链技术的食品溯源系统，该系统将区块链技术应用于食品生产、加工、运输、销售等环节的数据追溯。在设计该系统时，我们将考虑到系统的安全性、稳定性、可扩展性、可操作性等方面。同时，我们将结合实际需求和应用场景，制定系统的功能模块和交互流程，确保系统的用户体验和易用性。

1.3.2 食品溯源系统的应用场景分析

本研究将分析食品溯源系统在不同应用场景下的可行性和效益，具体应用场景包括食品生产、加工、存储、运输、销售等环节。在系统的数据记录和食品安全溯源等方面，我们将对系统在不同场景下的操作流程、数据处理方式等方面进行分析和评估，以确定系统的适用性和优势。

1.3.3 食品溯源系统的数据标准化与合规性

在食品行业，数据标准化和合规性是非常重要的。本研究将研究现有相关的数据标准和合规要求，对这些标准与法规进行适配，以确保系统的数据质量和合规性。我们将深入研究相关的法律法规和标准，将系统的数据格式和数据共享方式进行规范化和标准化，并考虑如何遵循相关的法律法规和标准要求，以保证系统的合规性。

综上所述，本研究的主要内容将围绕设计和实现基于区块链技术的食品溯源系统展开，以及对该系统的技术实现、应用场景、数据标准化和合规性等方面进行深入研究和分析。这些研究内容将有助于推动食品行业的数字化转型和升级，为食品安全和溯源提供更可靠、安全、高效的解决方案。

1.4 本文组织结构

本文的组织结构如下：

第 1 章 前言，主要阐述选题背景与研究意义，以及与选题相关的国内外研究现状，并简要概括了本文的主要工作。

第 2 章 关键技术介绍，详细介绍了本研究所遵循的规章标准、涉及的关键技术概念、使用的开发框架等信息。

第 3 章 存储结构设计，介绍了本系统存储结构的设计思路，并就设计时的具体问题探讨。

第 4 章 交易方法实现，涵盖交易函数 API 设计、数据校验算法的具体流程等。

第 5 章 RPC 溯源查询方法实现，介绍了溯源查询功能的具体流程、前后端交互等细节。

第 6 章 系统前端设计，展示了本系统的前端界面，对溯源二维码的生成等。

第 7 章 结论，对上述工作进行总结分析。

2 关键技术介绍

2.1 GS1 标准

GS1 是一个全球性的非营利性组织，致力于制定和推广各种标准，以实现商品和信息在全球范围内的有效和准确交换。GS1 标准包括条码标识、数据交换和业务流程等方面的标准。在全球范围内，GS1 标准已被广泛应用于各个行业的产品追溯和管理中。其中，GS1 全球追溯标准（GS1 Traceability Standard）是一种基于条码技术的全球性追溯标准，它可以实现产品的源头追溯和流向追溯，旨在帮助生产商、供应商、零售商等各参与方更好地保障消费者的健康安全和权益。该标准提供了一种统一的全球解决方案，使得各国在产品质量安全方面的追溯标准达到协同一致，从而实现全球化生产、流通和消费的目标。

GS1 全球追溯标准使用一系列的标识码，通过互联网技术，使得追溯的信息可以通过全球网络实时地共享。GS1 标准是全球最通用的商品标识码，该标准已经被全球超过 200 个国家和地区的超过 150 万家企业所采用。GS1 标准能够为全球供应链提供基础的标准化支持，实现生产、物流、销售等环节的协同优化。

GS1 全球追溯标准的主要目的是确保消费者的权益和安全。标准要求生产商、供应商等各方在产品生产、流通和销售等环节，必须对其产品进行标识和追溯。标识包括产品标识码、批次号、生产日期、过期日期等。这些信息通过互联网技术，能够在全球范围内进行追溯，使得消费者能够得到准确的产品信息和相关的质量安全信息，从而对产品进行有效的风险评估和消费决策。

实现 GS1 全球追溯标准需具备如下四个核心要素：

（1）产品编码：产品编码是实现 GS1 全球追溯标准的第一步，它是将产品与标准化的信息流连接起来的关键。GS1 全球追溯标准使用的产品编码是全球唯一的、可追溯的、标准化的 GTIN（全球商品交易编号）。GTIN 是由 GS1 组织分配的，它是一个由 14 位数字组成的编码，可以表示产品的类型、品牌、型号和包装规格等信息。

（2）数据标准化：GS1 全球追溯标准要求对产品信息进行标准化，使得不同环节的信息可以在全球范围内通用。标准化的数据格式包括产品编码、批次号、生产日期、过期日期、产地、生产企业、物流信息等等。数据标准化可以使得整个供应链的信息流畅通无阻，可以追踪和记录每个环节的信息，从而保证产品的安全性和质量。

(3) 信息共享：GS1 全球追溯标准要求各个环节的信息共享，即供应链中各个环节之间应该交换产品的信息。通过信息共享，企业可以更好地了解供应链中的情况，包括生产、加工、存储、运输等环节的信息。这可以帮助企业更好地管理和控制整个供应链，提高效率和减少风险。

(4) 技术支持：GS1 全球追溯标准需要借助先进的信息技术，包括互联网技术、数据分析技术等，以实现全球范围内的信息共享和数据管理。此外，GS1 全球追溯标准还需要提供一系列的技术支持和培训服务，以帮助企业更好地实施和应用该标准，提高企业的管理效率，降低成本，提高竞争力。

总之，GS1 全球追溯标准是一套全球性质的物流追溯系统，通过使用统一的标识码和数据标准化要求，使得各国在产品质量安全方面的追溯标准能够达到协同一致。该标准的实施将有助于提高产品质量和安全，保护消费者权益，促进全球贸易，提高企业效率和竞争力。

2.2 区块链技术

区块链技术是一种去中心化的分布式数据库技术，其基本原理是将数据分散存储在网络中的多个节点上，每个节点都可以验证和记录数据，而且没有单一的控制中心^[1]。区块链技术的本质是一个不可篡改的、去中心化的、可信的交易账本数据库，每一个账本都由若干个区块组成，每个区块包含前一个区块的哈希值和当前的交易信息，形成一个链式结构。区块链技术的特点在于，数据被存储于多个独立自主的节点中，任何单独的节点都不能单独修改数据。只有经过共识算法的节点才能将新的数据添加到数据库，这样可以确保数据的安全性和可靠性，使得区块链技术成为一种非常适合用于溯源系统的技术。

区块链技术的主要特点如下：

- **不可篡改性：**区块链中的每一笔交易都被记录在一个块中，并且每个块都有一个唯一的哈希值，前一个块的哈希值会成为下一个块的一部分，这种链式结构使得区块链的数据不能被篡改或删除，保证了数据的完整性和可靠性。

- **去中心化：**区块链没有中央控制机构，每个节点都有相同的权利和责任，数据的管理和验证由网络中的节点共同完成，避免了单一中心化机构的风险，提高了系统的稳

定性和安全性。

- **透明性:** 区块链中的所有交易信息都是公开可查的,任何人都可以查看账本上的所有交易信息,确保了数据的公开和透明。

- **匿名性:** 在区块链中进行交易时,不需要披露个人身份信息,交易双方可以保持匿名,同时也可以防止交易数据被盗用或篡改。

在食品溯源领域中,区块链技术的应用主要是利用其不可篡改、去中心化、透明等特点,实现对物流、交易、商品流等方面的追溯和监控,提高供应链的效率和安全性^[7]。食品溯源系统作为一种保障食品安全的技术手段,需要追踪记录食品从生产到消费的全过程,及时掌握食品的流向和质量状况。将区块链技术应用于食品溯源系统,不仅可以提高溯源系统的可靠性和透明度,还能够降低成本、提高效率,增强监管能力,为消费者提供更加安全和可信的食品购买体验^[10]。

2.3 Rust 语言

Rust 语言是一种由 Mozilla 开发的一种现代的、系统级别的编程语言。它于 2010 年首次亮相,并于 2015 年发布了首个稳定版本。Rust 被设计为一种内存安全、并发安全和高性能的语言,具有许多先进的语言特性,如所有权和借用系统、模式匹配、泛型编程、函数式编程等。Rust 这些特性使得 Rust 可以在保障内存安全的前提下,实现高效的系统级别编程,并在不损失性能的情况下提供更高的安全性和可靠性。

Rust 语言具有许多优点,最突出优点是高性能、内存安全、并发安全。Rust 语言的高性能主要体现在 Rust 语言的编译器能够生成高效的本机代码,其性能与 C 和 C++ 等老牌系统编程语言不相上下。而 Rust 语言在功能上还能做到更多,通过引入所有权和借用的概念,防止了内存泄漏、使用非法内存和数据竞争等问题。Rust 的类型系统和所有权概念还能够帮助开发人员避免并发问题,如数据竞争和死锁。除此之外,Rust 还提供了一组强大的并发原语,如通道、锁等,使得编写并发代码更加安全、简单。这些概念也使得 Rust 代码更易于分析和调试。

Rust 语言很适合用于开发食品溯源系统,具有如下几个方面的优势:

- **安全性:** 食品溯源系统需要处理大量的敏感数据,如食品的来源、运输、存储等信息。使用 Rust 语言编写食品溯源系统可以保证其内存和并发安全性,避免数据泄露

