Progress towards Federated Logistics through the Integration of TEN-T into A Global Trade Network

D2.18 EGTN smart contracts and associated PI motivated workflows in the context of SLA management final version

Document Summary Information

Grant Agreement No	860274	**Acronym**	PLANET
Full Title	Progress towards Federated Logistics through the Integration of TEN-T into A Global Trade Network		
Start Date	01/06/2020	**Duration**	36 months
Project URL	www.planetproject.eu		
Deliverable	D2.18 EGTN smart contracts and associated PI motivated workflows in the context of SLA management final version		
Work Package	WP2 – PLANET Cloud-based Open EGTN Infrastructure		
Contractual due date	30/11/2022	**Actual submission date**	28/11/2022
Nature	Other	**Dissemination Level**	PU
Lead Beneficiary	Konnecta Systems Limited		
Responsible Author	Aristea Zafeiropoulou (KNT)		
Contributions from	Harris Niavis (INLE), Claudio Salvadori (NGS)		

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No 860274.

Revision history (including peer reviewing & quality control)

Version	Issue Date	% Complete[1]	Changes	Contributor(s)
V0.1	30/8/2022	0	Initial Deliverable Structure	Aristea Zafeiropoulou
V0.2	1/9/2022	5%	ToC added	Aristea Zafeiropoulou (KNT) Harris Niavis (ILS)
V0.3	18/11/2022	80%	Added content to all chapters	Aristea Zafeiropoulou (KNT) Harris Niavis (ILS)
V0.4	24/11/2022	100%	Adjusted report based on reviewer feedback	Aristea Zafeiropoulou (KNT)

Disclaimer

Copyright message

1 According to PLANET's Quality Assurance Process

Table of Contents

List of Figures

List of Tables

Glossary of terms and abbreviations used

Abbreviation / Term	Description
AI	Artificial Intelligence
ALICE	Alliance for Logistics Innovation through Collaboration in Europe
BC	Blockchain System
COSSP	COSCO Spain
ETA	Expected Time of Arrival
FF	Freight Forwarder
IoT	Internet of Things
LL	Living Lab
PI	Physical Internet
PoD	Proof of Delivery
PoR	Port of Rotterdam
PoV	Port of Valencia
SLA	Service Level Agreement
SP	Service Provider
T&L	Transport & Logistics

1 Executive Summary

This deliverable is the final report showcasing the work undertaken in subtask ST2.5.2 in PLANET. It focuses on the design and structure of the Blockchain-enabled smart contracts which are called to facilitate, verify, or enforce the negotiation or performance of a contract or an aspect of the SLA. Smart contracts are employed in the context of Blockchain interoperability, which aims at unifying multiple proprietary Blockchain systems of different T&L stakeholders. In this manner, actors across the entire T&L supply chain shall be able to collaborate and exchange information seamlessly.

The report aims to inform any stakeholder or consortium of stakeholders involved or interested in the design of innovative, cross-organisational EU-Global T&L networks, but also any stakeholders interested in the deployment of Blockchain interoperability solutions in T&L or any other field where the use of smart contracts can be applied by replacing existing paper-based contracts.

The *EGTN Smart Contracts*, developed in the context of the presented work, follow predefined and pre-agreed "if-then" rules and are integrated in the *EGTN Interledger Service* – as presented in D2.16. They are designed and developed based on the PI-inspired business workflows and related user stories that were set out in collaboration with the project's Living Labs. In this manner, they aspire to highlight the business value smart contracts offer to SLA management in the T&L domain.

Smart contracts guarantee a trustworthy, seamless, and distributed process of contract negotiation and execution that significantly reduces time, administrative overheads, and costs which are currently typically spent on manual interorganisational processes. In this manner, the benefits of the smart contracts are instrumental in the roadmap towards the PI, empowering T&L stakeholders to employ distributed and community-driven approaches, instead of centralised, and often proprietary technological solutions.

2 Introduction

2.1 Mapping PLANET Outputs

Purpose of this section is to map PLANET's Grant Agreement commitments, both within the formal Deliverable and Task description, against the project's respective outputs and work performed.

Table 1: Adherence to PLANET's GA Deliverable & Tasks Descriptions

PLANET GA Component Title	PLANET GA Component Outline	Respective Document Chapter(s)	Justification
DELIVERABLE			
D2.18 EGTN smart contracts and associated PI motivated workflows in the context of SLA management final version		Chapter 4	Chapter 4 presents how smart contracts are used in the context of SLA management in PLANET, including the PI-motivated workflows.
TASKS			
ST2.5.2 EGTN Smart Contracts focuses on the contracts' structure and associated PI motivated workflows in the context of SLA management. Correspondingly, when the right conditions are satisfied, the smart contract also executes and records its outcome/transaction in the Blockchain.	Blockchain enabled smart contracts will be exploited as computer programs stored in the Blockchain to facilitate, verify, or enforce the negotiation or performance of a contract or an aspect of the SLA. Blockchain.	Sections 4.1, 4.2, 5.2	Sections 4.1 & 4.2 focus on SLA Management in PLANET, with a focus on the use cases explored in the project. Section 5.2 presents the implementation of the smart contracts.
	Consequently, Smart contracts can be triggered to execute automatically when predetermined terms and conditions are met by encoding 'if then' rules that depend on other actions that occur in the supply chain and are recorded through the IoT and the connectivity infrastructure in the	Sections 3.2, 5.1	Section 3.2 presents the smart contracts in the context of the EGTN Platform, more importantly in relation to other EGTN components (IoT, Data Lake etc.).

	Blockchain. Correspondingly, when the right conditions are satisfied, the smart contract also executes and records its outcome/transaction in the Blockchain.		Section 5.1 presents the “if-then” rules that define the terms & conditions of the smart contracts.

2.2 Deliverable Overview and Report Structure

This deliverable is organized as follows:

- The first section of Chapter 3 is dedicated to the Physical Internet and more specifically on the relationship between the PI, Blockchain technologies, and smart contracts. The second section offers an overview of the smart contracts in the EGTN Platform.
- Chapter 4 presents how SLA management is achieved through smart contracts in the two use cases identified in the Living Labs of PLANET.
- Chapter 5 is a primarily technical chapter, and it presents the EGTN Smart Contracts and more specifically the functions developed based on the requirements defined by the Living Labs. The Living Lab use cases are also presented in detail in *D2.16 Integration and Interoperability of proprietary Blockchain Systems for Seamless Global Trade Workflows final version*. A section setting out the business requirements with regards to the development of AI-enabled smart contracts is also included in this chapter.
- Chapter 6 focuses on the business value of smart contracts as well as the value they may bring in the roadmap towards the PI.
- Finally, Chapter 7 concludes this report by offering a summary and final remarks.

3 Physical Internet & SLA Management

3.1 The PI Paradigm

The Physical Internet (PI) is an emerging breakthrough logistics paradigm that can be defined as an *open global logistics system founded on physical, digital, and operational interconnectivity through encapsulation, interfaces, and protocols* [1]. It is driven by technological, infrastructural, and business innovation, hence the emergence of smart contracts and their application in modern T&L networks is highly intertwined with the evolution of the PI. Indeed, as it described in the following paragraphs, Blockchain features, and functionalities have the potential to meet PI implementation requirements as well as overcome key PI barriers and deficiencies [2].

A well-known hurdle that the PI needs to overcome is that regardless of its advantages, it cannot continue relying on centralised networks nor on the existence of a leading authority [3]. The PI will be able to proliferate only if it is handled in a distributed and community-driven approach. This is where Blockchain technologies can make a difference, as not only is their entire premise based on strictly decentralised solutions, but they also overcome issues of information asymmetry [4]. In fact, the benefits of the PI are enhanced through Blockchain technologies, as they offer better tracking of data across the entire T&L network, efficient, trustworthy, and objective contract execution through smart contracts and increased security protection through encryption. In this manner, the T&L process chain becomes more effective through increased information transparency and power balance [4].
A key enabler for the PI is digital interoperability as it enables separate T&L networks to conduct business with each other seamlessly [5]. More specifically, digital interoperability is defined as “the ability to achieve quick, seamless, secure, and reliable data and information exchange between computing devices (devices capable of transfering data), between information systems (of different organisations, logistics networks), or between devices and systems, for the aim of enhancing cooperation or coopetition of independent logistics parties or networks“[5]. As the PI aims at interconnecting T&L networks, i.e., moving from collaboration to interconnection of the involved actors, in a similar manner, Blockchain solutions need to take a step towards the interconnection of the different Blockchain solutions that are currently employed by separate T&L stakeholders. Blockchain interoperability is a growing research topic; in PLANET it has been explored under the scope of T&L and it has been the focus of the work undertaken in Task 2.5. More specifically, it has been addressed by developing a Blockchain Interoperability solution for T&L and applying it to the project’s Living Labs.

