# Event-based Supply Chain Network Modeling: Blockchain for Good Coffee

**Simon L. Bager** [1], **Boris Düdder** [2,*], **Fritz Henglein**[2], **Juan Manuel Herbert**[3], **and Haiqin Wu** [2]

[1] *Earth and Life Institute, Université catholique de Louvain, Belgium*
[2] *Department of Computer Science, University of Copenhagen, Denmark*
[3] *Deon Digital, Denmark*

Correspondence*:
Boris Düdder*
boris.d@di.ku.dk

## ABSTRACT

Blockchain provides the potential for a secured decentralized information infrastructure for efficient value chains and effective horizontal integration across institutional boundaries. It offers a multitude of innovative ways to handle transactional data in a decentralized fashion. Supply chain management has been a prominent blockchain application due to the trust and immutability provided by a blockchain. Blockchain-as-a-Service could enable secure value transfer and transparent governance of complex supply chains, such as premium and sustainable commodities like certified or specialty coffee.

We present REALISTIC, a modeling framework for supply chain networks that includes production processes. It is built on McCarty's Resources-Events-Agents (REA) accounting model and is extended with secure transformations, which guarantees that physical resources can be consumed and produced but cannot directly or indirectly lead to resources being produced out of thin air.

We present a case study for an event-based blockchain solution for end-customers in an end-to-end commodity supply chain. The model handles product provenance, transparency, sustainability and quality information, production, transportation, and certification standards across the coffee supply chain. The model is based on an existing coffee supply chain involving farmers in Colombia through cooperatives, processors, traders, importers, and a Scandinavian roasting company. Its REALISTIC-based analysis is the foundation for the design and prototypical implementation of a decentralized supply chain network proxy for validation. Its source code is made available on GitHub.

Overall, the development process allows insights into the iterative co-design of general blockchain solutions for sustainable commodity supply chains. Our fully implemented open-source prototype validates the practicability of the event-based supply chain network modeling in field tests. We find that REALISTIC modeling for event-based systems using iterative co-design is a promising approach for the development of complex supply chain applications.

*The co-authors are listed alphabetically; their order carries no significance. Please contact a co-author for authoritative information about co-authors' individual contributions.

**Keywords: Blockchain, coffee, supply chain, provenance, certification standards, sustainability**

## 1 INTRODUCTION

A modern *supply chain network (SCN)* is a decentralized network of independent legal entities that collectively produce, transport, trade and finance physical products so they eventually reach consumers reliably and efficiently. An SCN describes the flow of economic resources and information across economic agents and the processes by which raw materials are transformed into the final goods eventually distributed to customers.

SCNs play a particularly prominent role in the agro-food sector, helping move agricultural commodities from places of production to places of consumption. Today, people in cities across the world rely on food produced by farmers far from the place of consumption and handled by numerous traders and manufacturers along the way. Demand for agro-food commodities is growing, as populations rise, economies grow, and consumption patterns change (Godfray et al., 2010). At the same time, the agro-food system is the second largest source of greenhouse gas emissions ((IPCC, 2019) and the main driver of global biodiversity loss ((IPBES, 2019). Actors across the agro-food system are increasingly aware of the impact that commodity production, processing, transport, consumption, and disposal along SCNs has on the global climate and environment. As a result, NGOs, civil society, and consumers, and even some governments, pressure the sector to reduce the negative impacts (Bager et al., 2021; COWI et al., 2018).

Governing agro-food supply chain sustainability is extremely complex due to the opaque nature of global agro-food commodity supply chains, the general complexity of the system, as even simple chains can involve hundreds of companies and thousands of suppliers, and the commodification of products, where individual commodities are mixed, sold, re-sold, and repackaged several times before reaching the final consumer (Bager, 2021). This makes it challenging, often downright impossible to guarantee where, under what conditions, and with which impact individual commodities are produced. Take a coffee supply chain as an example. It involves coffee bean producing, processing, packing, transporting, and selling among farmers, cooperatives, mills, roasters, transporters, distributors, shops and customers (Ikhwana, 2018).

Trust and the lack of trust are also crucial concerns in SCNs, including agro-food supply chains, since the participants are typically spread over different countries, cultures and jurisdictions, and they focus on their own self-interests. Such trust issues may result in reluctant data sharing, sub-optimal processes and deliberate misbehavior (Düdder and Ross, 2017). For example, a farmer may be unwilling to disclose the cost of raw materials to another farmer due to fierce marketing competition. Relying on centralized trusted third parties (TTP) is a common way to build trust among participants, but it is impractical and infeasible in real world to find a TTP that everyone is willing to place not only their trust in, but also all their data with. Furthermore, the heavy workload in the centralized party may lead to slower information flow, time-consuming trading procedures, and delayed money transfers. In light of the above concerns, there is a need for a novel decentralized systems architecture that reflects the dynamic, decentralized nature of SCNs without particular producers, technology providers, government institutions, or banks that all other SCN participants would have to entrust sensitive data about financial and product flows.

As a type of agro-food commodity, coffee is produced by millions of farmers, many of these being smallholders farming a few hectares of land. Producers are often organized in cooperatives, who handle transactions with traders and mills further downstream. After initial processing in the country of production, the coffee is exported as green coffee to importers and roasters – mostly based in Europe and the US – who roast, process, and package the coffee for final retail (Bager and Lambin, 2020). The final coffee is often a blend consisting of coffee from several countries and hundreds of different producers. However, coffee is increasingly marketed as single origin or, in the case of specialty coffee, directly from individual cooperatives or farmers, so-called “direct trade” (Bager and Lambin, 2020; Grabs, 2017; Ponte, 2019). Some companies also aim at providing increased transparency of their operations or full traceability of their goods along the supply chain. Substantiating such claims requires segregated supply chains able to guarantee the provenance and specific characteristics of the coffee. Guaranteeing specific sustainability practices, pricing transparency, or compliance between certification standards and the given product also requires knowledge about the specific supply chain and the means to document this.

Fraudulent activities can be common in agro-food supply chains and, thus, trust between participants can be low, requiring constant and costly verification of commodities and data. Currently, companies’ main approach to foster trustworthy sustainability information is to initiate various governance mechanisms including, *inter alia*, pledging environmentally friendly production, certifying their products, reducing deforestation across supply chains, increasing efficiency, and lowering energy and water use (Bager and Lambin, 2020; Dauvergne and Lister, 2012; Lambin et al., 2018; Thorlakson et al., 2018; Ponte, 2019). The actual impact of these initiatives on improving livelihoods or environmental sustainability across coffee supply chains is mixed (Bager and Lambin, 2020; Garrett et al., 2021; Meemken, 2020; DeFries et al., 2017). Although standards and initiatives can reduce environmental impact or increase income under certain conditions, effects are often contingent on context-dependent factors such as commodity, region, or supply chain characteristics and confounded by the influence of strictness and criteria of the intervention and a lack of systematic verification and evaluation (Lambin and Thorlakson, 2018).

The emerging *blockchain and distributed ledger technology (BC/DLT)*, or just *blockchain* is featured as a decentralized network with a shared global ledger among all peer nodes, which perfectly fits in the supply chain with multiple participants involved and without a central controller. Owing to the cryptographic techniques like hash-chain and digital signature, and the distributed consensus protocols, blockchain achieves tamper-resistant and transparent recording of peer-to-peer transactions with a trustworthy and consistent state even though the peers do not trust each other. Moreover, the advances in smart contract development provide more flexibility in implementing the business logic and activities, such as real-time data entry, product sale, shipment, receipt confirmation, user certification, and peer-to-peer payment, in a reliable and automatic manner. Such prominent characteristics exhibit great promise in resolving the aforementioned issues and challenges in existing SCNs. Although some stakeholders may have their own information systems to handle transactions and commodity flows with local storage, the integrity of such information is guaranteed once the hash value is anchored to the blockchain. The on-chain data will serve as permanent and immutable (non-paper-based) proofs/evidence for tracing specific transactions and offer accountability in the presence of fraudulent activities from malicious participants or disputes.

Proponents have argued that BC/DLT may have a revolutionary impact on business and society, e.g., (Tapscott and Tapscott, 2017). SCNs employing blockchain have attracted considerable attention in recent year, and its integration with other technologies like the Internet of Things (IoT) enables efficient and flexible data recording for public audit of each commodity’s status through a supply chain (Wang et al., 2019b). As illustrated before, SCNs are characterized by having economically competing participants and limited trust between some actors. Further, an SCN should ideally be designed without a single, dominant supply chain agent controlling all downstream suppliers. A decentralized organization enables non-dominant actors, particularly upstream actors such as farmers, to gain better access to and control over the data entered onto the blockchain. However, the most compelling argument for using blockchain is the need for tamper-proof data handling and storage, as participants need to ensure that other (untrusted) participants do not alter the data after input. Although the integration of blockchain with SCNs are promising for a safer, immutable, and traceable supply chain management (SCM), commodity supply chains have complicated structures and aim for various objectives including high quality, low cost, fast speed, high trustworthiness, lower risk, and better sustainability. Blockchain-based SCM achieving the above goals is arguably still in an initial phase, requiring considerable additional research, education (Düdder et al., 2020), and development.