和其他安全漏洞。

- **可靠性:** 食品溯源系统需要在大规模的数据量和复杂的供应链网络中运行。使用 Rust 语言编写的系统可以保证其可靠性和鲁棒性，避免系统崩溃和其他故障。

- **跨平台性:** 使用 Rust 语言编写的程序可以在不同操作系统和硬件平台上运行，这对于一个需要跨越不同国家和地区的供应链网络的食品溯源系统来说是非常重要的。此外，Rust 语言程序还具有较小的二进制文件大小和较少的依赖，支持 WebAssembly 技术，这可以使得开发人员将应用程序打包成为 Web 应用，从而实现跨平台和跨设备的运行。

- **可维护性:** 食品溯源系统需要经常更新和维护，以保持其安全和可维护性。Rust 的类型系统和所有权概念使得代码更加易于维护。通过强制执行所有权规则，Rust 可以在编译时捕获一些错误，例如空指针引用和悬空引用。这些错误在运行时会导致系统崩溃，但在编译时就能够发现和修复。

总之，Rust 语言是一种内存安全、并发安全、高性能和可维护性强的语言，非常适合开发需要处理大规模数据、并且要求高安全性和可靠性的应用程序。在食品溯源系统中，安全性和可靠性是至关重要的因素。使用 Rust 语言编写食品溯源系统可以保证其内存安全、并发安全和高性能，从而确保食品的安全和质量，减少食品相关问题的发生。

2.4 Substrate 框架

Substrate 是基于 Rust 语言构建的开源的、泛用的区块链框架，提供了一系列功能强劲的手脚手架代码，能够供开发者快速构建任意特定领域的区块链应用。使用 Substrate，您可以完全创造性地控制要构建的应用程序：Substrate 框架采用模块化的设计思想，支持动态添加和删除模块，从而可以方便地进行系统扩展和升级。开发者可以从大型开源模块和模板库中选择预定义的应用程序逻辑进行功能组合，极大加快了区块链系统的开发时间。在这一系列功能丰富的模块中，最重要的模块是 Substrate Runtime，它是一个高度灵活和可扩展的运行环境，允许开发人员自定义区块链的逻辑和行为。Substrate Runtime 内置了一些通用的可插拔组件，例如账户管理、交易处理、安全性、共识算法、治理机制等等，同时也提供了许多扩展功能的接口，以方便开发人员根据自己的需求进行定制。除了 Substrate Runtime 之外，Substrate 框架还包含了一些其他的核心组件，例如共识算法、网络架构、链下存储、交易池和 RPC 等等。这些组件提供了区块链系统

必须的基本功能，例如共识算法的实现、节点间通信的管理、数据存储和管理、交易的处理和验证、以及外部应用和区块链节点之间的通信。

Substrate 框架的一个重要特性是可升级性。**Substrate** 框架支持链上升级和链下升级两种方式，可以方便地进行系统更新和功能扩展，而不需要中断现有的应用。链上升级是通过在区块链上运行升级合约来实现的，这种方式可以保持链的运行状态不变，同时也可以保证升级的安全性和可靠性。链下升级则是通过替换区块链的运行环境来实现的，这种方式可以更灵活地进行系统升级和扩展，实现系统的功能迭代。

Substrate 框架的另一个重要特性是可互操作性。**Substrate** 框架支持与其他区块链系统和应用的跨链互操作，包括与其他基于 **Substrate** 的区块链、**Polkadot**、以太坊和比特币等其他区块链系统的互操作。这种互操作性是通过 **Substrate** 框架内置的 **Polkadot** 插件实现的。**Polkadot** 插件是一个用于连接不同区块链系统的桥梁，可以使不同区块链系统之间进行跨链交互和数据传输。这种互操作性使得开发人员可以更方便地将现有的应用和服务集成到区块链系统中，并可以更灵活地构建多链互联的区块链生态系统。本项目是基于许可链的区块链项目，目前不涉及需要多链互通、跨链互操作等功能，但 **Substrate** 框架的这一特性为未来系统功能的扩展提供了无限可能。此外，**Substrate** 框架采用标准化的 **RPC** 接口和协议，并支持由用户自定义 **RPC** 接口，可以方便地与其他区块链系统和应用进行通信互操作。

另外，**Substrate** 框架还具有良好的安全性和可靠性。**Substrate** 框架的核心组件都是经过严格的安全测试和审查的，并采用了一系列的安全机制和防御措施，以保证区块链系统的安全性和稳定性。例如，**Substrate Runtime** 采用了沙盒技术和 **WASM** 虚拟机等机制，可以有效地隔离恶意代码和错误，同时也可以提高代码的性能和可移植性。此外，**Substrate** 框架还提供了一系列的开发工具和资源，例如 **Rust** 编程语言、**Web3.js** 库、**Polkadot.js API** 等等，可以帮助开发人员更好地编写安全、可靠和高效的区块链应用。

Substrate 框架还提供了许多高级功能，例如 **Substrate Front-end Template**、**Substrate UI**、**Substrate Developer Hub** 等等。这些工具和库可以帮助开发人员更快速地构建、部署和管理区块链应用。例如，**Substrate Front-end Template** 是一个基于 **React** 框架构建的前端模板，可以帮助开发人员更快速地创建区块链应用的前端页面，简化前端界面开发难度；**Substrate UI** 是一个基于 **Polkadot UI** 框架构建的可视化界面，可以帮助开发人员更方便地管理和监控区块链节点的运行状态，可以更方便的与区块链后端进行交互；**Substrate Developer Hub** 是一个全面的文档和资源库，提供了大量的开发教程、示例代码、**API** 文档和其他有用的工具和资源，降低了开发门槛和学习曲线，帮助更多的开发

人员和企业快速进入区块链领域。

综上所述，**Substrate** 区块链框架是一个灵活、可扩展和安全的区块链开发框架。它提供了丰富的组件和工具，可以帮助开发人员更快速、更轻松地构建、部署和管理区块链应用。**Substrate** 框架还具有良好的可升级性和可互操作性，可以方便地进行系统升级和扩展，同时也可以与其他区块链系统进行跨链交互和数据传输。**Substrate** 框架已经得到了广泛的应用和验证，可以为企业、开发者和用户提供更高效、更安全和更可信的区块链服务和应用。未来，**Substrate** 框架还将继续推动区块链技术的发展和 innovation，成为区块链领域的核心技术基础之一。

3 存储结构设计

3.1 简述

本系统采用区块链技术实现，使用了分布式全冗余存储设计，使每个节点都能够存储完整的数据库。该设计方案的优点在于可以增强系统的数据健壮性，有效地抵御恶意节点的攻击行为，如拒绝服务攻击和双花攻击等。系统的存储结构设计包含五个主要部分：食品持有项、溯源痕迹链、食品原料、账户到 GLN 映射和 GLN 到生产许可证映射。前两个部分是系统中与食品相关操作的重点结构，后三个部分则主要用于验证权限和简化数据输入。每个存储结构均采用哈希表实现，包括单键哈希表和多键哈希表。下面将逐一介绍这五个部分的存储结构及其详细内容。

3.2 食品持有项

食品持有项是对同一批次食品的抽象，这里的“同一批次”是指具有相同溯源痕迹链的食品。由于这些食品所经历的加工、交易和运输等历史事件完全相同，可以将它们归类为同一批食品持有项以进行统一管理。在处理具有相同条形码的食品时，系统可以将多个食品持有项进行组合或拆分，以凑足所需数量。这种方法将溯源功能与食品处理功能分离为两个部分，从而降低了系统的耦合度。

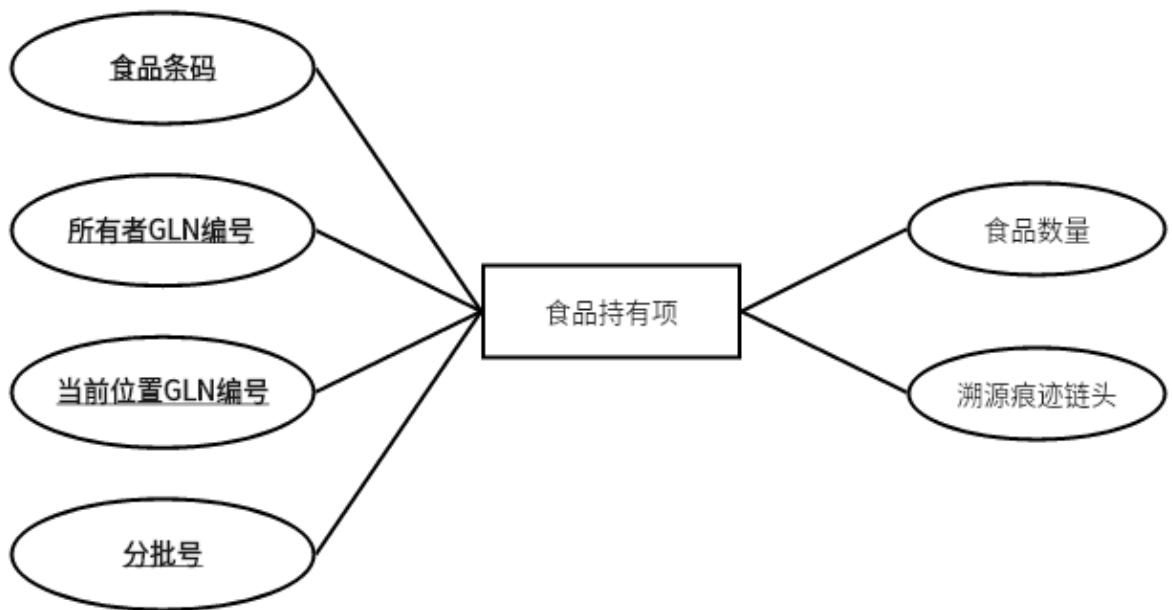


图 3.1 食品持有项存储结构图

食品持有项存储结构是一个多键哈希表,需要提供产品条形码、所有者 GLN 编号、当前位置 GLN 编号和分批号四个键来确定一批具有相同溯源痕迹链的食品。查询哈希表可以获取该批食品的元信息,包括食品的数量和食品的溯源痕迹链链头。其中,溯源痕迹链链头是查询该批次食品溯源痕迹的入口,具体原理将在后面章节详细介绍。

