Decentralized Supply Chain Management in the Automotive Industry

Team: 8
Members: Anirudh Palutla, Sumanth Balaji, Shanmukh Karra
Mentors: Sambhav Solanki
Submitted as Part of Course Distributing Trust and Blockchains (Monsoon 2021)
by Dr Sujit Prakash Gujar.

Abstract

Supply chain networks in the automotive industry deal with numerous participants and parts on multiple tiers that are spread across several geographical locations. Along with immense complexity, supply chains often lack transparency and responsiveness. This translates to vulnerability to counterfeiting and difficulty in resolving issues. To address this, we propose a decentralized supply chain traceability application that uses input at every step of the way to ensure accountability and quality.

1 Introduction

1.1 Problem Statement

Car manufacturers are known as OEMs (original equipment manufacturers). However, they do not produce vehicle parts internally, and instead rely on several suppliers from various geographical regions to obtain the parts used to manufacture the vehicles. These suppliers form the supply chain networks that act as the backbone of the industry. (Miehle et al., 2019)

These networks are intricate systems comprised of parties such as OEMs, suppliers, and logistics service providers(LSPs). These parties exchange products, information, and money flowing in a hierarchical manner based on tiers. Suppliers are categorized into these tiers, wherein a tier-1 supplier is a first rank supplier, who incorporates parts from a tier-2 supplier, and so on.

To manage and keep track of this supply chain, we propose a decentralized supply chain traceability application.

1.2 Motivation

Due to the pandemic, supply chain networks across the world have been strained, and the automobile industry has been no exception. These issues have also revealed how time-critical the automobile supply chains are. This makes the need for detection and rectification of issues between parties in supply chain very pertinent.

Counterfeiting products is a problem that has plagued the automotive supply chain industry for decades, and is estimated at several billion dollars worth of spare parts. Counterfeit spare parts often have poor quality, as they enter directly into the supply chain and impact the OEMs. It also leads to dissatisfied customers and affects trust in the brand. Hence, there is a need to ensure complete traceability of products in the supply chain.

1.3 Beneficiaries

The main beneficiaries of this system are OEMs, suppliers, and logistics service providers. In extension this system of tracking improves the rectifying of customer product issues hence improving customer experience.

2 Requirement of Blockchain

Each party in the supply chain uses a local database representation for tracking and monitoring of products, leading to unavailable and potentially inconsistent information on the event history of a part. Therefore, rectifications and recalls often requires significant human intervention due to no common master data, compatibility issues, and missing standards. Detection of defective and counterfeit parts is inefficient, expensive, and error-prone.

To combat this, we make use of a blockchain. Blockchain technology records and shares data using a distributed ledger, which provides a single source of truth improving the speed and responsiveness of supply chain networks. (Sharma et al., 2019)

3 Solution

3.1 Approach

The immutable, append-only properties of the blockchain allow us to create an accountable supply-chain management system. We propose a decentralized ledger approach to tracking each part of the manufacturing process throughout the supply chain starting from the supplier of the raw materials all the way to the original equipment manufacturer's final product including logistic service providers in between. Each party will be responsible for updating the information about their respective products.

3.1.1 Automotive Supply Chain

The automotive supply chain is split into a hierarchical supply chain consisting of separate tiers with tier 0 being the original equipment manufacturer (OEM). Tier 1 typically consists of system/module suppliers, tier 2 consists of component suppliers and tier 3 consists of the raw material or parts suppliers. Each tier supplies primarily to the tier directly above it (consider tier n to be above tier $n + 1$), but can also deliver to a higher tier directly. In our model, a party at tier n can only place orders to parties at tier n or greater.

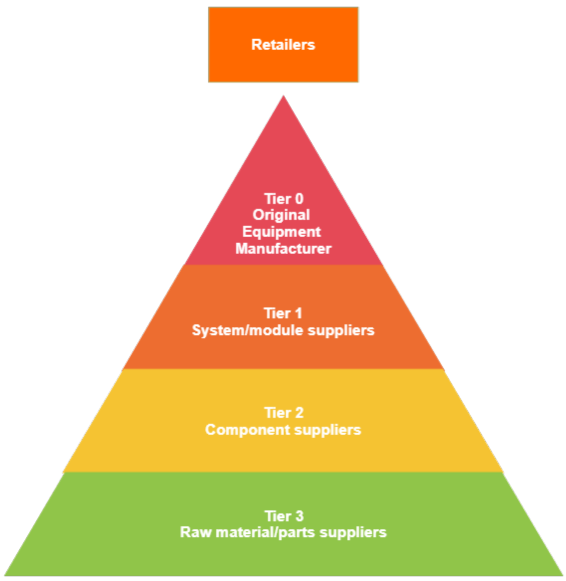

Figure 1: Hierarchical Supply Chain

A **party** in the supply chain refers to a certain manufacturer, supplier or logistics service provider (LSP). Every party is represented by a specific wallet address. This approach can be generalised for a party to be represented by a unique identifier and a set of wallet addresses all representing the same party, but each party is considered to be represented by a single wallet address without loss of generality for the scope of this paper.