The coffee sector is known for widespread sustainability initiatives with the highest certification levels among any commodity. It has widespread use of certification standards and private governance mechanisms, such as direct trade or transparency initiatives (Bager and Lambin, 2020; Ponte, 2019; Grabs, 2017). Still, only one third of all companies across the coffee sector take tangible action to address sustainability, leaving a large potential for additional action (Bager and Lambin, 2020). To improve market access and support sustainability governance, companies and other supply chain stakeholders propose various technological solutions, most notably blockchain. So far, there is limited application of blockchain across the agro-food sector beyond initial pilots, but the field shows massive growth and a large potential. In the coffee sector, large-scale roasters such as Starbucks already experiment with the technology, and various small, innovative solution providers are emerging, such as Bext360 or iFinca. Despite supply chain innovation, the field lacks systematic academic research, especially empirical studies and models. [1]

The main contributions of this paper are:

- A novel event-based methodology, REALISTIC, for systematic modeling of deep supply chain networks including decentralized production, transportation, economic resource transfers, user-definable data aggregation and analytics, third-party observations and assertions, and contract management between independent economic agents. It is based on classical Resources-Events-Agents (REA) (McCarthy, 1982) accounting and builds on extensions and experience with REA over the last 30 years.
- An event-driven system architecture whose components reflect and co-evolve with REALISTIC models. It is based on digitally signed event assertions made by humans, IoT devices or other systems. This provides the basis for tamper-proof real-time tracking, tracing and digital enforcement of physical constraints such as limiting the amount of certified products by the intake of certified raw materials. The architecture is implementation platform agnostic; in particular it supports being rapidly developed using centralized database systems with powerful and rapidly deployed out-of-the-box functionality and subsequently porting to BC/DLT-based systems for decentralized governance.
- Explicit, user-definable specification of functional models that analyze and aggregate the events and facilitate tracing a final product back through a complex network of production and transportation steps, ownership/possession transfers, tests and certifications, and the reverse flow of payments corresponding to product flows. Not only the events, the functional models themselves are transparent and tamper-proof and can be reviewed and audited.
- A case study that covers tracking and tracing coffee from coffee cherry to coffee cup, covering all stages of coffee production, certification, testing, storage, transportation and eventual consumption.
- A fully implemented open-sourced prototype system, including a web app for QR-code tracing and visual illustration of a coffee bag's journey and sustainability characteristics from original farm to retail bag of roasted coffee. Its design and implementation reflects and supports our hypothesized advantages

[1] For additional information about different blockchain applications in coffee supply chains in Colombia, see the study by (Bager et al., 2022), which is also part of the Good Coffee project.

of employing powerful database and user interface technology for rapid requirements elicitation and prototype development with full-fledged functionality, delaying migration to a BC/DLT platform until requirements are stable and decentralized governance is sufficiently important.

The remainder of the paper is organized as follows. We introduce materials and methods in Section 2. Section 3 shows the results with a Good Coffee use case and gives the design accordingly. We discuss our design with a comprehensive evaluation, followed by a brief retrospection and conclusion in Section 4.

## 2 MATERIALS AND METHODS

In this section we present a functional characterization of blockchain systems, broadly construed as decentralized systems providing tamper-proof data recording and digital resource ownership. We identify two additional aspects crucial for digital shadows of SCNs and introduce an event-based framework for systematic modeling of and accounting in SCNs. Based on the framework, we derive a functionality-driven method for identifying important, missing and opportunistically usable (electronic) data and develop an ontology for fine-grained tracking and tracing of coffee from farmer to consumer, which will serve as the blueprint for the component architecture, design and implementation of our Good Coffee system.

### 2.1 Blockchain deconstructed

*Blockchain (BC)* is a popular term used ambiguously for the original Bitcoin system (Nakamoto, 2008) (where the name originates) including systems built on its core assumptions and technical design decisions such as Ethereum (Buterin, 2014; Wood, 2014), or for the much broader class of peer-to-peer computer systems that provide a combination of decentralization, tamper-proof digital recording of events, and guaranteeing non-duplication of digital assets (“tokens”) such as IBM Hyperledger Fabric (Androulaki et al., 2018) and R3 Corda (Brown et al., 2016a,b).

The term *distributed ledger technology (DLT)* is often used to emphasize that such systems may significantly deviate from the specific principles of the original Bitcoin design. Below we use the term *blockchain/distributed ledger technology (BC/DLT)*, or just the slogan *blockchain*, in the general all-encompassing sense. We describe a BC/DLT system by the following defining characteristics (Henglein, 2018).

- *Decentralization.* The system is decentralized, that is it is not only a distributed system, but also a peer-to-peer system where the nodes and network connections are controlled by *legally and economically independent* parties such that each party has no more control over or access to information than any other party. Strictly speaking, this excludes a technical set-up where the BC/DLT-based system is cloud- or server-hosted since the cloud provider has highly privileged (i.e., much more) access to and control over the system than the other parties. In this sense, “hosting a blockchain system on the cloud” would be a *contradictio eo ipso*.
- *Tamper-proof recording.* Collectively, the system establishes and achieves a consistent state of recorded information over replicas by maintaining a logically single log of digitally signed events. The recording may contain auxiliary data as *evidence* of *real-world* events, e.g., pictures, DNA-analyses, signed statements, IoT device signatures, etc. Consistency means that there is no significant discrepancy between information provided by one node and any other node in the peer-to-peer network. Furthermore, the recorded information is *tamper-proof* and *available*. It cannot be altered or deleted without being noticed by other participants, and it remains effectively available to all parties entitled to access it. A particular BC/DLT-based system *may* restrict who gets access to which information in contrast to Bitcoin/Ethereum-style systems, which require that all recorded information is completely public

and accessible to anybody without requiring any authentication. A widely used technique for tamper-proofing is cryptographic hashing, which can also be done on a centralized system without any replication or distribution of the data.

- *Secure resource management.* The system provides support for managing ideal (purely digital) and, by proxy/tokenization, physical resources. A resource is something that can be transferred, but is guaranteed by the system not to be duplicated or lost. This is in contrast to (data representing) *information*, which when transmitted from one party to another is *copied*: both the sender and the recipient have it after transmission. After a successful transfer of a digitalized *resource*, however, only the recipient has it. Resources are money, securities, shares, exclusive rights (ownership), IOUs (promissory notes, bonds, loans), (digital proxies/tokens representing types of) physical objects (trucks, books, apples, coffee cherries, parchment coffee, green coffee, roasted coffee), etc. Transferring guarantees that the sum of resources is not changed; combined with (a generalized form of) double-spend prevention the system guarantees that transfers are valid (the sender is the correct owner) and atomic (whether or not the transfer succeeds, exactly one of the sender and receiver ends up owning the resources).

The second and third requirements, tamper-proof recording and secure resource management, express an *Olympic view* of the world: there is always one authoritative ("true") state of the world (single point of truth), and resources can neither pop out of nowhere nor disappear mysteriously.

- An event has happened or it has not; if it has, there is one true description of it, and once happened it cannot be made unhappened or made to look like something else happened. Mortals may disagree on the state of the world, but there is one and only one truth: the authoritative Olympic view.
- Resources, both physical and ideal, exist only once; an event may rearrange them—change their location, ownership or possession—but does not change their sum total. Mortals may claim to own something, but that is either true of false, determined by the authoritative Olympic view.

The first requirement, decentralization, expresses that the Olympic view is entirely conceptual: There are no Olympian gods or even demigods, only mortals collectively emulating the Olympic view.[2]

## 2.2 Modeling supply chain networks

A (physical) SCN has the following general properties.

- It is organizationally *decentralized*. Ideally the SCN should be open to participants joining and leaving, and no participant should have more control over and access to the network as a whole than others—each participant manages their own information, processes, and products.
- It is *event consistent*. En event involving multiple parties (e.g., a money transfer, a product delivery, or an invoice sent) either has happened or it has not; if it has, then it cannot subsequently be changed or be undone.
- It is *resource preserving under transfers*. Physical (products) and ideal (money, ownership rights) resources are transferred without getting magically copied or disappearing. In a successful resource transfer, it is guaranteed that the previous owner was the correct owner and, whether successful or not,

[2] The requirements constitute security guarantees, that is the *guaranteed absence* of functionality deemed to be undesirable. In the case of BC/DLT-based systems this is the *guaranteed* or at least trustworthy absence of a party getting privileged access to other parties' data, the guaranteed absence of being able to change or delete (records of) events, and the guaranteed absence of duplicating or losing digitally represented resources. Technically achieving the trustworthy absence of such undesirable functionality is *incomparably more difficult* than simply offering it with a *caveat emptor* sticker attached to a system that says that ensuring the security requirements is the application programmer's job, with little or no support from the system.

after the transfer exactly one of the two parties in the transfer is the owner; no resources are duplicated by accident or by cheating.

Additionally, an SCN intrinsically deals with *production* and *transportation* of physical resources, not only natively digital ones, which yields additional general properties.

- It is *location consistent*. Physical resources cannot be in two disjoint locations at the same time.
- It is *resource preserving under transformations*. Physical resources are not only transferred and transported, but also produced, that is *transformed* into each other: roasted coffee is produced from green coffee; green coffee from parchment coffee; parchment coffee from coffee berries. After each production step there is less of the input resources and more of the output resources. So the sum total of resources of each kind *can* change as the result of a transformation event, but nothing can directly or indirectly be produced out of thin air. It is not possible to produce 160 bolts from 3 kg of steel and later melt the bolts into 4 kg of steel.

We stipulate that a *digital proxy* of an SCN should incorporate these general properties as fundamental *invariants*; that is, it should guarantee that they always hold.

The first three properties amount to a decentralized (peer-to-peer) computer system with no single party in control, authoritative event recording with no possibility of tampering, and digital resource management (ownership and transfer validation) guaranteed to prevent digital forging (double spending). These are the characteristics of a BC/DLT-based system; see Section 2.1. The fourth property is a form of no-double-spending guarantee, where a location “owns” a physical resource and transportation is a transfer of the resource to another location.

Guaranteeing resource preservation under transformations, the fifth property, is much trickier than only preventing *double spending*, resource preservation under transfers. We employ a new technique that subsumes double spend prevention and prevents resource creation *ex nihilo*. It is based on requiring resource transfer and transformation steps to be elements of the kernel space of a linear mapping (in the sense of linear algebra) from resources to resource characteristics such as mass or weight (for physical resources). The property is then enforced by admitting only production steps that must not lead to an increase in the sum total of these characteristics (Torres Garcia, 2020).[3]

In this analysis, a *digital proxy for a supply chain network* consists of

- a *network of digital twins* of real-world products that are produced by authenticated producers by transformation from explicit ingredients/raw materials;
- a *decentralized (peer-to-peer) computer system* with no single party in control, authoritative event recording with no possibility of tampering, and transfer and transformation validation that is guaranteed to prevent digital forging (double spending).