对细粒度定位食品持有项的方案讨论

在设计食品持有项存储结构时,最难以定夺的问题就是如何准确的定位一批具有相同溯源痕迹的食品,简单来说,就是如何设置食品持有项哈希表的键结构的问题。在系统最终确定使用的存储结构之前,我曾设计过三种不同的食品项存储方案,但最终都因不能准确定位食品项而成为废案。下面将回顾这些未被采用的方案,以此来探讨食品溯源系统对最细粒度食品项的定义与跟踪的最佳实践。

• 方案一：直接使用食品条形编码

每个食品项在生产时都有其对应的食品条形编码,而目前市面上售卖的食品中绝大多数食品都是直接使用食品条形编码来进行商品类型的标识,采用这一方案的优点是能使溯源系统直接与现有市场架构进行兼容,无需引入额外的编码方式。然而,即便是为食品条形编码引入了批次级编码的概念,也不能很好的做到定位一批原子性的食品产品。试看下面的案例:一批食品生产后,拥有相同的食品条形编码;生产商将这批食品分成两部分,一部分卖给供应商 A,另一部分卖给供应商 B;此时供应商 A 与供应商 B 所持有的食品都拥有相同的食品条形编码,在根据食品条形编码进行溯源时,无法区分其所指的是属于哪位供应商的食品。因此,直接使用食品条形编码无法满足本系统的设计需求。

• 方案二：强制使用单品级食品编码

单品级食品编码对每个不可分割的食品赋予独一无二的编码,确实能够做到精准定位每一份食品产品,不会出现方案一中混淆的情况。这种方案不仅实现简单,还能符合系统设计的需求,似乎是可行的方案。然而,这一方案所附带的缺点也很明显:首先,目前市场上售卖的食品中,采用单品级食品编码的食品占比极低,实施这一方案会带来极高的市场转型兼容成本。其次,强制使用单品级食品编码,对每一个单独的食品单独存储完整溯源信息,会使存储空间开销呈指数级增长,增加了系统的运行载荷。综上所述,强制使用单品级食品编码作为食品标识,也不是一个合理的解决方案。

- **方案三：食品条形编码 + 所有者 GLN 编码 + 所处位置 GLN 编码**

前两个方案的失败说明了仅通过单一字段来精准定位一批食品是不可行的，因此决定引入多键哈希表，试图通过多多对一的映射关系解决此问题。此方案采用食品条形编码、所有者 GLN 编码、所处位置 GLN 编码作为键，以期精准定位到所有者在某位置存放的某批食品。然而这种方式也无法避免与方案一类似的混淆问题：在给定食品所有者与所处位置时，同一编码的食品可能从不同渠道进货，并不能保证溯源链条的唯一性。

最终，本系统结合了方案三与方案二的优点，提出了前文介绍的方案四。方案四在方案三多键哈希表的基础上，为同一所有者、同一位置、同一食品编码的食品进行单独编号，引入方案二中单独编码能精准定位食品的优势，真正做到了区分具有不同食品溯源链的食品批次。

3.3 溯源痕迹链

食品的溯源痕迹是指在食品生产、交易、存储、运输等过程中记录下来的关键数据，这些数据是食品溯源系统必须获取的溯源信息。在完整的食品加工供应链条上，一批食品产品从生产到销售的过程中必然存在多个溯源痕迹，且这些痕迹之间具有时序递增的关系。因此，我们可以设计出一种溯源痕迹链存储结构，来有效地存储和管理这些数据。

溯源痕迹链的本质是单键哈希表，但其内部存储的元素通过特殊的方式相互连接成链状或树状的关系。在该结构中，每个元素不仅存储当前溯源痕迹的内容，还额外存储上一溯源痕迹元素的键。通过查询该键，我们可以访问上一个溯源痕迹元素。这样，整个溯源痕迹链就被连接成了一条有序的链表。

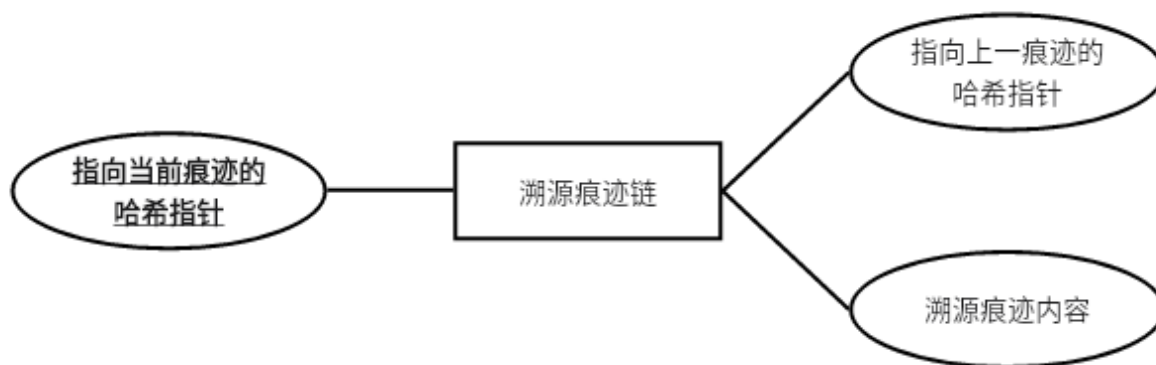


图 3.2 溯源痕迹链存储结构图

上一小节中，我们提到将溯源功能与食品处理功能解耦为两个部分的实现方法，其中包括将食品持有项进行拆分，将其变为不同批次的两批食品。这两批食品在拆分之前属于同一批次，因此它们的溯源痕迹链必然具有相重合的元素部分。溯源痕迹链采用的是面向供应商的树形结构，原理类似于前缀树，可以有效地复用相同的前导溯源痕迹元素。我们只需要将相同父元素的键附加在不同子元素的值中，就可以节省存储空间的开销，避免无谓的浪费。这种存储结构的设计非常灵活和高效，能够有效地支持食品溯源系统的实现和应用。

3.4 食品原料

在本系统中，食品原料存储结构采用了单键哈希表的形式来记录食品生产所消耗的原料集合。每种原料在系统中都被赋予了一个食品条码和溯源痕迹链链头。通过哈希链头，可以方便地查询到该批次食品原料的完整溯源信息，拓展了溯源痕迹链所能追溯到的历史范围。



图 3.3 食品原料存储结构图

通过这种食品原料存储结构的设置，系统能够将单个食品的溯源痕迹链表示成一种面向消费者的树形结构。消费者在进行食品产品的溯源时，不仅可以追溯到当前食品产品的历史痕迹，还可以追溯到每一种食品原料的历史痕迹，以及原料的原料的历史痕迹，直到追溯到农田、养殖场等没有原材料消耗的源头，实现了真正意义上的食品溯源。这种面向消费者的树形结构与上文提到的面向供应商的树形结构形成了互补关系，能够有效帮助消费者和供应商建立起简单的心智模型，降低用户使用本系统时的学习成本。