Blockchain interoperability ranges from relatively simple - i.e., interoperability between smart contracts on the same ledger - to very complex – i.e., inter-Blockchain interoperability or between a Blockchain network and a legacy system. A different challenge that Blockchain interoperability addresses is to enable data exchanges between a public Blockchain where pseudonymous data are used and a private Blockchain where transactions require user identification. In the PLANET project, the EGTN platform interconnects existing backend Blockchain systems and exposes universal, standardised interfaces to actors of the supply chain.

Notably, smart contracts play an instrumental role in the roadmap towards the PI. The Alliance for Logistics Innovation through Collaboration in Europe (ALICE) has proclaimed that they can achieve this in two ways:

- Logistics nodes shall interact with each other in an automated manner creating seamless booking systems backed by smart contracts, which will in turn increase response time significantly.
- Supply chains shall become proactive in the smart use of resources and capabilities by employing fully autonomous services and operations with predefined smart contracts [6].

The advantages of smart contracts are also evident when changes or unforeseen events occur at some point within the supply chain and certain actions need to take place (e.g., rerouting of goods). Blockchain enables new smart contracts to be issued (e.g., Bill of Lading) in a distributed manner, and in this way reducing the previously manual administrative overheads and avoiding delays. In addition to this, any interested party, such as customs officials, that has the required permissions can access the updated information, facilitated by Blockchain and the

Blockchain interoperability. This is of crucial importance, as it enables cross-organisation information flow, which means that decisions are data-driven and, therefore, can be changed at any step of the supply chain e.g., in case of damage to the cargo during transport, reroute the cargo from the distribution centre to a repair centre; all this happening in real time [7]. In addition to this, all transactions within the Blockchain network are auditable and data is immutable ensuring in this way that the entire process can be traced backwards in a trustworthy manner [8].

Overall, the entire T&L chain can be sped up using smart contracts that are executed based on shipping steps. For instance, the following steps can be optimised:

- Real-time access to all the relevant information by all stakeholders involved in international trade processes can reduce administrative costs.
- Increased transparency through asset tracking, which can also help avoid shipment delays.
- Optimised load capacity can minimize the shipping costs of T&L companies [9].

Following this manner, IoT data and AI models can make the step towards the realisation of the PI even smaller, by decreasing reaction time to such events (especially the use of real-time IoT data) and increasing efficiency in smart contract negotiations (through AI predictive models). Both solutions are integrated in the EGTN Platform along with Blockchain and are presented in the following sections.

3.2 Smart Contracts in the EGTN Platform

The emergence of smart contracts alongside Blockchain technologies has disrupted the operational flow and contract negotiation in several fields, including T&L. Through smart contracts the rules and conditions of an agreement are defined and agreed in advance between all parties involved. Once a smart contract is established, these cannot be altered without the consent of all parties. Smart contracts execute automatically upon event trigger (e.g., empty truck of freight forwarder arrives at a logistics warehouse triggers a new booking) and guarantee the consignor the acceptance and execution of the pre-agreed offer, but also enforce them to fulfil their own contractual obligations (e.g., fulfil a payment). These powerful benefits can contribute greatly and bring T&L a step closer to the PI.

The entire T&L chain can be sped up using smart contracts that are executed based on shipping steps by offering: (i) real-time access to all the relevant information to all involved stakeholders of international trade processes - i.e., this can reduce administrative costs, (ii) increased transparency through asset tracking, which can also help avoid shipment delays, and (iii) optimised load capacity - i.e., this can minimise shipping costs of T&L [1].

The PLANET project aims to empower T&L stakeholders to increase trust in their interactions through the usage of the EGTN Interledger Service and smart contracts and to address challenges such as traceability of shipments, transparency between different stakeholders. To this end, it develops an original blueprint architecture accompanied with an instantiation of a cloud-based platform as well as best practices that support T&L actors in the definition and implementation of clear digital strategies and to offer support in their physical operations. The cloud-based platform, namely the EGTN Platform, lays the foundation on top of which the EGTN services are developed, as shown in Figure 1. The EGNT Platform aspires to be an inclusive and powerful platform, as it can be adopted by any size of T&L actor or firm. In this manner, it is not limited to be adopted only by large enterprises with expensive IT budgets, while it aims at taking a step closer to the realisation of the Physical Internet paradigm.

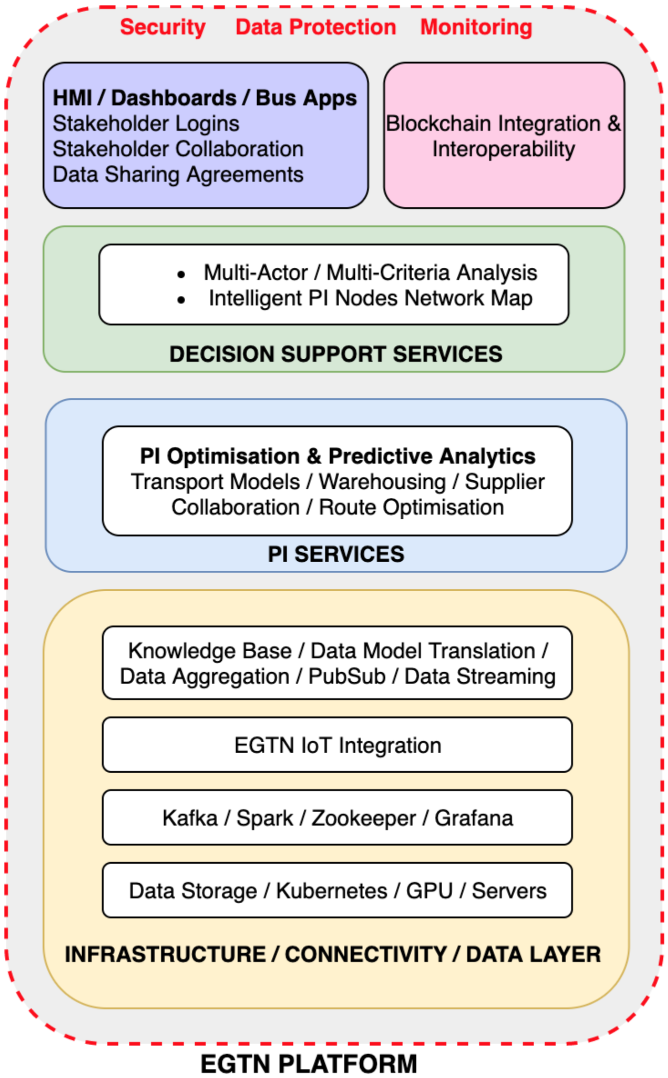

Figure 1: The EGTN Platform

The EGTN Interledger Service is a Blockchain "front-end", available through the EGTN Platform, that unifies multiple backend Blockchain systems and supports interorganisational and intercommunity trade workflows. The EGTN Interledger Service includes functionalities, such as SLA monitoring based on smart contracts and driven by IoT data as well as automatic generation of AI-enabled smart contracts driven by the AI predictive analytics services of the EGTN Platform.