3.1.2 Components and Digital Twin

A **component** is any part or a product of multiple parts in the manufacturing process. For example, a box of screws could be identified as a component and an engine could also be identified as a component. These are two different component types and there can be arbitrarily many component types. A list of component types is prepared for the specific supply chain in question and is available to all parties.

Components are sent between two parties in the supply chain. Each of these components can be identified by using a unique identifier linked specifically to the physical object. Such an identifier is often referred to as a **"digital twin"** of the physical component. This identifier is unique to each component. In the example cited earlier, each engine piece would have its own digital twin and each box of screws would have its own digital twin. This digital twin is used to identify specific components exchanged between parties and can be used for tracking of each part of the finished product.

Component and **digital twin** are used interchangeably here on out to refer to a specific physical product and its linked ID.

An important functionality of components is that multiple components can be combined together to form a single component. This new component will have a new unique ID and the IDs of its sub-components are stored along with it. This allows for arbitrary combination of parts at different levels. This combination can be done by any party possessing the requisite component parts. This combination is the equivalent of building an engine from its requisite metal alloys, aluminium sheets, etc. or a multimedia system being built from its requisite parts which could be speakers, digital interface, etc.

3.1.3 Component Types

Each component belongs to a certain **component type**. A component type has a list of prerequisite components to build it, and is identified by a unique identifier. A set of components can only be combined to form a new component if the correct amounts of them are available.

Referencing the example above, a component type "multimedia system" might have prerequisite list of 4 speakers, 1 digital interface, 20 copper wires, etc.

3.2 Orders

An **order** is a purchase of a particular component types in a specified quantity by a given party. The order is placed from one party to another party, referred to as the *buyer* and *seller*. Each order only contains products that are manufactured by the seller. Orders are represented by unique identifiers assigned to them upon creation.

3.2.1 Logistic Service Providers and Shipments

The third type of party in the supply chain apart from the manufacturers and suppliers are the LSPs. These are responsible for the delivery of orders from one party to another. LSPs are identified by a similar party identifier not unlike the manufacturers/suppliers. From here on out, **non-delivery parties (NDPs)** will be used to refer to the manufacturers and suppliers exclusively and **delivery parties (DPs)** will be used to refer to LSPs.

DPs deliver orders from one NDP to another. Multiple orders can be packaged into a **shipment**. The DP uses and updates the status of this shipment in order to keep track of orders. *Shipments* have unique digital identifiers similar to components and orders. These IDs can be used to physically verify the reception of the shipment.

3.3 Architecture

The proposed model can be implemented as a smart contract. Each party can directly interact with the deployed smart contract using their respective assigned addresses. The class diagram of a codified version of the proposed architecture is illustrated in Figure 2.

3.3.1 Parties

Each party is represented by a party object. The contract has a global list of parties identified by their unique addresses.

The party object contains the following information:

- *Party address*
- Name
- Role [Delivery party or non-delivery party]
- Tier [0, 1, 2 or 3]
- Location (physical address)

Italics are being used to represent the primary key of the model.

3.3.2 Components & Component Types

There is a global mapping list of components which are identified by their component ID. There is another constant reference list of component types.

A component contains the following information:

- *Component ID*
- Component Type ID
- List of sub-component IDs along their quantities
- Manufacturer address

The manufacturer is the party that combines sub-components to form the component or supplies the raw product in case the component has no sub-component.

The component type object contains the following information:

- *Component Type ID*
- Component Type Name
- List of prerequisite sub-component type IDs along with their quantities

Note that the base-level component types have an empty list of prerequisites.

3.3.3 Orders & Shipments

There are global lists of orders and of shipments each identified by their own unique IDs.