The latter are the characteristics of a BC/DLT-based system from see Section 2.1, but enhanced with *transformation validation*. A producer submits a digitally signed transformation event to the BC/DLT-based system for one or more new products, which the BC/DLT-based system validates only if

- the characteristic (e.g. mass) of the certified input resources in the transformation event equals or is greater than the characteristic of the new products and

[3] This is analogous to the First Law of Thermodynamics: The total amount of energy, the physical characteristic a system’s components are mapped into, can neither be created nor destroyed.

- the input resources are previously certified products and have not been spent or otherwise decertified in the meantime.

Once validated, the new products atomically constitute new *certified* products and their inputs are *spent* and thus decertified—they cannot be reused to produce additional certified new products. There may be additional requirements for product certification, but not fewer than these.

In our pilot study we keep track of the masses of inputs and outputs of production steps to guarantee that every production step does not output certified resources with more mass than the certified resources it consumes.

Secure resource transformations for certified production processes with guaranteed impossibility of increasing the amount of certified resources across a decentralized network of independent agents are a key innovation of our work for supply chain networks in general, since they are not supported out of the box by existing BC/DLT-based systems. A backtrace of a final product may have a flawless and complete history of steps via certified semi-finished products to certified raw materials, with carefully tracked transportation along the entire chain, yet it can be defeated by using an inferior semi-finished product at any stage and equipping it with a duplicate of a high quality certified product. [4] This is indirectly a form of double spending, which a digital SCN proxy as above with secure resource transformations disincentivizes or makes outright impossible.

Managing resources on a BC/DLT-based system that digitally guarantees resource preservation throughout production and transportation is a massive disincentive to cheat in itself. The most common attack vectors for bringing in an inferior (unknown origin, untracked, uncertified, not sustainably grown, etc.) product are impossible or disincentivized.

- Equipping an inferior product with the certificate of a high-quality certified product effectively replaces the original high-quality product by an inferior one since the certificate cannot be reused. (This is the standard no-double-spend guarantee provided by BC/DLT-based systems.) While *stealing* certificates is still possible, it disincentivizes collusive certificate forging and labeling inferior products with high-quality product certificates.
- Getting a certificate by claiming a low-quality product has been newly produced as a new high-quality product does not work since production requires the consumption of at least as many certified input resources. For raw materials the input can consist of quota issued by a regulated authority plus recycling of acquired *certified* materials. The latter incentivizes a circular economy: Getting one's hands on certified discarded end products increases one's production volume of *certified* raw materials beyond the quota issued.
- Producing one high-quality product and issuing two or more certificates for it to use the additional ones on low-quality products does not work: The two certificates require twice the amount of input certificates; a producer cannot digitally make products by "issuing" them *ex nihilo*.

### 2.3 REALISTIC modeling

We propose a novel model we call REALISTIC accounting for business and productgion modeling, in particular supply chain network modeling. REALISTIC is based on the REA accounting model and subsequent changes and extensions such as adding view independence (Jacquet, 2003), information and business events (David, 1997), and complex contracts (Andersen et al., 2006; Henglein et al., 2007). It is based on the following concepts.

[4] This explains why there is substantially more certified virgin press olive oil sold than olives actually produced for it.

- *Resources:* Resources are ideal (money, bonds, shares, any exclusive rights) or real (trucks, bicycles, parchment coffee, etc) things that can only be transferred; they can neither be duplicated (copied) nor discarded (lost).[5]
- *Events:* Events change the state of (what we know about) the world. Economically important events include transfer (an agent transfers ownership or possession of a resource to another agent), transport (an agent moves a resource from some real or virtual location to another one), transformation (an agent produces something, that transforms one resource into another one), information (an agent transmits data to another agent, afterwards they both know it) and observation (an agent ascertains something and makes a statement) events. An event is a digitally signed and dated statement, possibly with associated documentation as evidence of the claimed correctness of the statement.
- *Agents:* Agents are natural persons, legal entities, divisions of companies, associations, IoT devices, informal sets of these including dynamic and anonymous sets of them (e.g., "Bitcoin"). Agents are involved in events; they can cause or validate them. Conceptually, a primitive event is a signed and dated statement by a single agent; e.g., a transaction making it onto the Bitcoin blockchain expresses "I, Bitcoin, herewith validate this Bitcoin transaction and declare it to have happened in the 582,453th block"; or "I, the IoT device knowing the private key corresponding to public key 0xFFFA074l75A670CD herewith declare that coffee grader with ID 0x66878786DFA6F5E4 has measured 1.4 kg coffee beans and assessed it to have color blib and grade blob. Digitally geo- and timestamped and signed, 0xFFFA074l75A670CD." or "I, the Acme Cooperative herewith transfer ownership of coffee bag with QR code 0x77AB77819CDA67DD to Alice Traders. Digitally geo- and timestamped and signed, John Doe of Acme Cooperative."
- *Locations:* Both digital and physical resources are usually at some physical or virtual location; for example, a truck is located on some parking lot and some money is located in a particular jurisdiction. Locations occur in transport events.
- *Information:* Data and things that, in contrast to resources, are easily copied (duplicated); e.g., instruction manuals, invoices, emails.
- *Strategies:* Processes (in economic terminology strategies) employed by a single agent to get something done, e.g., whether to produce something for inventory or not, whether to accept a contract offer or not, when to perform something required or possible in a contract. Strategies are basically all the manual (human) and automated (algorithmic) mechanisms inside a company (a *single* agent) that cause it to generate and sign events. Strategies usually take the form of computer code and data in enterprise resource planning (ERP) systems that, possibly automatically or semi-automatically (controlled by a human sitting in front of a terminal, laptop, iPad, iPhone, etc), do something on behalf of their (sole) owner; e.g., automatically paying a bill according to a running contract.
- *Time:* Events happen in both time and space (locations). Time is used to express when an event happened, when that event was registered, by which deadline an event (such as payment or a delivery) should contractually happen, etc. Time is expressed in calendar time (next month, next week) or physical time (in 4.2 seconds) or both. Note that time often occurs in connection with a geographic location (such as a time zone), but exists independently: 5 seconds in the Pacific Standard Time zone and 5 seconds in the Central European Time zone are the same amount of time simply because 5 seconds are 5 seconds.

[5] In this setting, losing something in the real-world sense is possible by moving it to a "lost" account.

- *Identities:* Identities are identifiers associated with agents, resources, events, information, contracts or any other entity; these can be used to name and track particular entities and to keep the entities themselves secret. Identities cover tokens both in the cybersecurity sense (access tokens) and in the digital asset sense (named, trackable digital resources).
- *Contracts:* A contract is a specification of possibly multiple sets of events, each of which constitutes the *successful* execution of an agreement between multiple agents. A basic exchange contract, for example, consists of two transfer events that can be executed in any order. A contracted is *breached* if the agreement is not followed up by one of the set of events; e.g., if a bicycle in an exchange contract is delivered, but the requisite payment is not performed. Contracts are fundamental to economic activity.

An *event log* is a sequence of events, where events can neither be changed nor deleted, only added at the end. Since the prohibition against deleting or changing entries is a hallmark of bookkeeping, the container of an event log is called a *ledger*, which may be replicated or otherwise distributed; in such case it is often called a *distributed ledger*. Whether distributed or not, a ledger can be thought of as an append-only (paper or electronic) file, where events can only be added at the end.[6]

We have only required that events be (digitally) signed and dated. So a priori, a statement can be anything at all; also, the optional attached documentation can be anything at all.

An event can be classified into different important categories.

- *Resource transfer:* Ownership or possession of a resource is transferred from one agent to another.
- *Resource transport:* A resource is moved from one location to another, without changing ownership or possession.[7]
- *Resource transformation:* A resource is produced from another resource.[8]
- *Resource bundling and unbundling:* A (compound) resource may be bundled into an identifiable new unit; e.g., 96 jute bags of parchment coffee may be bundled into 1 (possibly numbered) pallet of parchment coffee containing the 96 jute bags. Unbundling is the inverse, it consists of retrieving the contents of the bundle. Bundling models the real-world action of putting something together (bag, box, pallet, container) for shipping and storage and equipping the whole item with an identifier (label, QR-code, sequence number) that can be tracked and traced.[9]
- *Information transmission:* Some data is sent from one agent to another; e.g., an invoice is sent. Information transmissions themselves are data; e.g., an agent can send the information that she has sent an invoice to her accountant.[10]
- *Observation:* An agent makes an assertion about some aspect of the state of the world, e.g., the quality and quantity of a particular bag of coffee or asserting (e.g., by QR-code scanning) where it has been seen when (tracking).[11]

[6] The requirement of a totally ordered event log—a sequential listing of all the events—is computationally difficult and expensive to achieve and unnecessarily strict, yet common in BC/DLT-based systems. Partially ordered or even weaker relations between the *set* of stored events are sufficient for common applications. Nonetheless, we stick to the idea of a totally ordered event log for simplicity of explanation.

[7] Abstractly, this is also a kind of transfer.

[8] A resource can be made up of many different kinds and quantities of some resource type. We call it a *compound* resource; so "a" resource may in reality be a large and complex set of more elementary resources.

[9] It is a special kind of transformation, where each transformation has an exact inverse. The "produced" resource (1 pallet with a certain label is "produced" from 96 jute bags of parchment coffee) can be "unproduced" by unbundling (the 96 jute bags of parchment coffee are "produced" from the pallet).

[10] Sending the information may or may not be allowed. In contrast to resource transfers, after transmission *both* the sender and the receiver "own" (know) the data; it is copied.

[11] Observations are conceptually a form of "lazy" information transmission; e.g., a coffee quality and quantity observation is an information transmission to everybody permitted to see it.

Note that REA and REALISTIC are ontologies for modeling real-world concepts, not for expressing computer system architectures or implementation techniques. In particular, REA can be used without employing technical terminology from blockchain systems, relational database systems or for that matter any computer systems eventually used to digitalize its concepts. This is analogous to entity-relationship modeling (which identifies real-world entities and describes their relationships in information technology independent terms) and relational database design (which describes a technical system architecture based on table schemas, keys, relational queries, etc). The latter is derived from the former; the former is independent of the latter. A litmus test for retaining this modeling-implementation separation is this: Coffee supply chains have existed before the advent of blockchain and database systems. How would we describe them in, say, the 1950s?