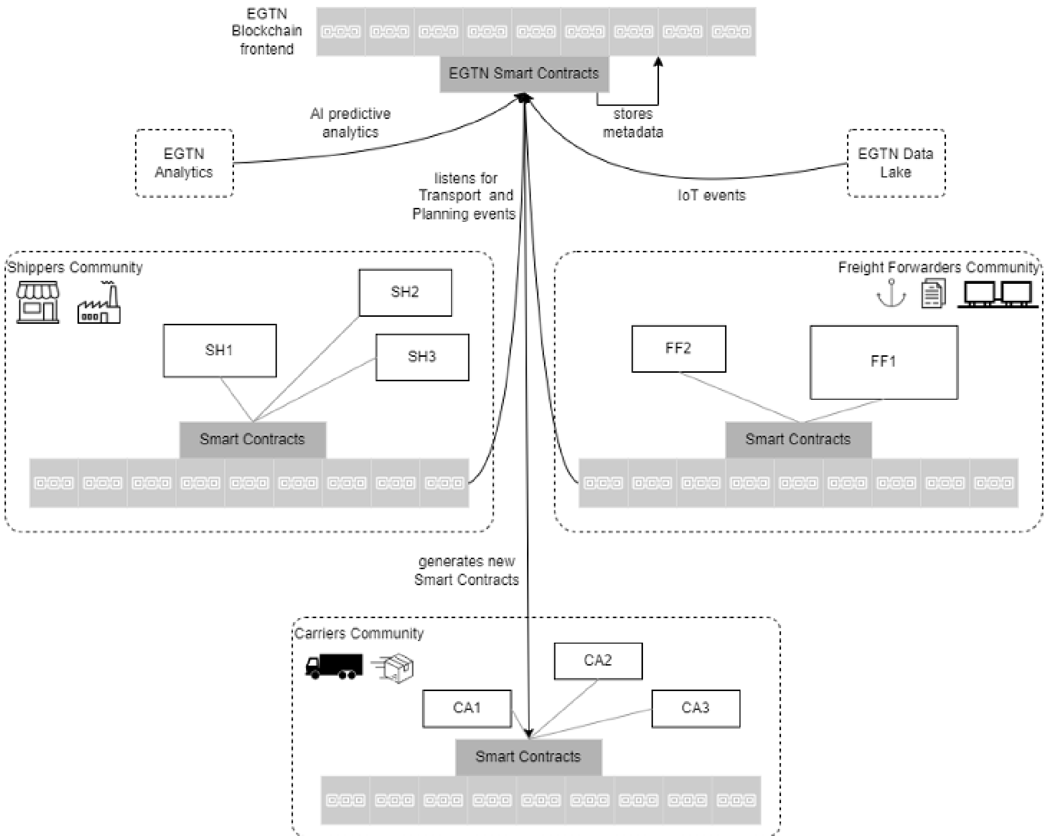

Figure 2: The EGTN Interledger Service and Smart Contracts

As shown in Figure 2, it takes advantage of other cutting edge microservices provided within the EGTN Platform i.e., Artificial Intelligence (AI) Analytics, Decision Support Systems (DSS), as well as of the plethora of datasets available in the EGTN Data Lake to enable an advanced functionality such as forwarding of logistics events between disconnected communities and cross-checking the validity of events through IoT data enabling trusted Proofs of Departure/Arrival/Pickup of a logistics unit e.g. a container. A detailed description of the EGTN Interledger Service architecture, its internal modules and data flows can be found in D2.16.

Smart contracts are employed to guarantee the integrity of the data entering the ledger and to automate procedures in the supply chain that are currently handled using paper-based documents. The goal of data integrity is achieved through the concurrent execution of the smart contracts by all the network participants.

As Figure 2 illustrates, the Data Lake provides an event-streaming service that ingests logistics events and IoT data from sensors, following the latest GS1 standards. The smart contracts consume data within the EGTN Platform, combine these with other information sources (e.g., outputs from AI predictive models), and automatically trigger actions, such as the creation of trusted metadata that can be used as a single source of truth in Blockchain communities or even the generation of new smart contracts that monitor and safeguard Service Level Agreements (SLAs) between logistics communities - i.e., Shippers (SHx), Carriers (CAx) and Freight Forwarders (FFx).

Smart contracts run when predetermined terms and conditions are met ("if-conditions") and execute actions or trigger events/alarms ("then-rules") without an intermediary's involvement or loss of time. The automatic execution of the actions by all the involved actors ensures the integrity of the data pushed to the Blockchain and ties them in secure and transparent agreements. EGTN Smart Contracts take advantage of the diverse data and knowledge residing within the EGTN Platform (output from backend Blockchain systems, IoT data, Big Data analytics) to enable reliable generation of metadata and trusted contract execution

4 SLA Management in PLANET

The lifecycle of smart contracts correlates highly with the lifecycle of SLAs as described in D2.17. They have been recently proposed to address the challenge of efficiently and securely monitoring agreements and to introduce a distributed, fair and transparent decision-making process for SLA management in the cloud domain [2], [3], [4]. Similarly, the inherent characteristics of smart contracts can guarantee a trustworthy, distributed process that reduces time and costs spent on manual interorganisational processes in the Physical Internet paradigm. The decentralised nature of smart contracts ensures the engagement of all T&L stakeholders in the negotiation and monitoring of a contract paving the way towards a transparent and global logistics system. This section analyses the proposed solution in PLANET for SLA management.

In the context of PLANET, two use cases have been identified aiming to show the business value the EGTN Smart Contracts bring in SLA management in the T&L domain and are described in the following sections.

4.1 Exchange of Transport Orders

The first use case involves Port of Valencia (FV), COSCO SHIPPING LINES SPAIN (COSSP) and DHL EXEL SUPPLY CHAIN SPAIN (DHL). Its purpose is to demonstrate the need for seamless information exchange between maritime ports, shipping lines, warehouse providers and last mile delivery services to increase efficiency across the supply chain. In this scenario, COSSP manages shipments from Asia to Spain (Port of Valencia) while the inland transport is managed by DHL Spain and CityLogin. Figure 3 illustrates the entire operational overview of the scenario workflow. At the top of the diagram all stakeholders are displayed: from COSSP and Port of Valencia to DHL and CityLogin. The key takeaway of the diagram is, however, the flow of information exchanged between all these stakeholders.

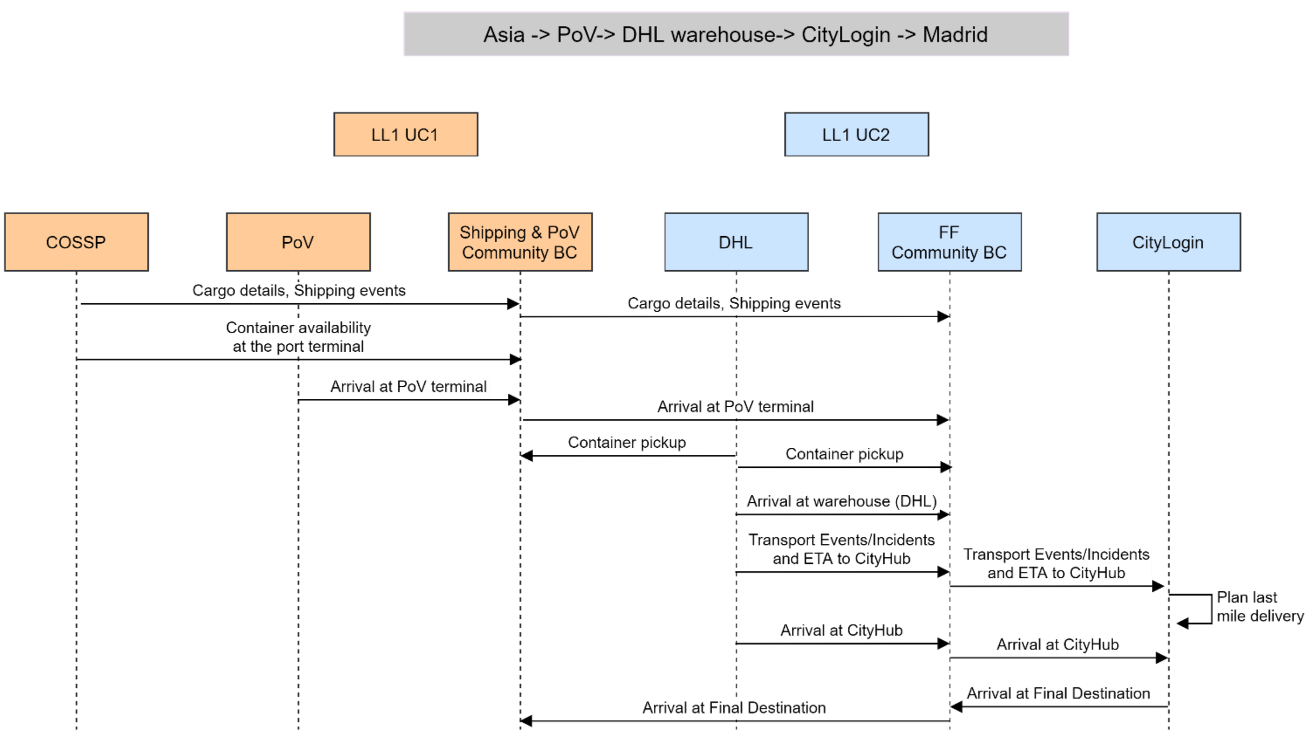

Figure 3: Exchange of Transport Orders

The following user stories have been identified in the context of this scenario that show the exchanges of information between the involved actors and will be tested during the LLs evaluation in the upcoming months.