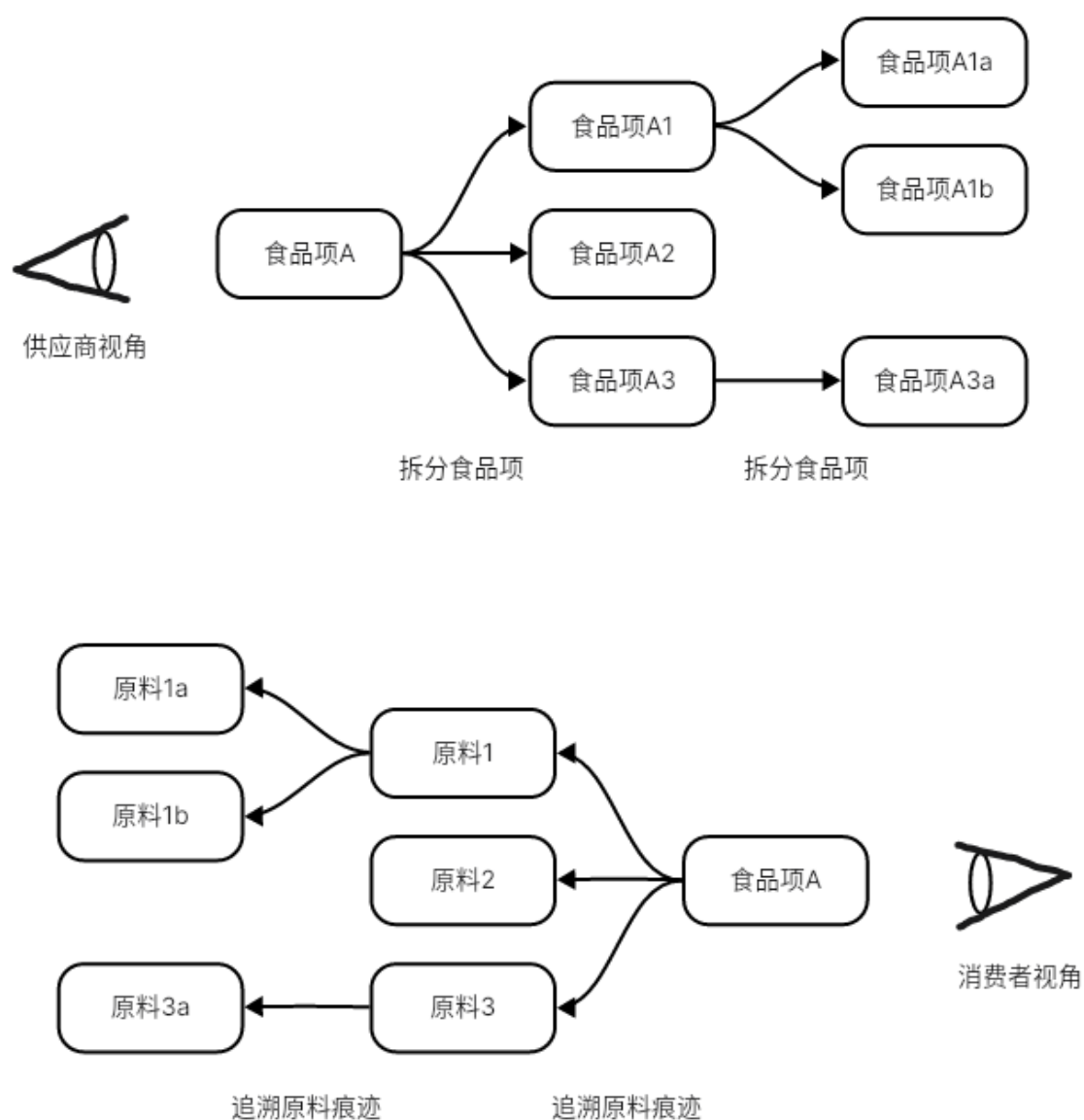


图 3.4 两类不同的树状存储结构示意图

两种树状模型同时存在但并不冲突。面向消费者的树根在供应链的末端，逆着供应链向上发散，设计目的是对单个食品的原料溯源路线进行建模。面向供应商的树根在供应链的首端，在食品的商业流通活动中某批次被拆分为多个子批次，设计目的是对真实供应链的商业场景进行建模。这两种模型的同时存在，旨在为不同主体用户提供更加全面和具体的信息，增强食品追溯体系的可信度。

3.5 账户到 GLN 映射

账户到 GLN 映射是通过单键哈希表来建立映射关系的。通过账户 ID，用户可以获取该账户所绑定的 GLN 位置编码。此存储结构的主要作用是简化用户的参数输入。当需要确定某个供应商的位置时，用户可以选择直接输入 GLN 编码，也可以选择输入供应商的账户 ID。后端在接收到用户 ID 作为参数输入时，会自动查找存储结构，以获取相应的 GLN 编码。



图 3.5 账户到 GLN 映射存储结构图

相比于直接输入 GLN 编码，通过账户获取 GLN 编码可以简化输入，这是因为在前端界面上可以访问到用户存储的常用账户“通讯录”，并以下拉框的形式供用户选择。对于尚未注册本系统的供应商，用户也可以通过直接输入 GLN 编码的方式使用系统。这种方式可以确保在系统尚未普及的初期，用户仍然能够正常使用系统。

3.6 GLN 到生产许可证映射

针对食品生产商而言，每次生产食品时都需要获取其生产许可证编码，这无疑是一项繁琐而冗余的任务。为了解决这个问题，我们设计了 GLN 到生产许可证的映射关系。当以 GLN 编码作为键查询哈希表时，就可以获得该食品生产商的生产许可证编码。

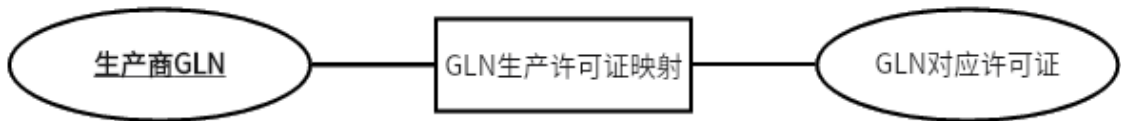


图 3.6 GLN 到生产许可证映射

在本系统的后端实现中，当供应商用户调用交易方法时，系统可以自动获取调用交易的账户 ID，并通过此 ID 查询到账户对应的 GLN 编码。如果交易方法是生产食品，就可以通过 GLN 编码查找到对应的生产许可证编码。若任何一个映射查询失败，交易都将无法完成。

账户到 GLN 的映射和 GLN 到生产许可证的映射都需要管理员权限进行设置。换句话说，在本系统真实的应用场景中，供应商必须由官方工作人员进行审核并批准才能注册本系统。这种许可链设计可以有效地防止无关节点进入系统，并能够将不安全的交易过滤掉。当发生欺诈等恶意行为时，也可以快速定位恶意账户的 GLN 全球位置编码，在现实世界中及时采取措施，避免造成更大的损失。

4 交易方法实现

4.1 简述

在区块链系统中，“交易”不仅限于描述商业交易行为，而是指使系统底层数据发生变化的动作。交易方法也可以称为状态转换函数。当系统使用者调用交易方法时，会发生对分布式数据库的写入或删除操作，从而引起整个系统的共识行为，推进系统工作进程。本系统提供六种交易方法：生产食品、交易食品、存储食品、运输食品、设置账户到 GLN 映射以及设置 GLN 到生产许可证映射。这些交易方法的实现为系统提供了高效的溯源功能，保障食品安全的同时也提升了行业效率。需要注意的是，虽然食品溯源查询功能是系统最为重要的功能之一，但由于其只是读取系统底层数据库信息而不进行修改，因此不将其作为交易方法来实现，而是作为 RPC 请求接口向外部提供。接下来，本章将详细介绍这六种交易方法的实现细节和功能特性。

4.2 生产食品

食品生产商生产食品时，需要输入待生产食品的条形码、生产商所在位置、需要生产的食品数量、食品的计数单位、食品的生产日期以及食品所要消耗的原料数组这六个参数。其中，食品的原料数组需要输入的是每个食品原料的条形码编号和要消耗的数量，系统会自动查询库存中的所有相同条形码编号的食品持有项，拆分一个大的项或合并多个小的项，以取得恰好所需要的原料数量。

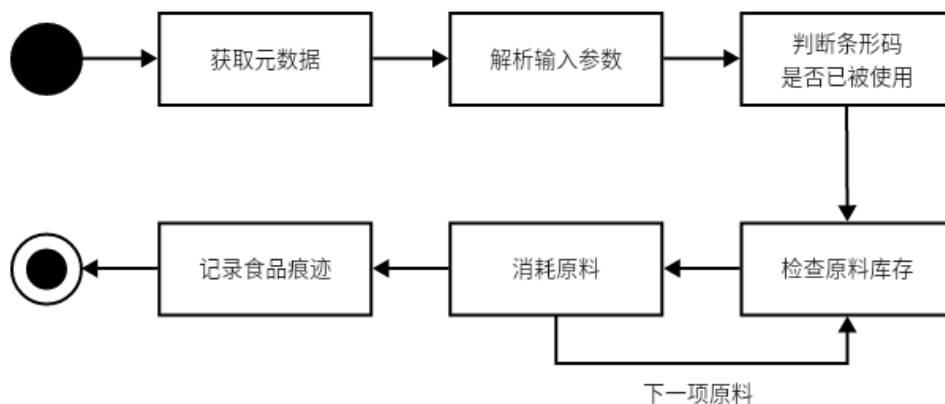


图 4.1 生产食品交易方法流程图

首先，该方法会根据对发起当前交易并签署数字签名的账户，获取到该账户相关的元数据，即当前操作的食品的所有者的信息。这是通过 Substrate 区块链框架提供的 `ensure_signed()` 函数实现的，`ensure_signed()` 函数会获取到对该交易签名的账户 ID，用获取到的 ID 查询账户到 GLN 映射，即可取得食品所有者的 GLN。同时，系统还会自动获取发起交易时的时间戳，对其进行时区调整后，将其格式化为人类可读的格式记录。