### 2.4 Digitalization for coffee provenance

How can REALISTIC modeling be used for tracking and tracing individually serialized retail coffee bags and reliably associating provenance information, ecological sustainability practices, independent certifications, and assuredly fair pricing for farmers with them? Which data is needed, desirable or already available in digital form?

For field trips, one to Colombian coffee growers, cooperatives, and a dry mill, the other to a roasting facility in Denmark, we developed the following methodology for collecting on-the-ground information and identifying requirements driven by the concepts of REALISTIC modeling.

- Think about the questions you want to answer eventually and use them to figure out which primitive events are *needed*. The three *driving* questions for the Good Coffee project are:
  - Where is the coffee in the cup I am holding from and how did it get there?
  - How much money did the farmers pay for the coffee that eventually ended up in my cup?
  - Which quality and production characteristics are associated with this coffee?

  More specifically, our end user system should include
  1. traceability and chain-of-custody information for coffee at all stages;
  2. associated economic transactions, specifically the price paid to farmers when selling their coffee;
  3. tracking of fine-grained certification information, e.g., Rainforest Alliance or Fairtrade certification standards;
  4. tracking of fine-grained quality-related information, such as cupping scores, roasting profiles, and crop variety;
  5. various ecological and social sustainability indicators and the data related to these, such as renewable energy consumption, tree (re)planting, water recycling, labour contracts.

  These functional requirements were given as core objectives from the outset.
- Identify the *primitive events* required and, in the process, the agents and resources involved in them. A primitive event is a single-sentence statement (see categories above) that is not broken down into more fine-grained events and is made by a single agent. The agent furthermore *takes sole responsibility* for the correctness of the *entire* event.[12]
- Think about events that are, in traditional terms, related to *master data* and *transactional data*, respectively. Transactional events are production (transformation), bundling/unbundling, transport, transfers including payments, notifications and tracking events; master data events are general

[12] A roaster would for example take sole responsibility for “I received this bag with QR-code XYZ at my factory gate”, but not for “The coffee in bag with QR-code XYZ is from Cooperative C and sustainably grown in Columbia.”

statements about the agents and resources, which certifications a farmer or cooperative received when by whom, etc.

- Think of a simple *functional specification* for getting an answer to one of the driving questions from an entire log of primitive events. This is to identify missing and useful information (primitive events) that should be registered. For example, if there is not a single registered event that says *anything* about how and which coffee berries were transformed into parchment coffee, then tracing coffee back ends with parchment coffee. Conversely, even the least amount of information in registered events can be exploited; e.g., if the coffee berries were delivered by Farmer X and Farmer X happens to have some certification at that time, it may justifiably be concluded that the coffee came from certified berries. Think about *clearly specifying* the answer, but do not think about the computational *complexity* of computing it.[13]
- Which events are in *exchange* for other events? Contracts are tit-for-tat agreements eventually tying multiple events together, including payments and ownership transfers. For answering the second question it is important to record what the *dual* event, the *particular payment* for the sale (ownership transfer) of a particular batch of coffee is.[14]
- Observe in the field which information is already collected for whatever reason. Take pictures, videos and notes on the spot. Whether needed or not such information may contribute to answering the original questions and may *opportunistically* be used to answer additional questions, e.g., in disputes. Indeed, their availability may motivate raising questions simply because they can be answered. In particular:
  - Which tracking observations (by automatic or semi-automatic scanning or manual registration) are made when transporting the coffee in its various forms? The more fine-grained and detailed this information the more difficult it is for somebody to "cheat" without being detected.
  - Which information is registered about resources (coffee in various forms and packaging) coming into and out of a production or trading site? This information is crucial for tracking and tracing raw materials and semi-finished products into the finished products; in the case of coffee, from coffee berries via parchment and green coffee to roasted coffee.
- If relevant information is already collected or it is clearly advantageous to collect, what does it take to digitize it? i.e., get it recorded on a computer, smartphone or other networked device? If it is documented on paper, is it because it was captured electronically (in which case the information is already digitized) or is it because it is still an entirely manual (analogue) paper process? What does it take to produce it in electronic form instead?

We think of a particular specification of an answer based on primitive events as a particular *interpretation*. We do *not* make computing the answer according to specification an event in its own right. Primarily this is to keep focus on "real" real-world events, separate from their eventual interpretation, without spinning off into a black hole of treating interpretations as events, interpreting interpretation events, etc.

Mathematically, a specification is a *function* from a given event log to an answer; in database terms, this is analogous to a particular query being applied to the table(s) containing the registered primitive events. There can be *different* functions applied to the same data, yielding different results to the same questions. As such a particular function represents a particular *model* of what happened. Making the model explicit

[13] If you are a computer scientist, do not think about it *yet*; if you are not a computer scientist, do not think about it; that is what computer scientists are for.

[14] This can be done in different ways. A conceptually simple model is to associate a unique identifier, somewhat confusingly sometimes also called a *correlation id*, with all the events that pertain to the same contract.

and separating it from the registered raw events rather than conflating it with raw events ensures that it can be analyzed in isolation of particular individually signed events.

For example, if all we know from the event log is that a particular container of 8,000 kg of green coffee was produced Friday from 4,000 kg green coffee from one cooperative delivered the Tuesday before and 6,000 kg green coffee from another cooperative delivered Thursday, then a *particular model* of the output composition may specify that it is made up proportionally from each input, so it contains 3,200 kg from the Tuesday batch and 4,800 kg from the Thursday batch. *Another* model might say that the input batches are processed in temporal order, so the particular output contains 4,000 kg from the Tuesday batch and 4,000 kg from the Thursday batch; or it employs an analytic mix that depends on quantities of inputs *and* their delivery dates and reflects a mathematical approximation of how bulk coffee is unloaded in the particular warehouse. Note that the input data to all three models are the same, the two deliveries Tuesday and Thursday.

# 3 RESULTS

We present our findings on requirements and UI-driven co-design; REALISTIC-based application design; and employing a blockchain-late implementation strategy for functionality-first development.

## 3.1 Good Coffee user requirements and system specification

The coffee supply chain targeted by our event-based digital shadow runs from coffee farmers in Colombia through to a coffee roastery in Scandinavia. The coffee farmers involved are all smallholders farming around 2 hectares of coffee on average, who are organized in a cooperative. All farmers are Fairtrade certified. The farmers grow and pick the coffee - some higher labourers for this - and subsequently wetmill and dry the coffee before packing it in jute bags. The farmers transport the coffee to purchasing points, which are operated by the cooperative. Here, the coffee is quality controlled, and farmers are paid according to the daily rate, factoring in the quality. The farmers receive payment and a receipt. The cooperative stores the coffee in jute bags, but individual farmers' bags are not labeled separately. Given current practices, provenance is lost at the purchasing point, as individual farmers' coffee bags are not kept separate. For the purpose of this project, we labeled individual bags using QR-codes. From the purchasing point, the cooperative organizes transport to the dry mill. Here, the coffee is milled (hulling, sorting, quality control, etc.). At large scale mills, such as those currently used in this supply chain, coffee again becomes mixed, as multiple bags (from several farms and cooperatives) are milled in the same batch. Micro-mills can mill coffee from different farms separately, and thus for the purpose of this project, the coffee was milled at a micro-mill. After milling, the coffee is packed in 70-kg jute bags carrying the name of the processor and stored at a warehouse before it is prepared for transport. For the purpose of this project, we used bags containing QR-codes and the project name to ease identification and facilitate data entry. For transport, the coffee is placed in a sealed container and transported on truck to the port. From here, it is exported to Europe on container ships. From the importing port, it is transported by rail to the warehouse. Here, the container's seal is broken, and the coffee is distributed (by truck and ship) to the different roasteries. At the roastery, the coffee is roasted according to the roaster's preference and re-packaged for retail. The final bags also contain a QR-code (see Figure 1 for overview of the supply chain).

The existing sustainable coffee supply chain involved multiple independent organizations. Unfortunately, there exists no end-to-end coherent IT solution for supporting the whole supply chain from farmers to coffee cups. The horizontal integration along the supply chain is hindered by disconnected data silos operated by the different organizations. The data transfer is mostly based on physical documents involving media breaks between text document and data storage. Manual data entries and very little automation due