4.1.1 User Stories

The first user story focuses on Container Ready for Transport at the PoV, as shown in Figure 4.

a. DHL adds a Blockchain asset in the Freight Forwarder (FF) community Blockchain, referenced by GSIN.
b. PoV internal system (e.g., an employee) creates an asset in the Shippers community Blockchain, upon the arrival of the container at the PoV.
c. PoV Blockchain emits the event ContainerArrivalPort.
d. The ContainerArrivalPort event is forwarded by the EGTN Interledger Service to the Freight Forwarder (FF) Blockchain.
e. Container passes all the checks within the PoV, so the internal system invokes a transaction in the Shipper's Blockchain, which emits the event ContainerReadyForTransport.
f. The EGTN Interledger Service forwards the ContainerReadyForTransport event to the FF Blockchain.
g. The EGTN Interledger Service checks the GPS sensor readings coming from the container and if the coordinates match the ones of a specified checkpoint, it emits the trusted event **EGTNContainerReadyForTransport**.

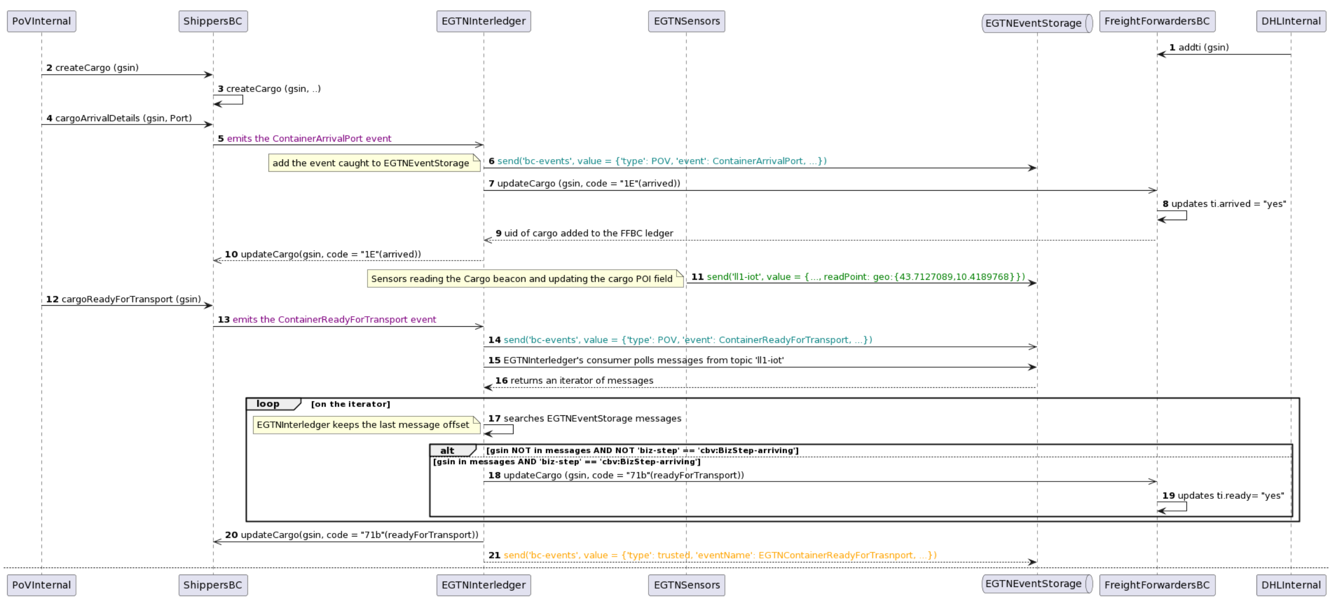

Figure 4: Container Ready for Transport

The second user story focuses on a container that departed from PoV, as illustrated in Figure 5:

a. The container departs from PoV, so the internal system invokes the corresponding transaction in the Shippers Blockchain, which emits the event ContainerDeparted.
b. The EGTN Interledger Service forwards the ContainerDeparted event to the FF Blockchain.
c. The EGTN Interledger Service checks if the ContainerPickup event has been emitted by the FF Blockchain.
d. DHL picks up the container and the DHL internal system invokes the corresponding transaction in the FF Blockchain, which emits the event ContainerPickup.
e. The EGTN Interledger service emits the event EGTNContainerDeparted.

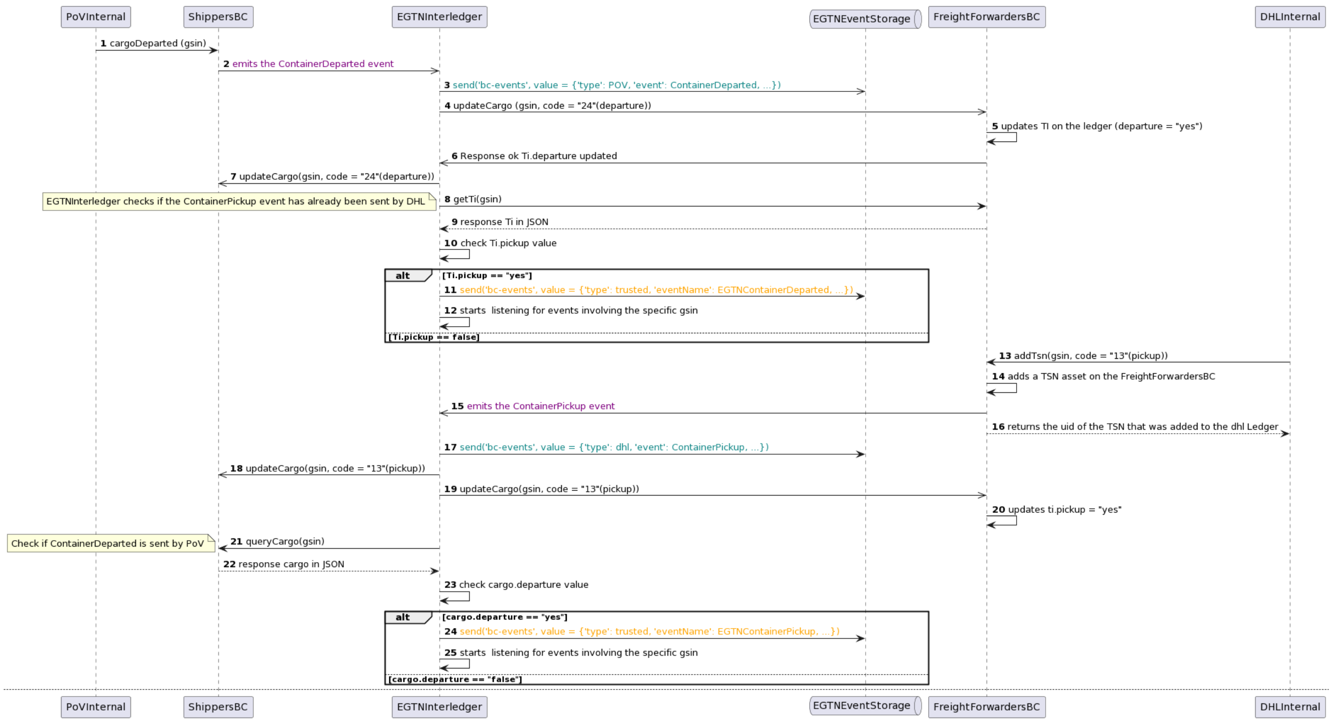

Figure 5: Container Departed

The third user story deals with a contract violation (Figure 6):

a. DHL adds an SLA agreement in the FF Blockchain regarding the transport of a container from PoV to DHL warehouse.
b. The EGTN Interledger Service enables a smart contract that monitors the IoT data coming from the IoT devices monitoring a specific logistics unit (SSCC) or the whole container/truck (GIAI associated to a certain GSIN).
c. Temperature measurement is higher than the threshold specified in the agreement and the EGTN Interledger Service emits the trusted event EGTNContractViolation containing information about the agreement and the reason of the violation. The event is forwarded to all the Blockchains connected to the EGTN Interledger Service.