接着，交易方法会尝试对输入的参数进行解析，将输入的裸字符串参数解析为特定的格式，取得所需要的数据。对食品条形码的解析完全按照国家标准 GB 12904-2008 和 GB 16986-2018 来设计。系统会尝试解析两种格式的条形码：单品级食品条码和批次级食品条码。单品级食品条码是指能够追溯到一份单独产品的个体的条形编码，由 GTIN 和序列化 ID 两部分组成。由单品级食品条码所标识的食品持有项，其数量必须为 1，通常用于标识比较珍贵或风险程度较高的食品产品，以实现单独管理。批次级食品条码则由 GTIN 和批次号两部分组成，指的是一组数量较少的食品实例，例如在一段时间内生产的食品的集合。这两种条码在编码格式上的区别在于，序列化 ID 的前导码为 21，而批次号的前导码为 10。通过而系统的实现也针对这两种编码进行特殊优化，主要表现在于单品级食品编码的食品产品在处理时，系统会检测其所设定的操作数量是否为 1；若不是，系统会抛出错误，拒绝执行该次交易方法。同时，系统会根据国家标准 GB 12904-2008 所规定的校验方法对条形码的正确性进行校验，拒绝未通过校验的错误条形编码。

在食品的条形编码解析成功并验证无误后，系统会查询食品原料存储结构，判断当前食品条形编码是否已经被使用。在每一批食品生产时，必定会在食品原料存储结构中录入生产该食品所消耗的原料列表。就算是不消耗原料的源头食品产品，也会录入一个空列表作为占位作用。因此，若在食品原料存储结构中查询到当前条形编码已经录入系统，则说明该食品条形编码已经被使用过，系统会抛出错误并拒绝此次交易，以确保溯源系统能够更精细化、更严格地区分管理每一批不同批次的食品。交易失败时，抛出的错误信息会提示生产商更改单品级食品的序列号字段或批次级食品的批次号字段。

在生产食品交易方法中，会发生状态转换的底层存储结构主要是食品持有项、溯源痕迹链和食品原料。系统会遍历用户输入的原料数组参数，对每个原料的条形编码进行解析与校验，并查找位于生产商位置且属于所有者的该原料库存。若库存充足，则在食

品持有项存储结构中，删除被完全消耗的原料，未完全消耗的原料需要将库存数量进行相应的减少。原料消耗成功后，在食品持有项存储结构中新建食品库存，以表示一批新食品的生产成功。同时，需要将此批食品消耗原料的食品条形编码与原料的溯源痕迹链头汇集到数组中，录入食品原料存储结构。此外还需要向溯源痕迹链存储结构中录入食品的生产痕迹，用于后续的溯源行为。

在每一个交易方法的最后，需要调用 `deposit_event()` 函数，此函数与 `ensure_signed()` 函数一样，也是有 Substrate 区块链框架内置提供的，用于向区块链系统通报该交易事件成功发生。在前端也可以通过 `polkadot.js` 提供的接口接收到这一通报，从而向用户展示。

4.3 交易食品

的交易食品时，交易方法的参数需要输入食品条形编码、食品所在位置、购买方所在位置以及购买数量。同生产食品交易方法一样，系统通过 `ensure_signed()` 函数自动获取交易发生的时间以及卖出方所在位置的 GLN 编码，也会对输入的食品条形编码进行解析和校验。

交易食品的工作流程如下图所示：

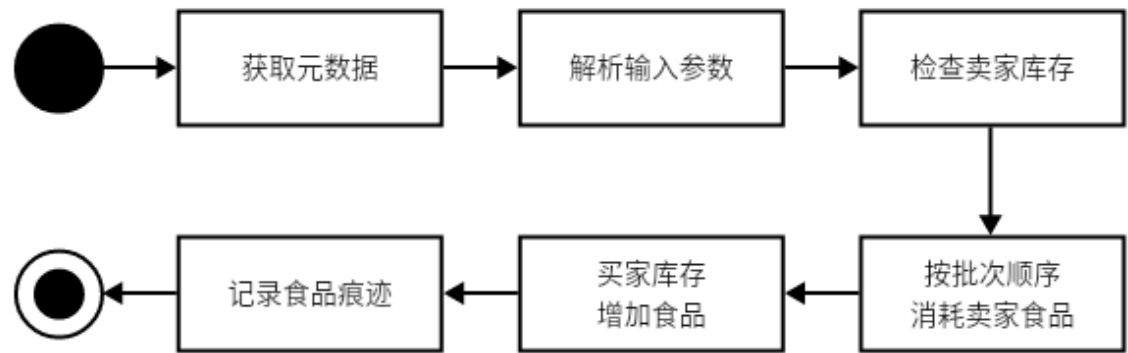


图 4.2 交易食品交易方法流程图

调用交易食品的交易方法，会使食品持有项和溯源痕迹链这两个存储结构的状态发

生变化。系统自动查询所有位于参数指定位置且属于卖出方的食品库存，拆分或删除相应数量的食品持有项，然后增加属于买入方的食品持有项。值得注意的是，交易食品仅会将食品项的所有者从卖出方变换为该食品的购买方，而不会改变食品所在的位置。这样设计更符合现实生活中的供应链情景，即食品在交易完成后仍会停留在原地；要想将其移动到购买方的位置，需要使用后文的运输食品交易方法。

交易完成后，需要将交易的行为信息写入溯源痕迹链存储结构进行记录。这与生产食品时写入溯源痕迹链不同，因为在生产食品时，该批次食品的痕迹首次出现，在时序上是最起源的状态；而在交易食品以及后文的存储食品、运输食品行为发生时，溯源痕迹链已存在前驱状态，我们需要在将溯源痕迹写入到存储结构的同时建立起链接关系，使得多个离散的溯源痕迹能够串联成链。在前文介绍溯源痕迹链存储结构时，提到这种存储结构通过存储指向上一痕迹的键来实现链式结构，而这个键就是通过对该痕迹求哈希值来生成的。系统会先通过食品持有项存储结构获取到上一状态的溯源痕迹链链头，然后生成新的溯源痕迹，并将链头附加到新溯源痕迹之后进行组合哈希；再用组合哈希所得的哈希值作为键，将新痕迹与上一状态链头作为值，存储进溯源痕迹链存储结构。新的哈希值自然成为了新的链头，系统将其存入新生成的食品持有项中，后续对该食品项进行操作时，也会取出该链头进行上述操作，以此类推。

4.4 存储食品 & 运输食品

存储食品交易方法与运输食品交易方法的实现原理基本相同，类似于交易食品，也是通过修改食品持有项存储结构，删除旧的持有项并创建新的持有项，以实现存入仓库或改变位置的功能。这两个交易方法都需要接受的参数包括食品条形编码、食品处理数量和 SSCC 编码；除此之外，存储食品交易方法还需要仓库位置作为参数，运输食品交易方法还需要出发点位置、终点位置以及承运商位置作为参数。

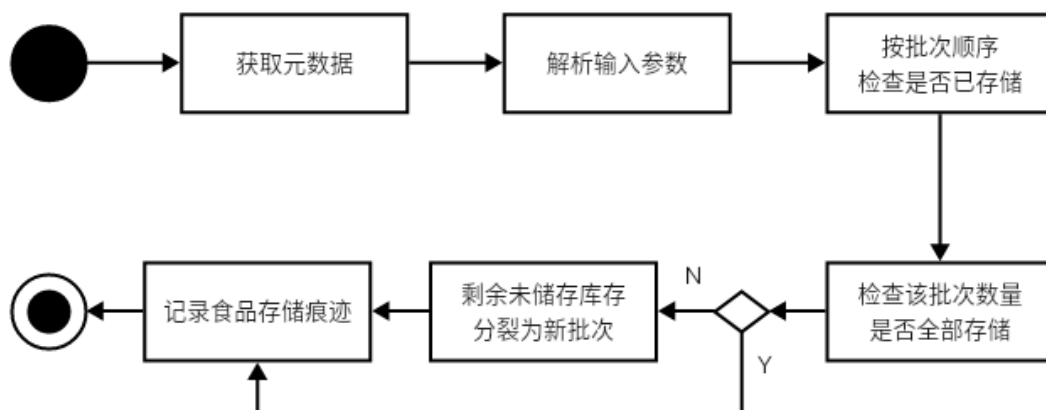


图 4.3 存储食品交易方法流程图

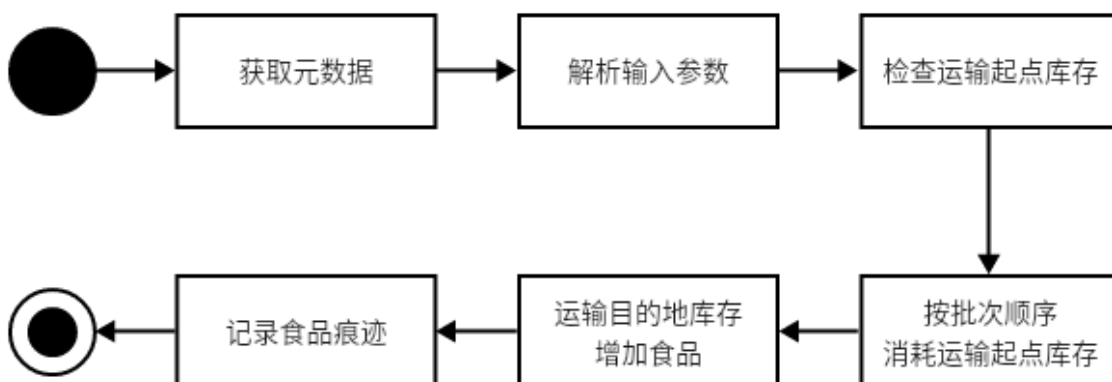


图 4.4 运输食品交易方法流程图

由于存储食品交易方法与运输食品交易方法的流程基本和前述的交易食品交易方法相同，故本小节不再冗述完整的代码展示，而是详细介绍一下 SSCC 编码以及对应的编码校验算法。SSCC 编码的全称是“系列货运包装箱代码”，是为运输时或存储时使用的物流单元提供唯一标识的代码，通常用于标识集装箱、货运托盘等物流包装单位。SSCC 编码由 18 位数字组成，包含了厂商识别代码、系列号和校验码等信息位，能够根据国家标准 GB 12904-2008 中所规定的校验算法进行有效性校验。由于 SSCC 编码与 GTIN 编码、GLN 编码都是由相同的国家标准定义的，故而可以使用一套相同的校验算法验证这三种编码。