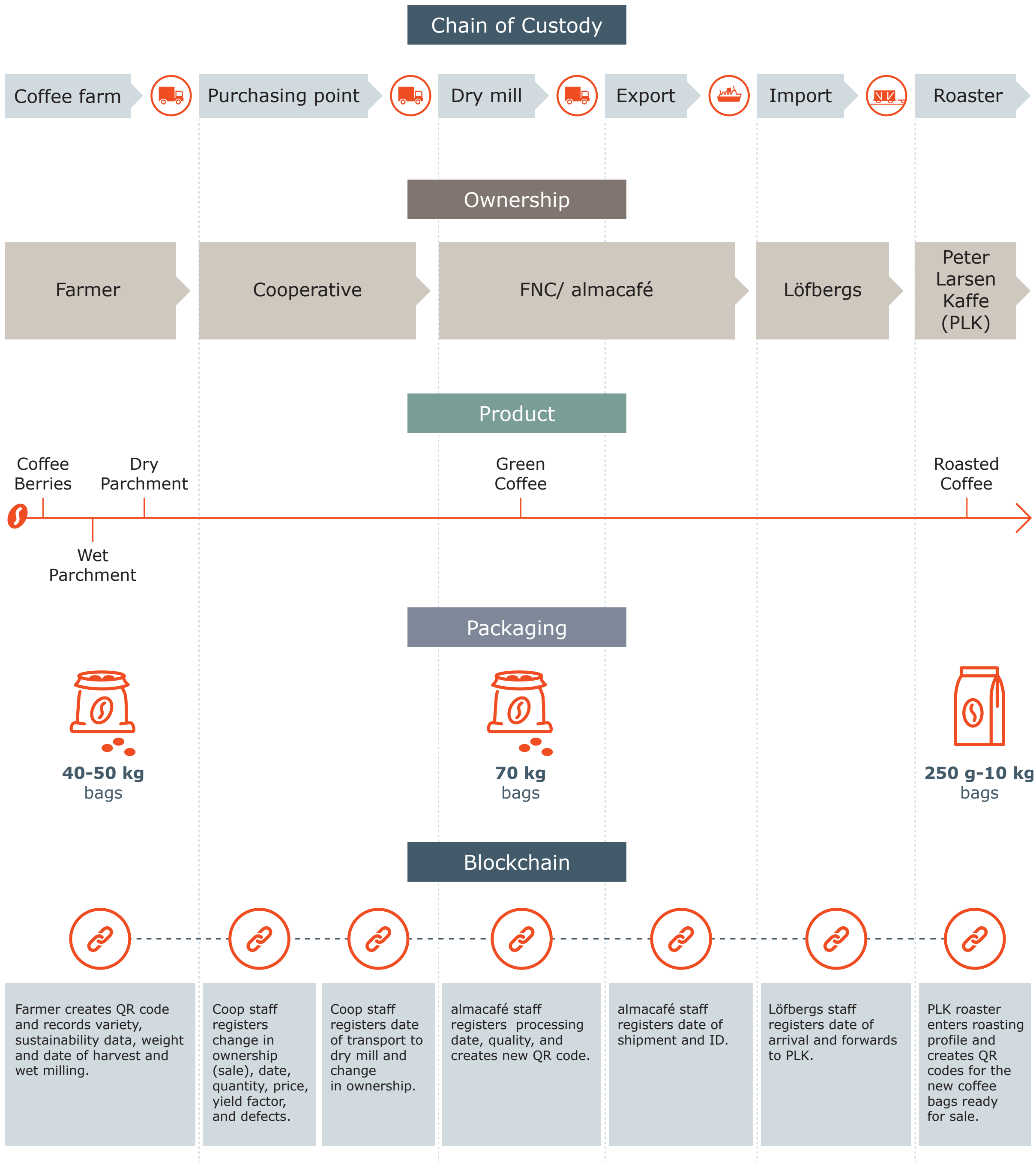


**Figure 1.** Overview of a coffee supply chain with blockchain support.

to legacy systems and their difficult integration become the dominating factors impairing SCM efficiency and quality. The paper trails can only get verified and checked at pre-defined points.

The downstream buyer (importer-roaster) has good and trusting relations with mid-stream processors (drymill and exporter), but relies on the midstream participants for engagement upstream, having relatively little ability to engage in or detect fraud in the upstream parts of the supply chain. The verification is conducted either at quality gates (e.g., purchasing point, dry mill) as part of the product quality assurance or due diligence in the case of fraud detection late in the supply chain, e.g., roaster or importer complaints.

Coffee as a commodity undergoes different production steps which inherently transform the product and render physical tagging impossible for securely tracking either coffee beans or batches. The tags in use are connected to batches of coffee beans and are not forgery resistant.

The supply chain is, furthermore, characterized by a high degree of fan-in and fan-out, e.g., parchment coffee collection points of cooperatives, as well as importers and exporters purchasing from and selling to multiple companies. The supply chain is decentralized without a dominant participant controlling the entire supply chain, though certain agents have more leverage than others. For example, the roaster and importer are much larger and more financially capable than the cooperative or the individual farmers. Thus, the organization and governance of an end-to-end supply chain has to answer the question who is paying for investment? The policies around the supply chain are issued/made by government and other regulators, as well as issuers of voluntary sustainability certification standards, in this case, Fairtrade International and auditors acting on their behalf. Individual agents in the supply chain have a partial and limited knowledge about the rest of the supply chain and involved agents. This is especially pronounced for upstream participants. Farmers only know that the coffee is sold to the cooperative, and then further processed and exported, but have no knowledge of downstream agents or their desires. Cooperatives know the processors/exporters, but do not engage with the roasters nor the final consumers. The roaster, for their part, knows only the processors/exporters, and has limited knowledge of the cooperative and very little engagement with the individual farmers. As the only agent engaging both up- and downstream, the processor cum exporter hold more knowledge about practices and requirements along the supply chain. The business relationships are characterized by mutual trust between adjacent business partners enforced by contracts and constant checks.

Some notable user requirements that have been elicited and derived into functional requirements are:

- special requirement of some sustainability certificates is a reverse cash flow of premiums payed by the end-user to the cooperative (and subsequently, albeit indirectly through investments in production facilities, to the farmers) (in an ideally fair amount).
- reverse traceability was necessary to realize fair payments and to support sustainability- and quality-related data.
- modeling data and evidence of production steps, storage, and transfer.
- because of fraud prevention, storage and production within a production step were captured in detail, e.g., individual storage silos and their content as well as individual coffee mills.
- logical validation of data based on on-chain historical data as well as provenance information.

During the discussions with experts and other project stakeholders, some non-functional requirements emerged, constraining the solution space for the design, for example: demand for confidentiality and privacy; the usability of the final product by untrained farmers and a cooperative's staff; no special hard- or software because of a lack of local IT support staff. The remote location of farmers poses a special challenge, while transportation with ephemeral connectivity disallows always online solutions. This is particularly the case at the farm level, where not all farmers have access to the internet or smart devices (For an in-depth discussion of in-field implementation and farmer challenges, see (Bager et al., 2022) and Singh et al. 2022, respectively).

### 3.2 Co-design with project stakeholders

A pre-project analysis indicated multiple unstable requirements and various goal conflicts, e.g., transparency and intellectual property protection. A full up-front analysis was considered unrealistic

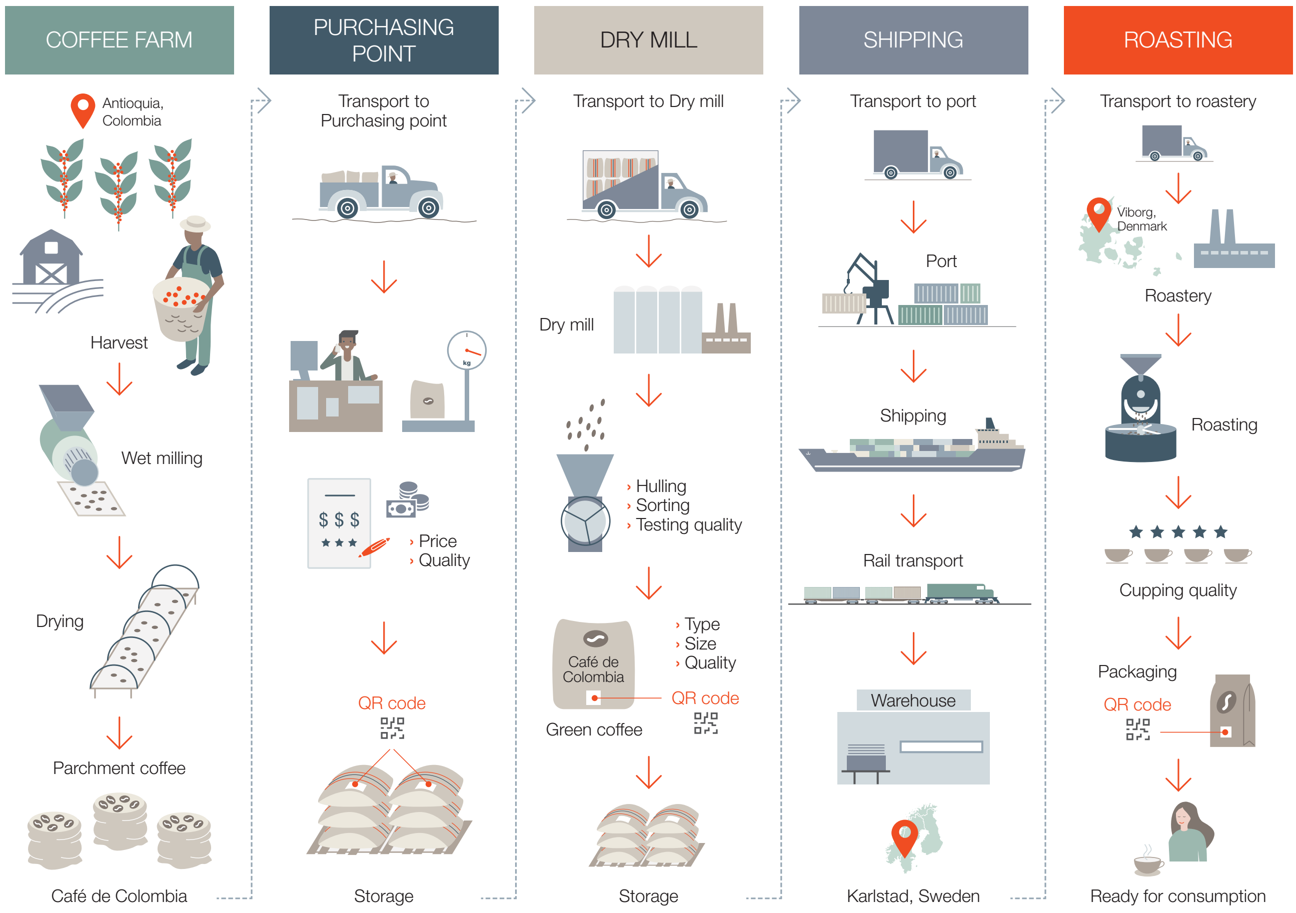


**Figure 2.** Full supply chain with all stakeholders and their corresponding processing procedures (Bager et al., 2022).

and, therefore, an explorative approach for the design was chosen, in particular, due to the inexperience of the developer team about the complex coffee supply chains. Unstable requirements emerged from changes in feature prioritization and emerging requirements, e.g., due to the Covid-19 pandemic. Thus, we adopt a co-design method as a viable design process, a.k.a., participatory design, allowing comprehensive and immediate feedback and allowed to exploit an unrestricted solution design space in close collaboration with all stakeholders. The co-design methodology has a long tradition in Scandinavian countries allowing all stakeholders as peers to participate in the design process, which has exhibited prominent significance and value in eHealth system development (Kildea et al., 2019; Smith et al., 2020), e.g., the co-design of a patient portal involves patients and health care providers.