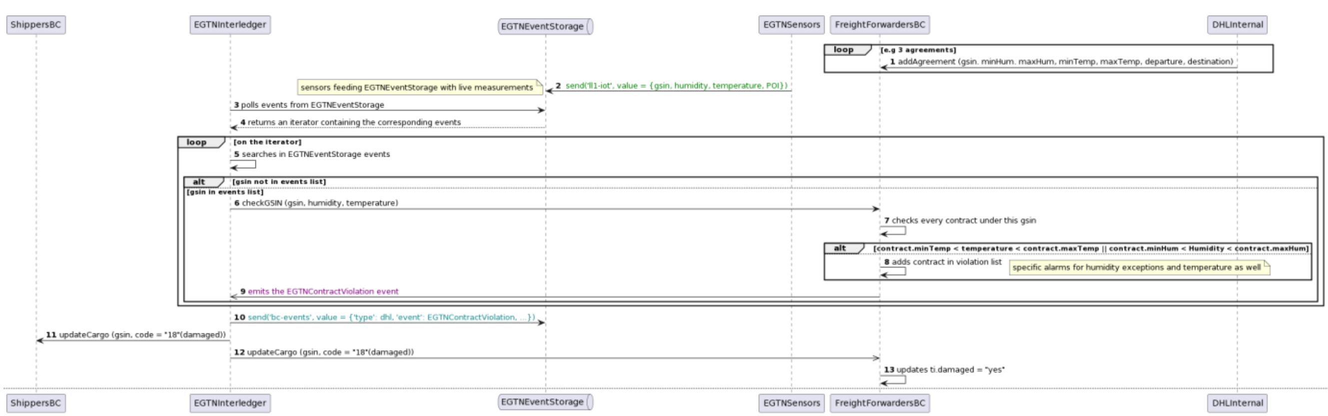

Figure 6: EGTN Contract Violation

4.2 Integration and Synchronisation of Maritime Ports

The second use case involves stakeholders between LL1 and LL2 and more specifically Fundacion Valenciaport, COSCO Spain, DHL and BLOCKCHAIN FIELDLAB BV (BlockLab). It aspires to highlight the requirement for Blockchain interoperability between maritime ports to exchange information related to cargo dispatching, cargo reception and terrestrial transportation and, in this manner, promote collaboration and transparency between two maritime ports (PoV, PoR) and a Freight Forwarder (DHL). Figure 7 provides an overview of the scenario, in which PoR manages the shipment of a container from Asia to PoR, then DHL manages the inland road transportation to PoV.

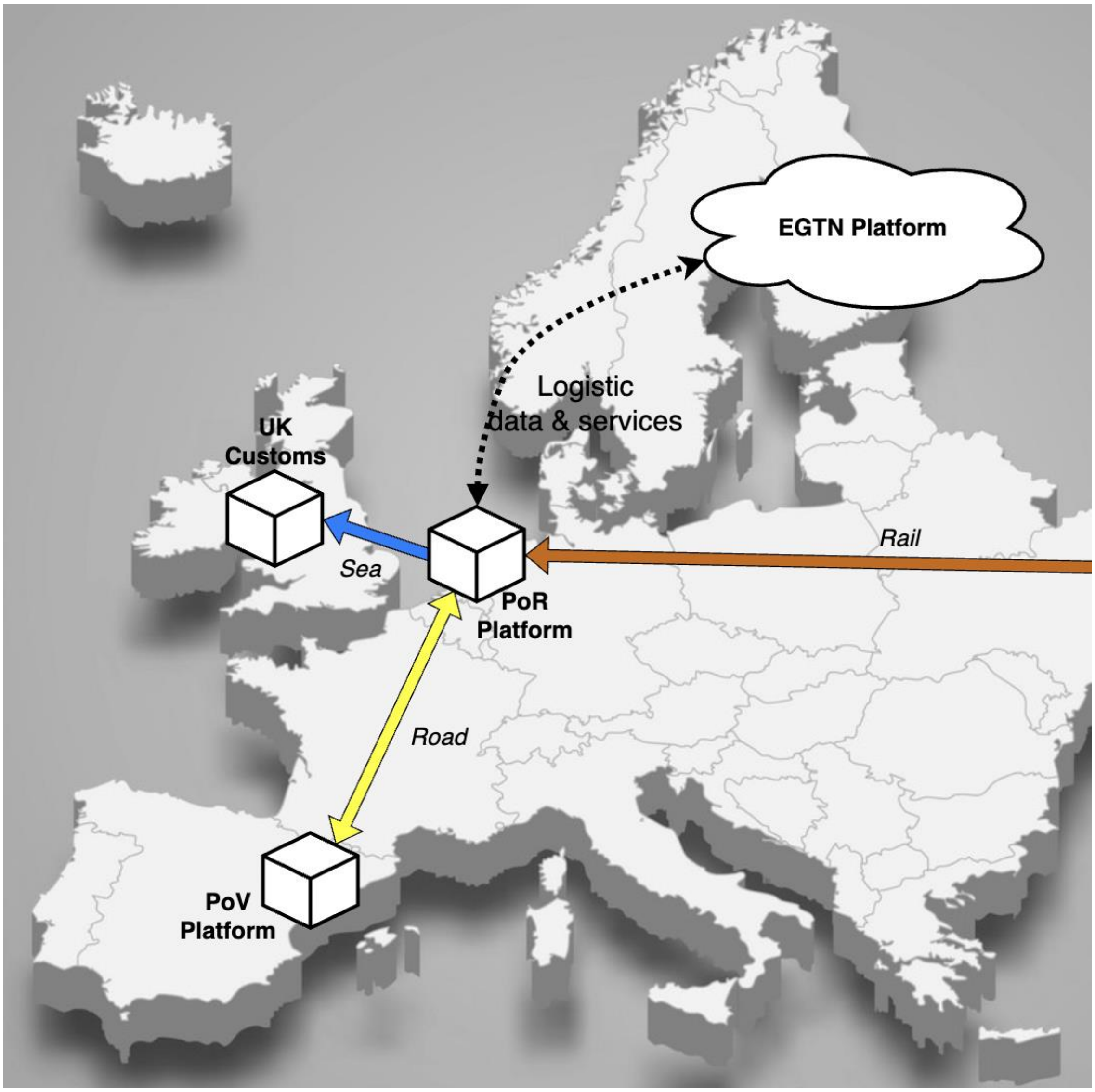

Figure 7: Synchronisation of Maritime Ports

As Figure 8 illustrates when a cargo from United States arrives to the Port of Rotterdam all relevant data (cargo details, shipping events) are stored withing the port's Blockchain system, including the availability of containers at the port terminal. This information is also shared with the other Blockchain systems that are involved in the process, the Freight Forwarder's Blockchain and the Shipper's Blockchain. Connected warehouses and logistics service providers at the Port of Valencia are now empowered to anticipate the incoming cargo and reserve storage or transport space for the next segment of the shipment, while the freight is in transit.

Once the Freight Forwarder picks up the container, they send to the Blockchain of the Port Community and the Freight Forwarder's Blockchain the pick-up confirmation as well as the Estimated Time of Arrival (ETA) to the Port of Valencia.

Upon terrestrial arrival to the Port of Valencia, all involved Blockchain systems are updated accordingly to ensure that every party is aware that the cargo has reached its destination and is ready to be transported to Africa.

The ports involved use different road transport documents; Port of Valencia uses the Unified Transport Document (UTD, or DUT as the Spanish acronym) while the Port of Rotterdam uses the electronic Road Transport Document (eCMR). Using these electronic documents, information and events related to road freight transport can be shared across both platforms enabling road transport optimisation between the Port of Rotterdam and Port of Valencia. The EGTN Interledger Service acts as a proxy between the two ports, which exchange hashes of the documents along with metadata through their Blockchain systems, with the actual documents shared through the EGTN Platform and retrieved only by trusted actors.