Orders can only be placed by a party of tier n to a party of tier n or greater. The order object contains the following information:

- *Order ID*
- List of component type IDs along with their quantities

- List of component IDs in their respective quantities
- Shipment ID
- Buyer address
- Seller address
- Status [placed, confirmed, shipped, finished or cancelled]

The list of the specific component IDs is left empty and is left for the seller to fulfill. The buyer places the order with only the component type IDs along with their respective quantities. The shipment ID is also left empty and is filled once the seller ships the item through a LSP.

The shipment object contains the following information:

- *Shipment ID*
- List of order IDs
- Buyer address
- Seller address
- Shipping location
- LSP address
- Status [placed, confirmed, shipped, finished or cancelled]

Since each transaction is timestamped on the blockchain, it is trivial to track an order along with its respective shipping times.

3.3.4 Deployment

The entire model can be written into a smart contract. The model should be deployed on a permissioned-blockchain network such as Hyperledger Fabric. Alternatively, the contract could also be deployed to the Ethereum network with additional permission-gating contracts.

3.3.5 Example

Consider an order being placed by *party1* to *party2*. *party1* initially fills in the list of component type IDs with their respective quantities and submits the order. The list of the exact component IDs is left empty and so is the shipment ID. Once *party2* is ready to fulfill the order, they fill in the list of component IDs with the exact component IDs of each *component* object and the shipment ID is filled during shipping.

For example, suppose *party1* orders `1 engine` and `4 seats`, each of them being component types. The order initially looks like this (represented as a tuple):

```
(1, { engine: 1, seat: 4}, NULL,
NULL, party1, party2, PLACED)
```

party2 then gathers the necessary *engine* with ID `sefscji` and seats with IDs `wasicn`, `uisass`, `iaspvs`, `popsaf`. They then fill in the order:

```
(1, { engine: 1, seat: 4},
    { engine: [ sefscji ],
      seat: [ sefscji, wasicn,
        uisass, iaspvs, popsaf ]
    },
NULL, party1, party2, CONFIRMED)
```

This can then be shipped by the LSP *shipper1* to the respective location as part of a new shipment. The shipment will look like this:

```
(2131, [ 1 ], party1, party2,
"1212, Road no. 10, Gachibowli,
Hyderabad - 500032", shipper1)
```

This example has been illustrated in Figure 3.

4 Analysis

4.1 Security & Privacy

- As parties generate their own private and public keys, which are used to sign data and verify signatures respectively, security is ensured so long as the private key is not leaked.
- The use of Blockchain technology ensures transparency as information is available to the public in the form of logs. This allows for verification of different steps in the supply chain.
- Since the system is deployed on a closed network, tracing malicious parties within the supply chain is relatively simple. (Fraga-Lamas and Fernández-Caramés, 2019)

4.2 Cost of Operation

On the Hyperledger Fabric, there is no cost of operation except for deployment. On the Ethereum blockchain, however, each transaction will cost some gas fees. Since the management system benefits everyone in the supply chain, the costs can be