In contrast to a single-stakeholder design, co-design of stakeholders is fairer, providing all participants an equal opportunity to voice opinions and provide feedback. As each stakeholder has his/her own knowledge, interest, business policy and goal, hardware and software resources which are not exactly consistent or compatible with each other, co-design helps equalize the stakeholder-side knowledge and interest balance, unifying regulations and standards, and moderate design requirements and goals. Such a user-centered holistic approach also enhances the network value and utility with the increasing number of users. The impact of the IT systems can be more holistically assessed by all stakeholders (backed by evaluation using later field study and interviews). Moreover, the risk assessment is more comprehensive due to the

| Stakeholders | Responsibilities | Role in co-design |
|---|---|---|
| Coffee importer | Import green coffee | Describe requirements for blockchain model and desired characteristics + pilot test partner |
| Coffee exporter | Mill parchment coffee (hulling, sorting, quality and testing, incl. cupping. Prepare green coffee for export. | Interviewee + field visit + pilot test partner |
| Coffee roaster company | Roast green coffee. Pack roasted coffee for retail | Describe requirements for blockchain model and desired characteristics + pilot test partner |
| NGOs | Advisory services and auditing | none |
| Small-scale farmers | Produce, harvest, wetmill and dry coffee. Pack parchment coffee in jute bags | Interviewee + field visit + pilot test partner (6 farmers) |
| Cooperatives | buy parchment coffee. Control quality and characteristics of coffee. Sell coffee to exporters. Support farmers with extension services. | Interviewee + field visit + pilot test partner |
| Academic partners | Technical expertise | Coordination and management |

**Table 1.** Stakeholders and responsibilities.

involvement of all stakeholders and improved realistic estimations of risks and their impacts. Overall, co-design among stakeholders, featured by comprehensive intelligence, balanced benefits, in-time feedback and adjustment, unified regulations, and strengthened utility, is an integral component of sustainable quality improvement in our Good Coffee use case.

Multiple stakeholders are involved in the project and are influencing its design as depicted in Table 1 and visualized in Figure 2. On the industrial side, it includes small-scale farmers (in our case in Colombia), cooperatives, middlemen and logistics providers, exporters, warehouses, coffee roasting companies, retailers, final consumers, and Non-Governmental Organizations (NGOs), e.g., as concerns sustainability standards. Specifically, COWI (advisory firm and project coordinator), cooperative (name) and Federacion Nacional de Café (FNC) (exporter), Peter Larsen Kaffe (roaster) as representatives contribute to the co-design process and provide assessments from field studies and conduct interviews from the industrial perspective. On the academic side, project partners include Department of Computer Science, Department of Geoscience and Natural Resource Management at Copenhagen University, European Blockchain Center at IT University of Copenhagen, and Chalmers University of Technology. The academic partners provide extensive theoretical and technical guidance for sustainable Good Coffee requirement analysis and design, natural resource management, and prototype development including the here described smart contracts.

The major stakeholders joined in iterative design and status meetings with brainstorming, feedback, and acceptance sessions. Weekly developer meetings, monthly group meetings, and quarterly status meetings were held for progress reviews. The meetings were facilitated partially as role-plays for facilitate knowledge exchange. Representatives played their roles in the end-to-end supply chain process and provided early and regular feedback on the accuracy and correctness of our approach. We set the next plans and goals adaptively based on the feedback after each meeting. In the beginning, a pen-and-paper design prototype was used and replaced by a mock-up or an early technical prototype in later stages. It allowed participants to get an overview of the status and also allowed for influencing the next design and implementation tests. The decision phases of the meetings were in the form of the nominal group technique. From design theoretic perspective, our chosen approach is close to the distributed participatory design methodology.

The approach is aligned to conceptual design framework in (Groschopf et al., 2021) in which stakeholders are mapped to digital entities.

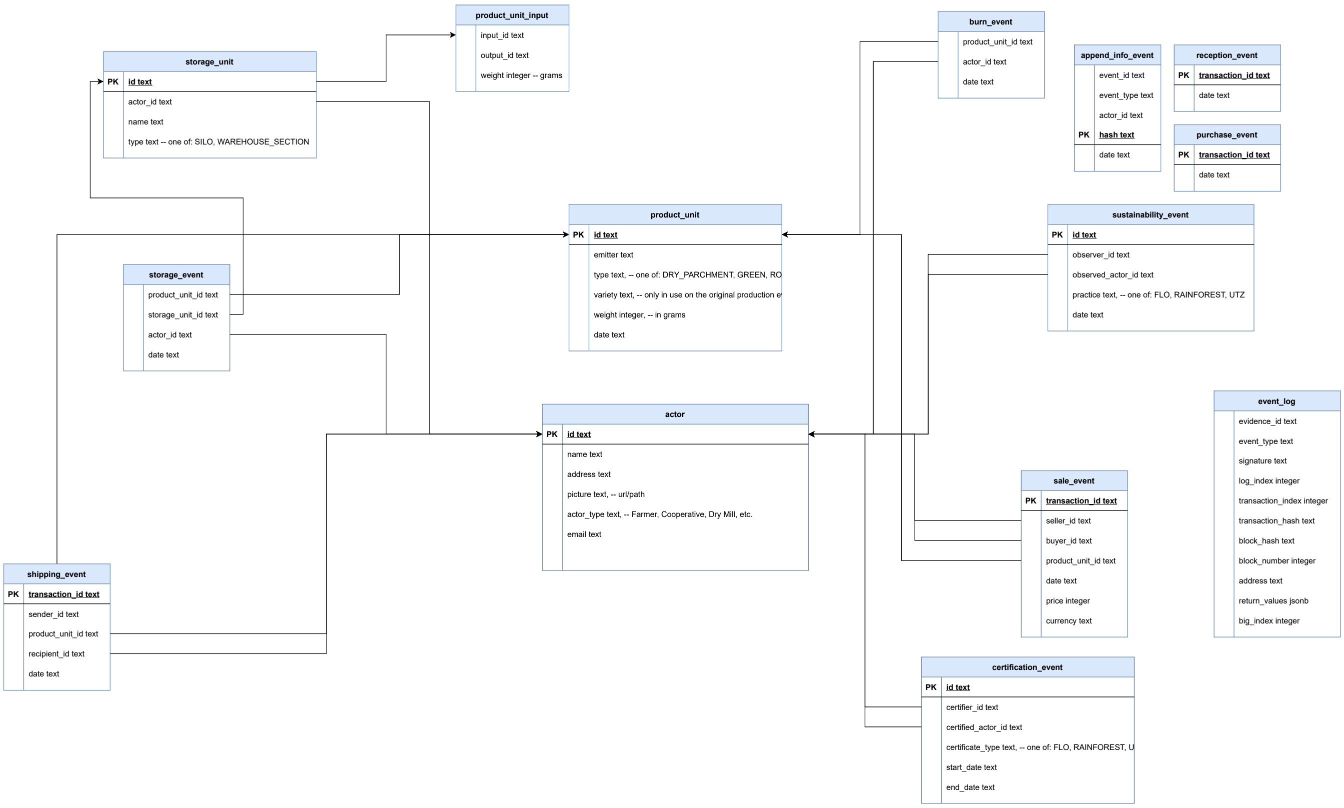


**Figure 3.** ER-Diagram of the core data model.

## 3.3 Software design and implementation

### 3.3.1 Database-driven design

Modeling and implementing the data types and model in standard relational database, PostgreSQL, allowing refinement iterations and visual and interactive discussion of Entity-Relationship (ER) model as depicted in Figure 3. After model stabilized transferring it to smart contracts.

Good Coffee involves different data types, including value types, reference types, and mapping types. Value types include booleans (e.g., identify if a transaction is canceled or confirmed), integers (for product weight), fixed-size byte arrays (for product identifier or file hash value), addresses (for account addresses), and dynamically-sized byte array (for stakeholder name, location, etc.). Reference types include data location, arrays, and struct. Data location refers to where the data is stored, including *memory* and *storage*. Specifically, *memory* is to hold temporary values that are erased between (external) function calls. In contrast, *storage* is where all the contract state variables reside, which is persistent between function calls and more expensive to use. In Good Coffee, the addresses of entities are saved as state variables. Different types of variables can be grouped into a custom defined type called struct. For example, the certificate struct contains *certifier* (address type), *startDate*, *endDate*, and *kind* with integer type, and *isDefined* with boolean type. Finally, mapping types are like hash tables with key-value pairs, in which the value can be found quickly using the key. For example, we declare a private mapping variable *_agents* as *mapping (address ⇒ Actor) private _agents*, then we can use *_agents[address]* to find the corresponding Actor if the agent has been registered.

### 3.3.2 Technology stack

- **NodeJS**: Runtime environment for building web-based JavaScript applications.

- **ReactJS**: Front-end framework used for the entire front-end of the platform. Joined by extra JavaScript NPM packages for extra functionality.
- **Express**: Used for developing a centralised server that responses to `GET` and `POST` requests from the client (React application).
- **PostgreSQL**: Centralised relational database management system (DBMS) used for storing the actual data objects submitted by the agents.
- **Truffle**: Development environment used for running the decentralised application (Dapp) on the Ethereum Virtual Machine.
- **Solidity**: Programming language used for writing the smart contract that the platform's users/agents interact with.

#### 3.3.3 User interface

In Good Coffee, users are categorized into different types, e.g., farmer, transporter, and certifier, in which rich graphical user interfaces are provided for different system functionalities/events. The major user interfaces in Good Coffee are as follows.