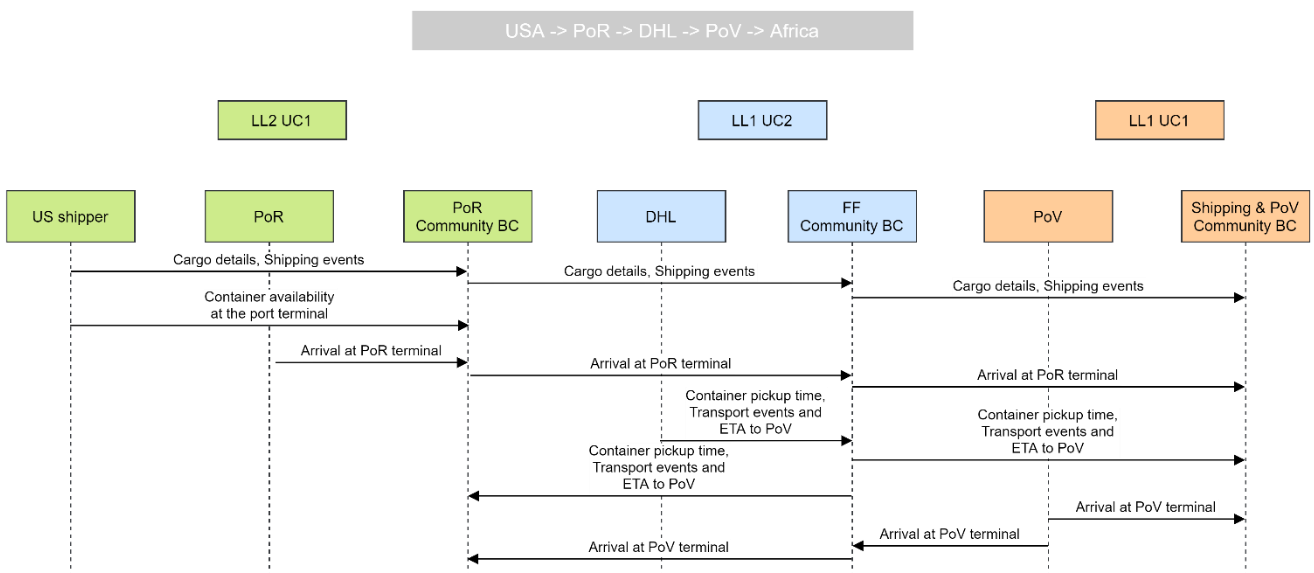

Figure 8: Synchronisation of Maritime Ports

4.2.1 User Stories

The following user stories have been identified in the context of this use case and aim at showing seamless information exchange between the involved actors and will be tested during the LLs evaluation in the upcoming months.

The first user story focuses on how Transport Document from Port of Rotterdam – i.e., eCMR - is notarised in the Port of Rotterdam (PoR) systemL

a. A user creates a shipment in PoR system, enters main shipment information and attaches a transport document i.e., an eCMR.
b. A root hash of the document is notarised as reference to the shipment and stored in the Blockchain of PoR.
c. The local Blockchain emits an event including the hash of the document as well as other metadata, namely the information of the EGTN Data Model.
d. The Blockchain connector of the EGTN Interledger Service listens to the PoR smart contract and pickups the event that contains only publicly shareable information.
e. The EGTN Interledger Service forwards the event to all connected Blockchain communities.

f. The event is stored in the EGTN Platform and displayed to the users on the EGTN Dashboard.

The second user story deals with a user who needs to retrieve the actual transport document:

a. A user logs in the EGTN Dashboard and explores the logistic events coming from multiple backend Blockchain systems through the EGTN Interledger Service.
b. The user sees an event and decides that they require more information about it, so they ask for the actual document.
c. The EGTN Dashboard makes a request to the API of the PoR's system using the user's token and the hash of the document to retrieve, found in the logistics event.
d. The PoR system validates the request by checking the permissions of the user to access the document.
e. The document is sent to the EGTN Dashboard and displayed to the user.

5 EGTN Smart Contracts

This chapter describes in detail the smart contracts that were developed and integrated in the EGTN Interledger Service in the context of PLANET. The signature of each function is presented, while more advanced concepts like the AI-enabled smart contracts are explained in separate subsections. The smart contracts code can be found in the PLANET's GitLab repository, https://gitlab.com/planet-h2020/egtn-smart-contracts.

5.1 If-then Rules

Table 2 presents the "if-then" rules (ITXX) that are followed in the EGTN Interledger service, which aim to standardise and streamline inter-organisational T&L workflows. The first two rules (IT01, IT02) aim to enhance the legitimacy of Dispatch claims from the Shipper, Freight Forwarder and Carrier Communities. The EGTN Interledger service monitors the backend systems of all communities and registers for events that declare:

- In IT01 the dispatch of cargo from the Shipper (ContainerDeparted) and the cargo pick-up from the Carrier (ContainerPickup) and
- In IT02 the Proof of Delivery (PoD) reported by the Carrier and Freight Forwarder (ContainerArrivalWarehouse).

Upon the collection of the corresponding events from the actors, the EGTN Smart Contracts emit a trusted Cargo Dispatched event in the case of IT01 (EGTNContainerDeparted) and a trusted Cargo Delivered event in the case of IT02 (EGTNContainerArrivalWarehouse). The rest of the rules, from IT03 to IT05, compare data coming from backend Blockchain systems with IoT data coming from smart T&L assets. More specifically, IT03 guarantees the trusted execution of the smart contracts between the actors involved - e.g., Blockchain system events are compared with "Reach a POI" records sent by smart pallets and if both sources agree then a trusted event is emitted (EGTNReadyForTransport). The IT04 smart contracts compare system defined thresholds with IoT data from smart T&L assets to drive their decisions, e.g., if the vibration measurement in a smart pallet exceeds a predefined threshold, then break the contract and emit the event EGTNContractViolation. The IT05 type of smart contracts employ system data and forecasts from the EGTN AI Analytics components to streamline internal processes, e.g., hire more personnel when the forecasted incoming number of pallets exceeds a limit, A more detailed presentation of the AI-enabled smart contracts can be found in 5.3.

ID	Data & If Condition	Then Description
IT01	Cargo Departure event reported from Shipper **AND** Carrier	Enhanced-trust Departure
IT02	Cargo Delivery event reported by Carrier **AND** FF	Enhanced-trust Delivery
IT03	Contract defined delivery point/time = Unloading IoT coordinates/time	Contract fulfilled
IT04	Contract defined vibration/temperature tolerance < IoT data	Contract violated
IT05	AI forecasts & System event	New Contract Generation

Table 2: Smart Contracts If-Then rules in the EGTN Use Case

5.2 EGTN Smart Contracts implementation

This section presents the smart contracts that were developed in the context of PLANET, namely the signature of each function together with a brief explanation about its functionality, inputs and outputs, in order to support the if-then rules presented in the previous section.

5.2.1 Transport Documents

This section presents the functions of the smart contracts that were developed to enable the trusted exchange of Transport Documents between the connected Blockchain communities. The data structures stored on the ledger follow the GS1 EPCIS terminology, i.e., Transport Instruction (TI) for storing master data of a shipment and Transport Status Notification (TSN) for storing transactional data such as events.

- **addDocument(stub shim.ChaincodeStubInterface, args [] string) peer.Response**

The *addDocument* function is the first one called during the lifecycle of a Transport Document and is responsible for creating the *document* asset for a specific shipment on the underlying Blockchain. The *document* asset follows the data model explained in D2.16 and it contains the minimum shareable information of a Transport Document, towards respecting the privacy and confidentiality of the involved T&L actors. Each *document* asset can be identified with the documentID and it contains information about the cargo.