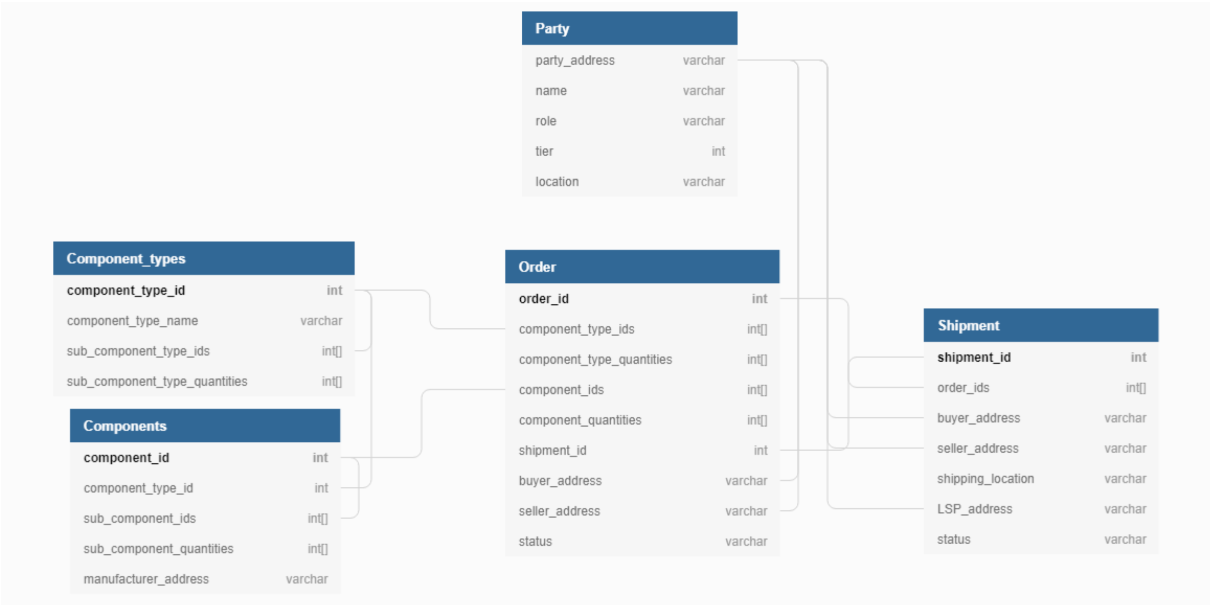

Figure 2: Architecture (This figure is a codified implementation of the model)

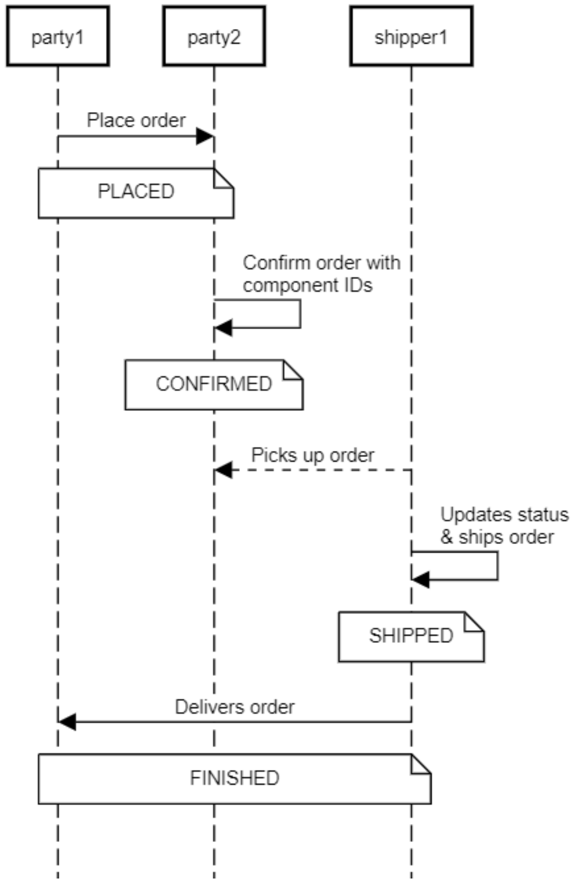

Figure 3: Example Flow

shouldered by each organisation that is part of the supply chain.

4.3 Feasibility

Implementing the solution on a large scale would require overhauling existing supply-chain management systems within the automotive industry. The current systems rely on centralized databases kept by each party, and overhauling it would require a certain amount of effort. However, implementing this system for a newly set-up supply chain is relatively easy as each step in the process is pretty straightforward.

5 Further Work

5.1 Mobile application

We can develop a mobile application that acts as a frontend layer interfacing with the backend and the blockchain. This would allow for ease-of-use for the entities involved in the supply chain.

5.2 Refinement and additional features

The process described in aforementioned sections can be refined by including additional functionality for payment, bills, etc.

6 Conclusion

In this paper, we have proposed an implementation of a supply-chain management system for the automotive industry. This management system utilises the power of blockchain to create an accountable, easily-tracable tracking system.

By implementing such a system within the industry, the goal is to improve transparency between parties, increase efficiency, have better accountability, and be able to trace defective parts in a significantly easier way. The solution proposed is relatively straightforward and its implementation is simple enough to be feasible.

References

[Fraga-Lamas and Fernández-Caramés2019] Paula Fraga-Lamas and Tiago M. Fernández-Caramés. 2019. A Review on Blockchain Technologies for an Advanced and Cyber-Resilient Automotive Industry. *IEEE Access*, 7:17578–17598.

[Miehle et al.2019] Daniel Miehle, Dominic Henze, Andreas Seitz, Andre Luckow, and Bernd Bruegge. 2019. PartChain: A Decentralized Traceability Application for Multi-Tier Supply Chain Networks in the Automotive Industry. In *2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON)*, pages 140–145, April.

[Sharma et al.2019] Pradip Kumar Sharma, Neeraj Kumar, and Jong Hyuk Park. 2019. Blockchain-Based Distributed Framework for Automotive Industry in a Smart City. *IEEE Transactions on Industrial Informatics*, 15(7):4197–4205, July.