- Registration Interface: each user can register in Good Coffee as agent with specific information, such as agent name, type, email address, organization (if he is affiliated with), location, and profile picture. Such information can be viewed by users in the agent profile interface.
- Product Issuance Interface: An authenticated farmer can issue and add a new product to the inventory with detailed product type, weight, coffee variety, and a picture (optional). After confirming the transaction with Metamask, the new product in the shape of *dry parchment* will appear in the inventory table of the farmer.
- Product Splitting Interface: Authenticated agents are able to split a single product unit into multiple weights and product units. For example, 250kg can be split into 2 product units with 100kg per unit and 5 product units with 10kg per unit. After successful split, the old product item will be replaced by the new ones in the *Inventory* table.
- Product Shipment Interface: Actors whose products have been purchased by other agents can perform the shipping process. The shipment interface is where the agents can choose transporter and recipient. Besides, evidence document and extra information can be added optionally, such as the transporter agreement and key-value pairs. For example, the *Number plate* can be the key, and the corresponding value can be CO234BOG.
- Product Repacking Interface: Repackaging can be performed on one or multiple units. When clicking on the Bundle button, the agent can input the weight of output and choose a type of output. Similarly, a picture is optional to be uploaded for better visibility.
- Product Sale Interface: The owner of a product can initiate an *offer* to a specific agent and start a selling process. The owner can specify the *recipient*, *amount*, and *currency* in the sale interface.
- Purchasing Interface: Anyone can purchase products that have previously offered to him but do not already belong to him. After clicking on the Purchase button, there is a Modal for confirming the purchase.
- Receipt Confirmation Interface: When the products have been shipped to the recipient, the recipient can confirm the receipt of each shipment at a time. After approving it, the products will be transferred to the *Inventory* of the recipient.

- Evidence Adding Interface: Authenticated agents can add evidence for any event that happened and displayed under the *Event* view. For instance, a PDF file of invoice can be added as transport documentation evidence to ensure the validity of the shipment event. It is worth noting that the evidence added by agents other than the one who submitted the event has a higher weight.
- Product Storage Interface: Before products are transformed, e.g., dry milling and roasting, the product items need to be stored in a storage unit such as a silo. The agent must select the storage unit where they want to store the product, e.g., Silo #1, Silo #2, and Warehouse #D7.
- Processing/Transformation Interface: The transformation is applied to a storage unit instead of a product unit. The interface will ask for the output quantity and type. After completion, the newly issued products can be found under the *Inventory* list of the agent.
- Actor, Events, and Product View Interfaces. These interfaces show the detailed information about agents, events, and products. The agent view consists of a sidebar and three different tabs: *Profile*, *Receive* (only when the agent is authenticated as the viewed agent), and *Inventory*. The sidebar displays the agent's profile photo or logo, name, type, and profile (brief description, statistics with the number of products processed and sold, certificates, and location). The event view includes the five parts as described in Section 3.3.4. The product view also has a layout including a sidebar on the left for identifying the product ID, type, weight, and producer. The content on the right shows the product details, including an image of the product, description, and a *Eligible for* section with some certification names. At the bottom, there is a *Provenance* section illustrating a map to display the location of each of the corresponding farmers. For increased transparency, the farmer payments, the coffee unit composition (in the form of pie chart), the sustainability parameters, and the product custody are also displayed.

### 3.3.4 Migrating shared information to Ethereum

Based on the analyzed ER-Model, we design the Coffeebrain smart contract, which implements the business logic as the central part of the system and run on the Ethereum Virtual Machine (EVM). It maintains the system's state and ensures that all operations are valid. Specifically, Coffeebrain includes the following elements.

- Variables. Each specified variable consists of a name and a data type, as introduced in Section 3.3.1. There are four types of visibilities for variables: public, private, external, and internal, where the default visibility is *internal*. For state variables, *external* is not possible. In Coffeebrain smart contract, we declare ten custom defined variables in the type of struct, including File, Product, Transaction, Silo, Certificate. Two state variables with address types are defined for the owner and the burning account, respectively. Moreover, we define six key-value mappings.
- Functions. Functions are executable units of code implementing the system functionalities. Each function has a function name and body where the functionality logic is defined. Functions share the same four types of visibilities with variables, but the default is public. In Coffeebrain, we define two private functions to update the product index set in terms of adding and deleting a product, respectively. Another two private functions are defined to add new products to the silo and pop certain weight from the silo, respectively. Moreover, there are twelve public API functions for user registration, product mint, shipment, selling, purchase, and other functionalities.
- Modifiers. Modifiers are used to automatically check a condition before executing certain functions. We define two modifiers named *authenticated* and *registered*. The former is to check if the message sender is registered and the latter with an input agent address is to check if the agent exists.

- Events. Events are convenience interfaces with the EVM logging facilities. All events are recorded on the ledger, sorted by their date in a descending order. Each event has five parameters: its ID/transaction hash, type (e.g., splitting, purchase, transformation, etc.), emitter, the block hash that the event was mined in and the index. Concretely, the ID field provides a link to the full event view and details. The emitter field displays a link to the agent view of the agent who has emitted the event, for an efficient navigation between views. We define twelve events in Coffeebrain and each is emitted in a public API function.

### 3.4 Empirical investigation

We conducted an empirical investigation of the model across the coffee supply chain. The investigation involved all actors from farmers to roaster, but did not involve final consumers. Due to the Covid-19 pandemic, part of the field test in Colombia had to be conducted as remote evaluation assisted through 2 in-field assistants, who worked with the farmers. The practical and "real-world" implications and outcomes of implementing and empirically testing a blockchain-as-a-service solution in a coffee supply chain is thoroughly analyzed and discussed in (Bager et al., 2022), which is also part of the Good Coffee project. Technical challenges particularly relates to data access and formatting, as not all actors involved in the supply chain currently store data in a digital format. Further, many of the smallholder farmers in the case supply chain have limited internet connectivity and access; a problem also reported by (Mehrabi et al., 2021). As such, while the initial setup, specifications, and programming of the model can accommodate the requirements of all of the actors involved in the study. Several complications arise when moving from theoretical presuppositions about the model and its functionalities to actual empirical investigation *in-situ*.

## 4 DISCUSSION

### 4.1 Evaluation

From the perspective of desired properties, our event-based blockchain solution for general commodity supply chains achieves decentralization, product transparency, provenance, sustainability, and processing automation. Such properties are more comprehensive than the basic transparency, decentralization, and automation considered in prior blockchain-based SCM studies (Dietrich et al., 2021). From the perspective of product structure, we deal with complex parts rather than single parts. Following the definition in (Dietrich et al., 2021), complex parts means that they can experience changes in their modular composition (e.g., splitting and merging in our model), while single parts means that there is no change in their modular composition, but transformation exists. Existing works like (Sun et al., 2019; Zhang et al., 2020; Kamath, 2018) are all limited to the single-part product structure. A holistic mapping of manufacturing supply chains should contain the mapping of raw materials, intermediate products, final products, and different events. However, as analyzed in (Dietrich et al., 2021), such a holistic mapping is not fully realized in the references (Malik et al., 2018; Wang et al., 2019a; Reimers et al., 2019). In contrast, our study enables the holistic mapping. Finally, from the perspective of implementation, some of the literature (Sun et al., 2019; Salah et al., 2019; Tönnissen and Teuteberg, 2018) only develop a system framework or concept without a detailed implementation. Besides a comprehensive system framework, we also illustrate the crucial components in implementation, including data type and model, smart contracts, technology stack, and user interface. For our Good Coffee use case, we develop a Good Coffee platform realizing the core business logic for all actors in the end-to-end supply chain.[15]

[15] The source code of the smart contract can be found at https://codeberg.org/juanhebert/coffeebrain-ethereum.

Our empirical investigation demonstrates that while blockchain-as-a-service applications can theoretically and conceptually address many problems related to sustainable commodity supply chains, actual ”real-world” implementation across diverse systems and actors carry several practical problems; from missing data and technical misalignment to lacking internet connectivity and device access. These aspects are thoroughly discussed in (Bager et al., 2022). Further, development and implementation costs are currently prohibitively high, making blockchain-as-a-service solutions ineffective from an economic perspective. However, with increased digitalization and reduced costs of blockchain systems, these barriers will likely reduce in coming years.

### 4.2 Retrospective

We present a blockchain solution for decentralized, certified, and traceable SCM Such a solution is founded with theoretical event-based methodologies, an event-driven architecture, and functional models. Furthermore, we present a Good Coffee case study with a full implementation over the Ethereum blockchain, covering all processing stages of coffee which is trackable and traceable with a visual web application. Our pre-project analysis and design involves multiple stakeholders, which aggregates intelligence, inspiration, and constructive feedback from different project stakeholders. Such co-design methodology makes the use of each stakeholder’s expertise and yields a comprehensive, professional, and fair assessment.

Li et al. (Li et al., 2017) proposed a framework which integrated event-based and transaction-based architectures for supply chains. The architecture of the framework is based on a dynamic hybrid peer-to-peer network and a private/public blockchain data model. It involves three types of components: index server, peers, and administrative nodes. The peer application consists of three tiers: a presentation layer, a middle layer, and a local database. There are two kinds of blockchain ledgers implemented: a semi-public ledger and a private ledger. Each ledger has specific types of events. For example, the semi-public ledger consists of monitoring events to locate the physical location of a given truck. The private ledger consists of shipment information and custody events. Such a hybrid framework makes the most of the salient transparency and traceability features of the public blockchain meanwhile addressing the potential privacy concerns of trading partners with the private blockchain. In comparison, privacy protection issues are exactly what we neglected in our public-chain design.

In contrast, our project on the event-based modeling using REALISTIC, in particular, to support product transformations. We share the same experience in the difficulty of balancing conflicting requirements (e.g., transparency vs. confidentiality).

### 4.3 Conclusion

Blockchain technology has exhibited promise in constructing decentralized and trustworthy SCM systems with secure values due to its inherent trust and immutability. Certified, sustainable or specialty coffee all entails complex supply chains, including a multitude of stakeholders and customers, who are willing to pay premium prices for high-quality products. In this paper, we proposed a novel event-based methodology REALISTIC and an event-driven system architecture for systematic modeling of supply chain networks and tamper-proof product tracking, respectively. Particularly, we presented the Good Coffee case study for the event-based blockchain solution, covering all stages of coffee supply chain and enabling real-time coffee track and trace from coffee cherry to coffee cup. An open-source prototype is fully implemented to support and validate our modeling and design in a field study.