In addition, the *addDocument* function adds a TI asset in the ledger, which contains the information about the status of a shipment. The struct of a TI asset is presented below:

```
ledgerTi := Ti{
    Gsin: TransportOrderReferenceNumber,
    Delay: "false",
    Damaged: "false",
    WrongRoute: "false",
    ArrivedPort: "yes",
    ArrivedWarehouse: "false",
    Unload: "false",
    Dropoff: "false",
    Enroute: "false",
    Departured: "false",
    Pickup: "false",
    Positioned: "false",
    Readyfortransport: "false",
}
```

Moreover, the *addDocument* function invokes a TSN transaction (using the *addTsn* function described below) to keep track of events happening during the transport of the shipment, supporting a subset of the GS1 Global Data Dictionary[2] codes, see also Table 3.

[2] Global Data Dictionary, https://www.nweurope.eu/media/14875/transportstatusconditioncode-gs1-120301.pdf

GS1 Code	Description	EGTN Interledger
18 - damaged	Conditions of a contract have been violated	**TSN - status 18 = damaged**
1E - arrived	At the port or at the warehouse	**TSN - status 1E = arrived (at a platform, multileg)**
71b - readyfortransport	All processes within the port are done (customs clearance etc.)	**TSN - status 71b = ready for transportation**
24 - departure	Departure from PoV	**TSN - status 24 = departure**
13 - pickup	Picked up from DHL at the PoV	**TSN - status 13 = pick up**
29 - unload	Unload at the CityLogin Hub	**TSN - status 29 = unload**

Table 3: GS1 GDD Codes Supported by EGTN

```
tsn := Tsn{
        Gsin: documentPoR.Transport_Order_reference_number,
        Code: "1E",
        Location: "Port",
}
```

All the assets on the ledger, namely document, TI and TSN, have a common reference, the GSIN, which can be used to retrieve both master and transactional data related to a specific shipment. At the end of this function, a ContainerArrivalPort event is emitted containing the document in JSON format.

```
err = stub.SetEvent("ContainerArrivalPort", documentPoRJSON)
```

- **addTsn(stub shim.ChaincodeStubInterface, args []string) peer.Response**

The *addTsn* function is the most used throughout the lifecycle of a shipment. This function adds a new TSN entry on the ledger that associates the trasnportOrderReference of the shipment given as input with an operational code following the GS1 standard. The list of operational codes used from the Interledger service is explained in Table 3. Apart from adding a TSN in the ledger, the *addTsn* transaction updates the relevant TI asset already on the ledger as well as it emits the corresponding event e.g. the code 1E emits the ContainerArrivalPort event.

- **getDocument(stub shim.ChaincodeStubInterface, args []string) peer.Response**

The *getDocument* function returns the document asset (as an array of bytes) associated with the GSIN given as input.

- **getDocuments(stub shim.ChaincodeStubInterface, args []string) peer.Response**

The *getDocuments* function returns every document asset that is on the ledger.

- **queryDocByGsin(stub shim.ChaincodeStubInterface, transportOrdRefNum string) (LedgerDocumentPoR, string)**

The *queryDocByGsin* function returns the document asset associated with the specified transportreference number i.e., the GSIN.

5.2.2 SLA Agreements

This section presents the functions of the smart contract that was developed to enable the monitoring of SLA agreements between T&L stakeholders connected in the EGTN Interledger Service.

- **addAgreement(stub shim.ChaincodeStubInterface, args [] string) peer.Response**

As its name suggests, the *addAgreement* function creates an agreement asset on the ledger. Each agreement can be referenced by its agreementId. The agreement struct can be found below.

```
type Agreement struct {
    AgreementID string `json:"Agreement_ID"`
    IdDocument string `json:"Document_ID"`
    Name string `json:"Name"`
    Gsin string `json:"Transport_Order_reference_number"`
    Departure string `json:"Departure"`
    Destination string `json:"Destination"`
    MinHumidity string `json:"minHumidity"`
    MaxHumidity string `json:"maxHumidity"`
    MinTemperature string `json:"minTemperature"`
    MaxTemperature string `json:"maxTemperature"`
    LatestDeliveryDate string `json:"latestDeliveryDate"`
    Status string `json:"status"`
}
```

First, the smart contract reassures that there is not another agreement asset on the ledger with the same agreementID and then the agreement is added in the ledger in a JSON format.

err = stub.PutState(agreementID, agreementJSON)

The involvement of a specific shipment in multiple agreements between different pairs of stakeholders is enabled, as the function creates a composite key[3] to associate agreements with the GSIN.

indexAgreementName := "agreement~gsin"

agreementIndexName,err := stub.CreateCompositeKey(indexAgreementName, [] string{agreement.Gsin, agreement.AgreementID})

[3] Composite key, https://en.wikipedia.org/wiki/Composite_key

- **getAgreement(stub shim.ChaincodeStubInterface, args []string) peer.Response**

The *getAgreement* function returns an agreement struct by agreementID.

- **getAgreements(stub shim.ChaincodeStubInterface) peer.Response**

The *getAgreements* function returns all the agreements assets that are currently on the ledger (active or not).

- **queryAgreementsByGsin(stub shim.ChaincodeStubInterface, args []string) peer.Response**

Given the transport order reference number i.e., the GSIN as input, returns every agreement that is associated with the already mentioned transport order reference number.

- **checkAgreementsMeasurements(stub shim.ChaincodeStubInterface, args []string) peer.Response**

As explained in D2.16 the EGTN Interledger Service acts as a validating and monitoring mechanism for agreements based on measurements coming from sensors attached to the containers. In case of an agreement violation, the proper event is emitted. The *checkAgreementsMeasurements* function takes as input three parameters which are the GSIN of the shipment, temperature, and humidity. The EGTN Smart Contract finds and checks every agreement that is associated with the given GSIN. If either humidity or temperature measurement is out of range in an agreement, then an event is emitted containing the corresponding alarm violation. The condition alarm structure can be found below:

```
type ConditionAlarm struct {
    AgreementID string `json:"agreementID"`
    Gsin string `json:"gsin"`
    Reason string `json:"reason"`
    Value string `json:"value"`
    Code string `json:"code"`
}
```

The snippet that does the validation is described below. It should be highlighted that only active agreements are taken into consideration.

```
if agreement.Status == "active"
{
    //Convert temperature of the agreement to float
    minTemp, _ := strconv.ParseFloat(agreement.MinTemperature, 64)
    maxTemp, _ := strconv.ParseFloat(agreement.MaxTemperature, 64)

    //Convert humidity of the agreement to float
    minHumidity, _ := strconv.ParseFloat(agreement.MinHumidity, 64)
    maxHumidity, _ := strconv.ParseFloat(agreement.MaxHumidity, 64)

    temperatureMeasurement, _ := strconv.ParseFloat(temperature, 64)
    humidityMeasurement, _ := strconv.ParseFloat(humidity, 64)

    // Check temperature and create a temperature alarm if needed
    if temperatureMeasurement < minTemp || temperatureMeasurement > maxTemp
    {
        temperatureAlarm := ConditionAlarm{}
```

```
        temperatureAlarm.AgreementID = agreement.AgreementID
        temperatureAlarm.Value = temperature
        temperatureAlarm.Gsin = agreement.Gsin
        temperatureAlarm.Code = "18"
        temperatureAlarm.Reason = "temperatureViolation"
        conditionAlarms = append(conditionAlarms, temperatureAlarm)
    }

    // Check Humidity and create a humidity alarm if needed
    if humidityMeasurement < minHumidity || humidityMeasurement > maxHumidity
    {
        humidityAlarm := ConditionAlarm{}
        humidityAlarm.AgreementID = agreement.AgreementID
        humidityAlarm.Value = humidity
        humidityAlarm.Gsin = agreement.Gsin
        humidityAlarm.Code = "18"
        humidityAlarm.Reason = "humidityViolation"
        conditionAlarms = append(conditionAlarms, humidityAlarm)
    }
}
```

5.3 AI-enabled Smart Contracts

This section outlines the scenarios for the AI-enabled smart contracts in the context of PLANET. The scenarios aim to explain the rationale behind dynamic rules that can trigger the automated establishment of smart contracts for the booking of extra trucks for pallet deliveries from a FF's warehouse.

Current warehouse operations are based on pre-agreed contracts with freight forwarders or carriers for a fixed number of trucks. However, unexpected demand surge at specific moments or other events often creates the need for the booking of extra trucks. The combination of predictive models with smart contracts brings added value, as it enables a more efficient and smoother operation of the T&L workflow. By using the predictive models, the operator will know in advance the number of extra bookings needed, while the use of smart contracts will allow for the automated generation of paperless contracts. This automated and ad-hoc trigger of smart contracts beyond normalizing the engagement of resources and reducing the overall operational cost, it also creates opportunities for the easy establishment of new relationships with freight forwarders.