在国家标准 GB 12904-2008 中，编码的最右侧的 1 位数字就是校验码。在对编码

进行校验之前，首先将校验码的索引设置为 1，并且定义从右到左索引逐位递增，以便于接下来对校验算法的说明。在去除校验码后，将剩下的数据中索引为偶数的位相加，得到偶数索引和；将索引为奇数的位相加，得到奇数索引和。接着，将奇数索引和与三倍的偶数索引和相加，取该结果的个位数，求该数的模 10 加法逆元，所得的最终结果即为正确的校验码。比较正确的校验码与原本校验位的值，即可判断该编码是否正确。

4.5 设置账户到 GLN 映射 & 设置 GLN 到生产许可证映射

设置账户到 GLN 映射交易方法与设置 GLN 到生产许可证映射交易方法的功能流程都比较简单，都是将接收到的参数写入到对应的映射存储结构之中。这两种交易方法都需要系统管理员权限才能正确执行，在现实世界中反映的场景是：食品供应商在注册相关资格信息时，需要由对应政府管理部门将其注册信息录入到系统中，使食品供应商获得生产、交易食品等行为的能力。交易方法中对系统管理员权限的检查是通过 `ensure_root()` 函数实现的，也是一个由 `substrate` 框架提供的身份验证函数，仅在当前签名账户具有管理员权限时才能顺利向下执行。

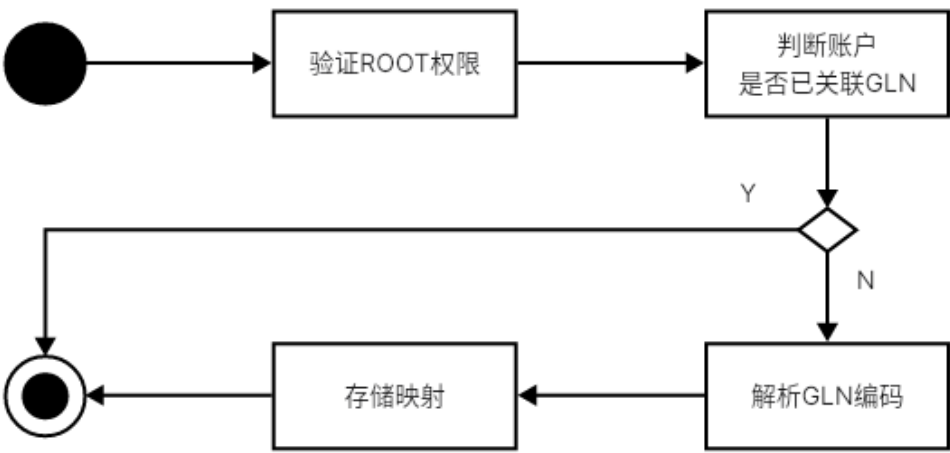


图 4.5 设置账户到 GLN 映射交易方法流程图



图 4.6 设置 GLN 到生产许可证映射交易方法流程图

在对历年发生过的食品安全问题案例的研究中，对于生产过不合格食品、发生严重食品安全问题的生产商，通常需要吊销其食品生产许可证。因此在本系统的设置 GLN 到生产许可证映射交易方法中，对生产许可证参数做了可选化设计，在不输入该参数时，即可实现注销许可证的功能。可选化输入参数是在 rust 语言中一种常见的设计模式，其数据结构本质上是一个名为 `Option` 的枚举，具有两种不同的变体 `Some(T)` 与 `None`。变体 `Some(T)` 可以附带一个任意类型的值，而变体 `None` 则不能附带值，在不同的时机选择两者作为传入参数，即可实现可选参数的多态化设计。除此之外，在市场调查中还发现一个问题，即食品生产许可证在世界各国中尚未有一个统一的标准。本系统最终选择围绕国内食品供应链的需求进行设计，对国内的食品生产商接受以字母 SC 开头的 16 位食品生产许可证编码，对国外的食品生产商接受以字母 C 开头的 18 位在华企业注册编码。这种可选的接受不同类型的参数也是通过 rust 语言中的枚举实现的，与 C 语言不同，rust 中枚举不同变体可以携带不同类型的数据，能够更灵活的构建所需要的功能。

在设置 GLN 到生产许可证映射交易方法中，需要对输入的生产许可证编码进行有效性校验。由于系统允许接受两种不同格式的食品生产许可证编码，故需要对这两种编码分类讨论。对于以字母 SC 开头的食品生产许可证编码，系统根据国家标准《信息技

术安全技术校验字符系统》（GB/T 17710-2008）中的规定，采用“MOD11，10”校验算法对其编码正确性进行验证。设置临时值初始为 0，将去除字母 SC 后剩下的 14 位数字从左往右遍历，每个数字与临时值相加后，进行模数为 10 的取模运算；模运算所得的结果若为 0，则用 10 代替，将该结果数值翻倍后，求取模数为 11 的模运算；模 11 运算的结果存入临时值中，循环重复上述操作直到对所有数字位处理完毕为止。最终所得结果除以 2，若商为 1，则校验通过，是合法的食品生产许可证编码。而对于以字母 C 开头的 18 位在华企业注册编码，其编码本身没有校验位设计，本系统只作了简单的数字字符检验。

5 RPC 溯源查询方法实现

远程过程调用（RPC）是一种通过网络报文向远程主机请求某些服务调用的方法。本研究在系统设计中使⽤ RPC 技术来实现溯源信息查询功能。通过 RPC 接口，任何连接到区块链网络的节点都可以成为 RPC 服务器，对外提供溯源信息查询服务接口。该 RPC 接口基于 HTTP 协议报文与外界进⽹络交互，接口需要接收 4 个参数，分别是食品条形码、所有者 GLN 编码、所在位置 GLN 编码以及分批号。

该方法实现的溯源查询不会对底层存储结构进⽣任何修改，只是以只读的形式获取并转换为适合人类阅读的格式，汇总信息后输出完整的溯源痕迹。具体的 RPC 溯源查询方法的流程图如下图所示：

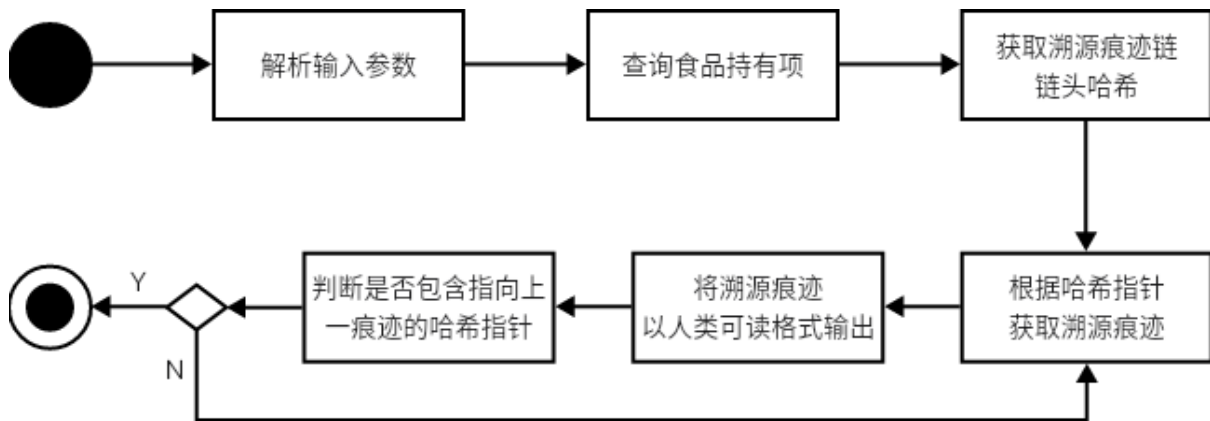


图 5.1 RPC 溯源查询方法流程图

在查询某批次食品的完整溯源痕迹时，需要先查询食品持有项，以获取该批次食品的溯源痕迹链链头哈希。接着，使用查询到的链头哈希作为键，查询溯源痕迹链哈希表。查询得到的当前溯源痕迹内容会存储到一个数组中，并判断该值中是否包含上一溯源痕迹的哈希指针；若包含，则取出该哈希值作为键继续查询，循环反复，直到查询结果不再包含上一溯源痕迹的哈希指针。这种查询方式类似于链表数据结构的遍历，只不过借助了哈希表作为数据的主要载体，将原本直接指向存储地址的指针转换为指向哈希表中某位置的哈希键。