## ACKNOWLEDGEMENTS

Simon Bager, Boris Düdder, and Fritz Henglein acknowledge funding from the COWI Foundation grant “Sustainable supply chains for bio-based products — Using blockchain technology to accelerate sustainability, transparency and traceability in bio-based value chains.” Simon Bager acknowledges funding from the European Union’s Horizon 2020 Research and Innovation Programme under the Marie Sklodowska-Curie grant agreement 765408.

## DATA AVAILABILITY STATEMENT

The datasets and prototypical implementation for this study can be found in the GitHub repository `https://github.com/diku-dk/coffeechain`.

## REFERENCES

Andersen, J., Elsborg, E., Henglein, F., Simonsen, J. G., and Stefansen, C. (2006). Compositional specification of commercial contracts. *International Journal on Software Tools for Technology Transfer (STTT)* 8, 485–516

Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., et al. (2018). Hyperledger Fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference* (ACM), 30

Bager, S. L. (2021). *Delivering Zero Deforestation: How Governance Interventions in Agro-Food Commodity Supply Chains can Foster Sustainable Land Use*. Phd thesis, Université catholique de Louvain

Bager, S. L. and Lambin, E. F. (2020). Sustainability strategies by companies in the global coffee sector. *Business Strategy and the Environment* , bse.2596doi:10.1002/bse.2596

Bager, S. L., Persson, U. M., and dos Reis, T. N. (2021). Eighty-six EU policy options for reducing imported deforestation. *One Earth* 4, 289–306. doi:10.1016/j.oneear.2021.01.011

Bager, S. L., Singh, C., and Persson, U. M. (2022). Blockchain is not a silver bullet for supply chain sustainability. *Current Research in Environmental Sustainability*

[Dataset] Brown, R. G., Carlyle, J., Grigg, I., and Hearn, M. (2016a). Corda: A distributed ledger. R3 CEV technical whitepaper

[Dataset] Brown, R. G., Carlyle, J., Grigg, I., and Hearn, M. (2016b). Corda: An introduction. R3 CEV whitepaper

[Dataset] Buterin, V. (2014). A next-generation smart contract and decentralized application platform. White paper

COWI, Ecofys, and Milieu (2018). *Feasibility study on options to step up EU action against deforestation*. Tech. rep., Brussels, Belgium

Dauvergne, P. and Lister, J. (2012). Big brand sustainability: Governance prospects and environmental limits. *Global Environmental Change* 22, 36–45. doi:10.1016/j.gloenvcha.2011.10.007

David, J. (1997). Three “events” that define an rea methodology for systems analysis, design, and implementation. In *Proceedings of the Annual Meeting of the American Accounting Association*

DeFries, R. S., Fanzo, J., Mondal, P., Remans, R., and Wood, S. A. (2017). Is voluntary certification of tropical agricultural commodities achieving sustainability goals for small-scale producers? A review of the evidence. *Environmental Research Letters* 12. doi:10.1088/1748-9326/aa625e

Dietrich, F., Ge, Y., Turgut, A., Louw, L., and Palm, D. (2021). Review and analysis of blockchain projects in supply chain management. *Procedia Computer Science* 180, 724–733

Düdder, B., Fomin, V., Gürpinar, T., Henke, M., Janavicene, V., Matulevicius, R., et al. (2020). Interdisciplinary Blockchain Education: Utilizing Blockchain Technology from Various Perspectives. *Frontiers in Blockchain* doi:10.3389/fbloc.2020.578022

Düdder, B. and Ross, O. (2017). Timber tracking: reducing complexity of due diligence by using blockchain technology (position paper). In *Proceedings of 2nd Workshop on Managed Complexity 2017*, ed. B. Johansson (CEUR-WS), Ceur Workshop Proceedings. Available at SSRN 3015219

Garrett, R. D., Levy, S. A., Gollnow, F., Hodel, L., and Rueda, X. (2021). Have food supply chain policies improved forest conservation and rural livelihoods? A systematic review. *Environmental Research Letters* 16, 033002. doi:10.1088/1748-9326/abe0ed

Godfray, H. C. J., Beddington, J. R., Crute, I. R., Haddad, L., Lawrence, D., Muir, J. F., et al. (2010). Food security: the challenge of feeding 9 billion people. *Science (New York, N.Y.)* 327, 812–8. doi:10.1126/science.1185383

Grabs, J. (2017). The rise of buyer-driven sustainability governance: Emerging trends in the global coffee sector

Groschopf, W., Dobrovnik, M., and Herneth, C. (2021). Smart contracts for sustainable supply chain management: Conceptual frameworks for supply chain maturity evaluation and smart contract sustainability assessment. *Frontiers in Blockchain* 4, 12. doi:10.3389/fbloc.2021.506436

Henglein, F. (2018). Blockchain deconstructed (abstract). In *Proc. 2nd Symposium on Distributed Ledger Technology* (Gold Coast, Australia). Full paper in preparation

Henglein, F., Larsen, K. F., Simonsen, J. G., and Stefansen, C. (2007). Compositional contract specification for REA. In *First Workshop on Formal Languages and Analysis of Contract-Oriented Software*. Invited contribution

Ikhwana, A. (2018). Supply chain management of coffee commodities. In *MATEC Web of Conferences* (EDP Sciences), vol. 197, 14003

IPBES (2019). *The global assessment report on summury on policymakers of the IPBES global asessment report on biodiversity and ecosystem services* (Bonn, Germany: IPBES secretariat). doi:https://doi.org/10.5281/zenodo.3553579

IPCC (2019). *Climate Change and Land: an IPCC special report on climate change, desertification, land degradation, sustainable land management, food security, and greenhouse gas fluxes in terrestrial ecosystems*. Tech. rep., Intergovernmental Panel on Climate Change

Jacquet, M. (2003). *REAlistic—En REA model uden perspektiver*. Master's thesis, IT University of Copenhagen

Kamath, R. (2018). Food traceability on blockchain: Walmart's pork and mango pilots with ibm. *The Journal of the British Blockchain Association* 1, 3712

Kildea, J., Battista, J., Cabral, B., Hendren, L., Herrera, D., Hijal, T., et al. (2019). Design and development of a person-centered patient portal using participatory stakeholder co-design. *Journal of medical Internet research* 21, e11371

Lambin, E. F., Gibbs, H. K., Heilmayr, R., Carlson, K. M., Fleck, L. C., Garrett, R. D., et al. (2018). The role of supply-chain initiatives in reducing deforestation. *Nature Climate Change* 8, 109–116. doi:10.1038/s41558-017-0061-1

Lambin, E. F. and Thorlakson, T. (2018). Sustainability Standards: Interactions Between Private Actors, Civil Society, and Governments. *Annual Review of Environment and Resources* 43, 369–393. doi:10.1146/annurev-environ-102017-025931

Li, Z., Wu, H., King, B., Miled, Z. B., Wassick, J., and Tazelaar, J. (2017). On the integration of event-based and transaction-based architectures for supply chains. In *2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW)* (IEEE), 376–382

Malik, S., Kanhere, S. S., and Jurdak, R. (2018). Productchain: Scalable blockchain framework to support provenance in supply chains. In *2018 IEEE 17th International Symposium on Network Computing and*

*Applications (NCA)* (IEEE), 1–10

McCarthy, W. E. (1982). The REA accounting model: A generalized framework for accounting systems in a shared data environment. *The Accounting Review* LVII, 554–578

Meemken, E.-M. M. (2020). Do smallholder farmers benefit from sustainability standards? A systematic review and meta-analysis. *Global Food Security* 26, 100373. doi:10.1016/j.gfs.2020.100373

Mehrabi, Z., McDowell, M. J., Ricciardi, V., Levers, C., Martinez, J. D., Mehrabi, N., et al. (2021). The global divide in data-driven farming. *Nature Sustainability* 4, 154–160. doi:10.1038/s41893-020-00631-0

[Dataset] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system

Ponte, S. (2019). *Business, Power & Sustainability* (London, UK: Zed Books)

Reimers, T., Leber, F., and Lechner, U. (2019). Integration of blockchain and internet of things in a car supply chain. In *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)* (IEEE), 146–151

Salah, K., Nizamuddin, N., Jayaraman, R., and Omar, M. (2019). Blockchain-based soybean traceability in agricultural supply chain. *IEEE Access* 7, 73295–73305

Smith, I., Hicks, C., and McGovern, T. (2020). Adapting lean methods to facilitate stakeholder engagement and co-design in healthcare. *Bmj* 368

Sun, W., Zhu, X., Zhou, T., Su, Y., and Mo, B. (2019). Application of blockchain and rfid in anti-counterfeiting traceability of liquor. In *2019 IEEE 5th International Conference on Computer and Communications (ICCC)* (IEEE), 1248–1251

Tapscott, D. and Tapscott, A. (2017). How blockchain will change organizations. *MIT Sloan Management Review* 58, 10

Thorlakson, T., de Zegher, J. F., and Lambin, E. F. (2018). Companies' contribution to sustainability through global supply chains. *Proceedings of the National Academy of Sciences* 115, 2072–2077. doi:10.1073/pnas.1716695115

Tönnissen, S. and Teuteberg, F. (2018). Using blockchain technology for business processes in purchasing-concept and case study-based evidence. In *International Conference on Business Information Systems* (Springer), 253–264

Torres Garcia, J.-M. (2020). *Algebraic Resource Accounting for Transfers and Transformations*. Master's thesis in computer science, Department of Computer Science, University of Copenhagen (DIKU)

Wang, S., Li, D., Zhang, Y., and Chen, J. (2019a). Smart contract-based product traceability system in the supply chain scenario. *IEEE Access* 7, 115122–115133

Wang, Y., Han, J. H., and Beynon-Davies, P. (2019b). Understanding blockchain technology for future supply chains: a systematic literature review and research agenda. *Supply Chain Management: An International Journal*

[Dataset] Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. Homestead Draft (Yellow Paper)

Zhang, X., Sun, P., Xu, J., Wang, X., Yu, J., Zhao, Z., et al. (2020). Blockchain-based safety management system for the grain supply chain. *IEEE Access* 8, 36398–36410