在本系统的前后端架构设计中，RPC 溯源查询方法应由前端通过 Polkadot.js 封装好的 `api::rpc` 接口向后端发起 RPC 请求。后端收到请求并进行查询与数据处理后，将食

品项的完整溯源痕迹信息以字符串的形式返回给前端。前端会调用第三方二维码服务商的接口，将文本信息直接转换成二维码输出给用户。用户只需扫描二维码即可获得原始的溯源文本信息^[12]。

综上所述，RPC 溯源查询方法是一种有效的、可靠的远程服务调用技术，它为溯源信息查询提供了高效、准确的解决方案。该方法在系统实现中充分利用了哈希表的优势，将查询效率和数据安全性达到了最优化。

6 系统前端设计

6.1 简述

本系统的前端界面采用 Substrate UI 框架进行开发，以保证用户界面的优化和用户交互的流畅性。Substrate UI 框架提供了一整套可重用的组件，以及基于 React 的实现方式，让前端开发者可以更轻松地创建复杂的用户界面。同时，本系统还基于 Substrate front-end template 模板进行二次开发，通过添加自定义系统逻辑实现了针对溯源信息查询功能的特殊需求。通过 Polkadot.JS 提供的接口与后端区块链代码进行交互，实现了前后端的数据传输和信息展示功能。用户可以通过直观的界面交互操作来查询食品溯源信息，同时系统可通过二维码的方式展示数据，提供更加直观的数据分析和判断。整体而言，系统的前端界面设计考虑了用户体验，主要面向商户提供了一种高效便捷的交易方法调用与生成溯源信息的方式。



图 6.1 系统前端界面模块展示

本系统的前端界面由三个主要模块组成，如图 6.1 所示。从上到下，这些模块包括当前账户模块、区块链信息展示模块和交易方法模块。在当前账户模块中，用户可以通过点击右上角的账户名称来进行不同账户之间的切换，以便以不同身份登录系统。中间

的区块链信息展示模块分成四个部分，分别为连接到区块链网络的端口及区块链版本、区块链运行时元信息、链上当前区块数量和链上已确定化的区块数量。最下面的交易方法模块是用户与系统进行交互的主要区域，用户可以在此处填写信息，并将参数发送到后端执行交易。除了简单的输入参数之外，本系统还提供了一些特殊界面交互，例如生产时追加原料和可选的参与方参数。下面将介绍这些特殊的界面交互，并测试本系统的溯源信息查询结果以二维码形式在前端界面上展示的功能。

6.2 生产时追加原料 交互界面展示

在图 6.1 中已经展示了生产食品界面的组件布局，用户输入完成生产食品交易方法所需要的参数信息后，点击左下角确认生产，即可发送交易。右下角提供了两个按钮，用于动态的添加原料和删除原料。点击添加原料按钮后，系统会在下方自动追加两行输入框，用于输入原料的食品条形编码以及消耗原料数量；点击删除原料，则会隐藏新增的原料输入框。图 6.2 展示了用户追加两种原料时，系统的交互界面。

The screenshot displays the '食品溯源系统' (Food Traceability System) interface. At the top, there's a header with a logo, a version string 'v4.0.0-dev-d28d299017a', a 'Show Metadata' button, and a user profile 'ALICE'. The main content area is titled '食品溯源系统' and features a sidebar with navigation buttons: '生产' (Production), '交易' (Transaction), '存储' (Storage), '运输' (Transport), '设置GLN' (Set GLN), '设置证书' (Set Certificate), and '生成溯源码' (Generate Traceability Code). The '生产' tab is active, showing a form for production details. The form includes fields for '生产商' (Manufacturer), '条形码号' (Barcode), '生产数量' (Production Quantity), '计数单位' (Counting Unit), and '保质期(日)' (Shelf Life). Below these is a table for '原料' (Ingredients) with columns for '原料' (Ingredient) and '数量' (Quantity). The table currently shows two rows. At the bottom of the form, there are buttons for '确认生产' (Confirm Production), '添加原料' (Add Ingredient), and '删除原料' (Delete Ingredient).

图 6.2 追加原料时的界面展示

6.3 选择参与方位置 交互界面展示

前文第 3.5 小节提到过一种简化用户输入参与方位置的方式,可以选择输入账户 ID 或者 GLN 编码;当输入账户 ID 时,系统通过查询存储结构来获取对应的 GLN 编码。在前端界面中,通过<Dropdown/>组件来创建一个下拉框,提供两个选项供用户选择输入参数的类型。

若用户选择“账户”一项,尾随的输入框会变成一个带有 search 属性的<Dropdown/>下拉框组件,此组件既可以直接通过下拉框点击选项选取,也可以通过输入一部分字段查询所需要的选项。在选择账户的下拉框中,系统通过 Polkadot.JS 所提供的接口,访问到用户保存的账户 ID “通讯录”去获得选项。用户可以方便快速地选择已保存的、熟悉的合作参与方。

图 6.3 与图 6.4 以交易食品的界面为例,展示了通过下拉框切换输入参数类型以及选取账户 ID 的前端交互。

食品溯源系统

生产 交易 存储 运输 设置GLN 设置证书 生成溯源码

商品位置 账户 ALICE

购买方 账户 选择买家位置

GLN编码

条形码号

交易数量

确认交易

图 6.3 切换参数类型时的界面展示



图 6.4 选择账户 ID 时的界面展示

在本系统的前端设计中，下拉框组件不仅用于简化参与方位置参数获取难度，还用于实现对生产许可证的设置与注销功能。在设置证书界面中，可以通过下拉框组件选择设置证书或注销证书。若为设置证书，则显示输入框以让用户输入证书编号；若为注销证书，则不提供输入框。注销证书时的界面如下图 6.5 所示：



图 6.5 注销证书时的界面展示

6.4 简单输入检查 交互界面展示

在用户输入参数时，前端会简单验证数据的正确性，例如字母数字检查、参数长度检查等。在检查失败时，会以适当的方式告知用户错误信息。图 6.5 通过“设置 GLN”的界面展示了在参数输入错误时，前端界面会作出的反应。



图 6.6 输入检查失败时的界面展示

6.5 测试生成溯源码

在本章的最后，我将尝试构建一条简单的供应链测试数据，并测试系统能否正确识别测试数据的溯源痕迹信息，并成功生成食品溯源二维码。

首先，定义某生产商的 GLN 编码为 6901234567892，食品生产许可证编号为 SC10634102130056。该生产商生产了 100 份苹果，该批次苹果的食品条形编码为 0104607163090624101111。生产完成后，生产商将苹果经由 GLN 编码为 6951234500009 的承运商，将 70 份苹果运往与承运方位置相同的仓库。抵达仓库后，对该批苹果就地存储，并导出苹果存储后的溯源二维码。

下图 6.7 与 6.8 展示了查询并导出二维码的界面和扫码该二维码所显示的食品溯源信息：

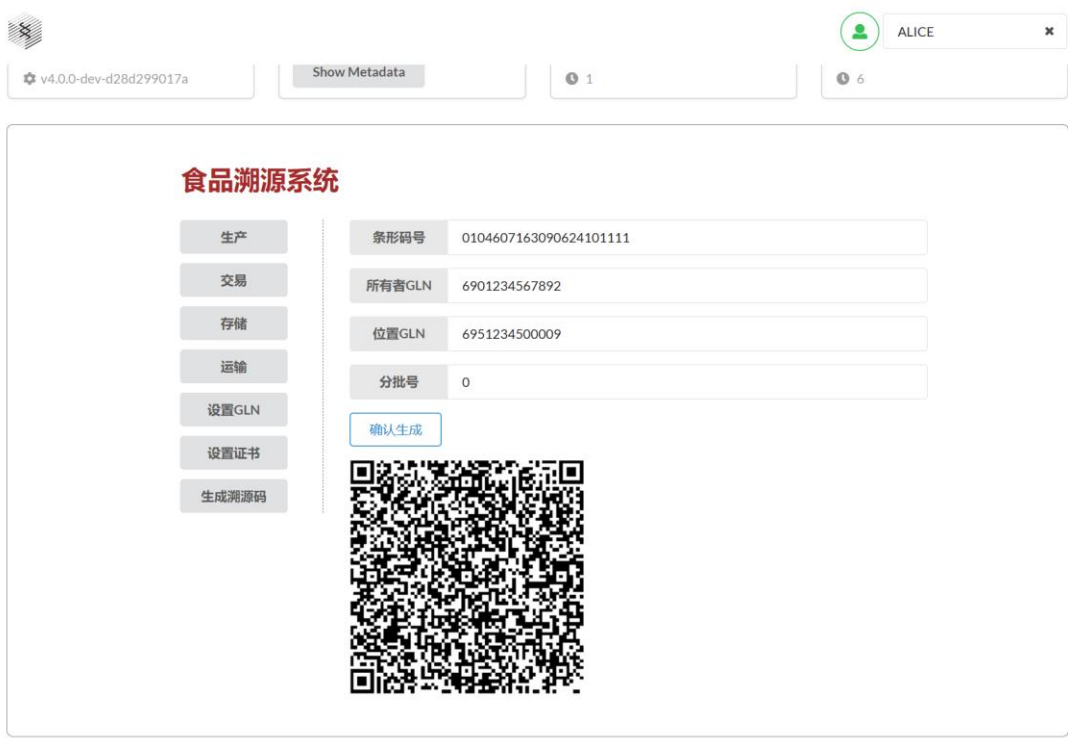


图 6.7 查询并导出食品溯源二维码的界面展示

以下扫码的内容并非微信提供，请
谨慎辨别信息

```
{{存储: 存储时间: 2023-04-29
14:29:42仓库GLN编号:
6951234500009装箱码:
069479426234567899}}{{运输: 运输
开始时间: 2023-04-29 14:27:36起
点GLN编号: 6901234567892终点GLN
编号: 6951234500009承运商GLN编
号: 6951234500009装箱码:
069479426234567899}}{{生产: 生产
时间: 2023-04-29 14:26:42生产商
GLN编号: 6901234567892许可证编
号: SC10634102130056计数单位: 一
份保质期: 3天}}
```

如需使用，可通过复制获取内容

图 6.8 微信扫描溯源二维码的结果展示

需要注意的是,溯源痕迹信息中并不包含任何有关食品条形编码和食品操作数量的相关信息。这样做是为了保护食品供应链中各方的商业机密,防止恶意人员通过溯源痕迹信息推测出参与方的产能和资金流等机密信息,保证了供应链的可信度和数据的安全性。

7 结论

本研究设计并实现了一套基于区块链技术的食品溯源系统,它能够使食品生产和流通的各方在整个过程中共享数据,并能够保证数据的安全性和不可篡改性。本文介绍了该系统的设计思路、存储架构和实现方法,并详细阐述了其中的六种交易方法以及食品追溯查询功能。通过该系统,消费者可以方便地了解所购买食品的生产、运输、存储等环节的信息,从而提高消费者的食品安全意识,保障消费者的健康权益。本系统利用区块链的不可篡改、去中心化等特性,解决了传统食品溯源系统中存在的数据造假等问题,保证了数据的真实可信性。然而,本研究的系统仍然存在一些需要进一步优化的功能,接下来将介绍这些功能以及可能的解决方案。

(1) 从地图选取 GLN

在本系统中, GLN 编码用于记录供应链中各参与方所处的位置,从而使得用户可以进行多种交易。不过,在一些较为复杂的供应链中,可能涉及到较多的参与方,手动输入或者搜索账户名会显得繁琐不便。为了解决这个问题,我们可以考虑在系统中添加地图 API,将地图上的位置点与其 GLN 编码相关联。当用户需要查找特定的参与方时,用户就可以在地图上直接选取所需的位置坐标,快速地找到目标参与方。通过这种方式,用户就不必费时费力地手动输入 GLN 编码或账户名来找到对应的参与方,从而提高了系统的使用效率和易用性。

(2) 使用摄像头识别条形编码

为了进一步简化用户输入数据的工作量和提高系统的易用性,后续可添加摄像头扫描条形码的功能。通过此功能,用户可以使用手机等设备对商品上的条形码图像进行扫描,即可快速识别出条形编码对应的数字信息。与手动输入条形码相比,这种方法不仅更快,而且减少了输入错误的可能性。这种功能的实现将使用户更轻松地使用系统,并大大提高其实用性和效率。

(3) 专门的溯源查询服务器

本系统在输出溯源二维码时,使用了第三方二维码服务商接口,将溯源信息文本转换为二维码。这种方式的缺点是对长字符串会生成尺寸很大的复杂二维码,内容展示的格式也不够美观,无法准确识别一些简单的转义字符,从而影响了用户的使用体验。因

此，我们建议在优化中引入专门的溯源查询服务器作为中转，该服务器可以将溯源信息转换为美观的网页格式，供用户扫码访问该网页。通过将溯源信息转换为更易于理解和访问的形式，用户可以更方便地获取所需信息，提高了用户的使用满意度。另外，在递归溯源食品原料时，还可以为用户提供超链接跳转的方式访问原料的溯源页面，这也符合用户常见的网络习惯，进一步增强了用户体验。

（4） 将编码转换为人类可读的名称

本系统在展示溯源信息时，展示的是最底层的条码编码信息，这种方式对于一般用户而言显得过于晦涩难懂，不利于信息的传达和使用。因此，在优化中我们建议结合政府维护的官方供应链注册信息数据库，将编码转换为更易于理解的形式，如正常的公司名称、食品产品名称等。通过这种方式，用户可以更直观地了解产品的信息，提高了数据的可读性和理解度。同时，这种转换方式可以避免用户对编码进行手动解析，避免了人为解析出错的风险，提高了信息的准确性和可信度。

参考文献

- [1] 邱炜伟,李伟. 区块链技术指南[M]. 北京:电子工业出版社, 2022.
- [2] 张聪. 沃尔玛中国启动区块链可追溯平台, 创新保障食品安全[J]. 食品安全导刊, 2019(19):15.
- [3] 张文胜,蒋翊. 中国食品安全问题现状分析及多元应对之策[J]. ICCS Journal of Modern Chinese Studies, 2015(8):1-5.
- [4] 易萌. 食品质量安全问题的成因及对策研究[J]. 产业与科技论坛, 2013,12(11): 9-10.
- [5] 曾小青,彭越,王琪. 物联网加区块链的食品安全追溯系统研究[J]. 食品与机械, 2018,34(9):106-111.
- [6] 董云峰,张新,许继平,等. 基于区块链的粮油食品全供应链可信追溯模型[J]. 食品科学, 2020,41(9):30-36.
- [7] 赵巧润,曹宇璇,曹怡凡,徐笑,程抒劼,陈贵堂,曹崇江,袁彪. 互联网+区块链技术在食品安全溯源体系中的应用及研究进展[J/OL]. 食品工业科技. <https://doi.org/10.13386/j.issn1002-0306.2022050328>
- [8] 苏瑞国,阳建,秦继伟,武晓雄,贾振红. 基于物联网区块链的轻量级共识算法研究[J/OL]. 计算机工程. <https://doi.org/10.19678/j.issn.1000-3428.0063845>
- [9] 卢瑞元,王子恒,李领治,王进. 基于 Hyperledger Fabric 的区块链农产品溯源方案[J]. 计算机科学与应用, 2020,10(5):811-823.
- [10] 于坚,孙嘉笛,纪剑,等. 区块链技术在食品溯源体系中的应用[J]. 食品工业科技, 2021,42(15):377-382.
- [11] 唐衍军,许雯宏,李海洲,等. 基于区块链的食品冷链质量安全信息平台构建[J]. 包装工程, 2021,42(11):39-44.
- [12] 王宇. 基于二维码的农产品溯源平台研究[J]. 自动化与仪器仪表, 2016(9):94-95.
- [13] 张瑞星. 基于区块链技术的食品溯源平台关键技术研究[D]. 四川:电子科技大学,2021.
- [14] 王丹妮. 基于区块链的可溯源供应链系统关键技术研究[D]. 湖南:湖南师范大学,2021.
- [15] 吴远. 基于区块链技术的食品安全追溯方法研究[D]. 上海应用技术大学,2019.

- [16] Jim Blandy, Jason Orendorff, Leonora F.S. Tindall. Programming Rust[M]. United States of America: O'Reilly Media, Inc.,2021.
- [17] Elise Golan, Barry Krissoff, Fred Kuchler, Linda Calvin, Kenneth Nelson, and Gregory Price, Traceability in the U.S. Food Supply: Economic Theory and Industry Studies[R], (2004-03). https://www.ers.usda.gov/webdocs/publications/41623/28673_aer830_1_.pdf?v=0
- [18] Baralla G, Pinna A, Tonelli R. Ensuring transparency and traceability of food local products: A blockchain application to a smart tourism region[J]. Concurrency and Computation-Practice & Experience. 2020,11(4).

致谢

在本篇论文的撰写过程中，我受益于许多人的支持和帮助，因此在此向他们表示由衷的感谢。

首先，我要感谢我的指导教师，他在学术上对我的研究进行了指导和支持，同时也在生活上给予了我许多关心和鼓励。他的教诲和帮助让我在论文撰写过程中更加有信心和坚定，没有他的帮助，我无法完成这篇论文。

其次，我要感谢我的同学们，他们是我学术生涯中最重要的支持系统。在论文撰写过程中，我们一起面对学术和生活的各种挑战，共同进步，互相支持和帮助。他们的友情和陪伴让我的学术之路更加充实和有意义。

此外，我还要感谢我的家人和伴侣，他们是我生活中最重要的支持和后盾。在论文撰写的压力下，他们始终给予我精神上的支持和关怀，让我能够专注地完成这篇论文。他们的陪伴和理解是我前进的动力和力量。

最后，我要特别感谢 OneBlock 社区的各位成员，感谢他们为本论文提供的参考资料和技术支持。在 OneBlock 社区的帮助下，我更深入地了解区块链技术和供应链溯源领域的最新进展，使得本论文更加完善和实用。他们的知识和经验对我的学术研究和未来的职业发展都有着重要的影响。

再次感谢所有给予我支持和帮助的人，愿我们在未来的学术和生活中都能够取得更加出色的成果。我将继续努力，为自己和他人创造更加美好的未来。