Several parameters may affect the demand for trucks in a warehouse. These include the following:

A. Periods of increased shopping
   a. Christmas and Easter Holidays
   b. The summer period (mainly August)
   c. Bank holidays (local/national). This also affects truck availability.
   d. Special campaigns, such as Black Friday, Cyber Monday, etc.

B. Events affecting routes and truck availability
   a. Strong weather conditions. E.g., the Filomena Storm caused chaos in transportation of goods due to traffic cuts.
   b. Transport strikes.

C. Current affairs and their effects on the economy
   a. The pandemic. For instance, lockdowns cause an increase in online shopping.
   b. International conflicts (e.g., the Ukrainian war)
   c. Fuel prices. Price increases often cause strikes.

D. Day of the week. It is worth paying attention to the day of the week when comparing the same date in different years, e.g., March 12th, 2021, was on a Friday, while the same date in 2022 is on a Saturday.

E. Continuous growth of e-commerce.

The number of truck bookings will be calculated based on the number of outgoing pallets in the warehouse as is predicted by the AI models developed under T2.3. Using a historical dataset, as well as IoT data (depicting real-time trucks/cargo position and status) the models can perform rolling predictions. In this manner, continuous planning of future pallet quantities is based on the actual pallet quantities to this point. For example,

- The prediction on January 1st, 2020, is based on the last two years of data (1st Jan 2018 - 31st Dec 2019)
- The prediction on January 2nd, 2020, is based on the last two years of data (2nd Jan 2018 - 1st Jan 2020) and so on.

Moreover, predictions need to take place within 10-day as well as 3-day windows. For daily operations, 3-day predictions provide a sufficient time frame for truck bookings. In addition to this, unforeseen events may occur that may be known only a few days in advance (e.g., strikes). In such situations, the warehouse operator follows a contingency plan. This is a manual process which requires the booking of extra trucks a few days prior to the event to ensure that pallets will reach their destination on time.

On the other hand, 10-day predictions become valuable from a business perspective as we are approaching periods of increased shopping (e.g., Christmas or Black Friday). For instance, an increase in bookings due to a special event such as Black Friday, can be predicted at least a week in advance. In such an occasion, truck bookings can be made several days ahead of the event and better prices can be negotiated.

Currently, as there is no use of AI, warehouse operators book extra trucks at the very last moment. On some occasions, when they learn about upcoming peaks in demand, they book trucks one week in advance. In such situations, they ensure that there are no associated penalties or cancellation fees.

To address these issues, confidence intervals will be introduced in the predictive models. The aim behind this solution is to ensure that when smart contracts are triggered, they will be used in practice. In this manner, the output of the models will be a more detailed prediction rather than a single number (i.e., the number of pallets). Confidence intervals are particularly meaningful in the case of the 10-day predictions, as their outputs may have higher degree of uncertainty.

In conjunction with the confidence intervals a pricing strategy shall be put into place. This means that low confidence predictions will be associated with higher prices in bookings and lower cancellation penalties. In such a scenario, should a prediction generate more bookings than needed, the warehouse operator will cancel the booking and pay the associated penalty.

Taking all the above into consideration, the combined use of confidence intervals with a pricing strategy provides a means of offering synchromodal logistics. In this manner, the use of AI and Blockchain offer efficient and flexible services that enable the coordination of different stakeholders across the supply chain.

5.4 Security and Privacy Considerations

This document presents the smart contracts developed in the context of the EGTN Interledger Service that manages the secure exchange of information between T&L stakeholders.

In terms of network security, most of the backend Blockchain systems are private permissioned networks where data is exchanged among trusted and identifiable participants. This is the case for the PoV and DHL, as they both use Hyperledger Fabric networks, while the PoR uses a public Ethereum testnet to upload non-confidential and anonymised shipping data.

Regarding the data exchanged between the private Blockchain systems, the EGTN Interledger Service connects to each of them through the Connectors using credentials and cryptographic keys that are natively used by the employed Blockchain frameworks (Hyperledger Fabric). This gives them read-only permissions on the ledger of the backend system. The EGTN Interledger Service acts as another typical client of the Hyperledger Fabric network taking advantage of the secure protocols employed for the communication between the nodes using mutual Transport Layer Security (TLS). On the other hand, typical Ethereum clients are used to connect to the public Ethereum testnet to listen to events generated by the Port of Rotterdam system. Extra attention should be given by administrators of both the EGTN Platform and the backend Blockchain systems to the correct management of the certificates used by the EGTN Interledger Service for the remote connection with the backend Blockchain systems.

Privacy is seriously taken under consideration in the proposed smart contracts since only non-confidential data and events are forwarded to the EGTN Interledger Service by the backend systems, according to the actors' privacy policies. The events include only hashes of anonymised data accompanied by metadata useful for identification and filtering. This ensures their security while the Blockchain itself guarantees their integrity, traceability, and non-repudiation. The EGTN Platform handles the secure retrieval of the information from the backend systems only by permissioned actors, using the hashes and the metadata stored in the Blockchain.

6 Smart Contracts and their Business Value

6.1 SLA Management

A Service Level Agreement (SLA) is a commitment between a Service Provider (SP) and a client that includes a list of objectives, services, and responsibilities that the client expects from the service provider to provide. SLAs also include metrics that measure the accuracy and extent to which SPs provide those services as well as potential penalties, contingency plans or simply alarms, if the levels of service specified by the agreement are not maintained. SLA management is the ongoing process of ensuring that all provided services and processes—including the underlying contracts— are in alignment with the agreed-upon service level targets stipulated by the contract. At present, third parties are mainly employed for the management of SLAs in various industry domains to ensure effectiveness and efficiency of the business services. Nevertheless, there is no effective supervision mechanism to monitor the third party and no efficient punishment mechanism on SLA violation.

The lifecycle of smart contracts correlates highly with the lifecycle of SLAs. They have been recently proposed to address the challenge of efficiently and securely monitoring agreements and to introduce a distributed and transparent decision-making process for SLA management in the cloud domain [10], [11], [12]. Similarly, the inherent characteristics of smart contracts can guarantee a trustworthy, distributed process that reduces time and costs spent on manual interorganisational processes in the Physical Internet paradigm. The decentralised nature of smart contracts ensures the engagement of all T&L stakeholders in the negotiation and monitoring of a contract paving the way towards a transparent and global logistics system. Data integrity is also achieved through the concurrent execution of the smart contracts by all the network participants (or most of them).

In the context of PLANET, the EGTN Smart Contracts are executed upon predefined terms and conditions – as presented in detail in *D2.17 EGTN smart contracts and associated PI motivated workflows in the context of SLA management v1* – that are already defined and agreed among all T&L participants beforehand. The EGTN Platform contains a plethora of data and intelligence coming from the various EGTN services, i.e., backed Blockchain systems, IoT data, Big Data Analytics, which are then used within the EGTN Smart Contracts to generate automatically reliable metadata and ensure a trustful contract execution.

6.2 Contribution to the PI Paradigm

As previously explained in section 3.1, the Physical Internet is a progressive logistics paradigm that aspires to lead modern T&L to an efficient and sustainable future. In 2020 a roadmap towards the PI was released illustrating the milestones and steps that need to be taken until 2040 [6]. In their report, recommendations to companies and R&D were provided, such as the recommendation to *publish mechanisms and practical examples on how to build trust among users of shared networks, platforms, and collaborative systems*. The EGTN Interledger Service along with the associated EGTN Smart Contracts provide a technical solution to achieve interconnectivity between systems of different T&L actors along with an automated and trusted collaborative solution through the generation of smart contracts.

Notably, applications of smart contracts offer tremendous business benefits to all stakeholders across the process chain, such as the following [1]:

- Reduction of manual tasks that allows stakeholders to focus on higher value tasks.
- Digitisation of processes that generate digital documents instead of paper-based ones that need to be shipped between stakeholders.
- Security, given that it is an inherent component of Blockchain technologies.
- A significant reduction in paper-based solutions paving the way for sustainable solutions.

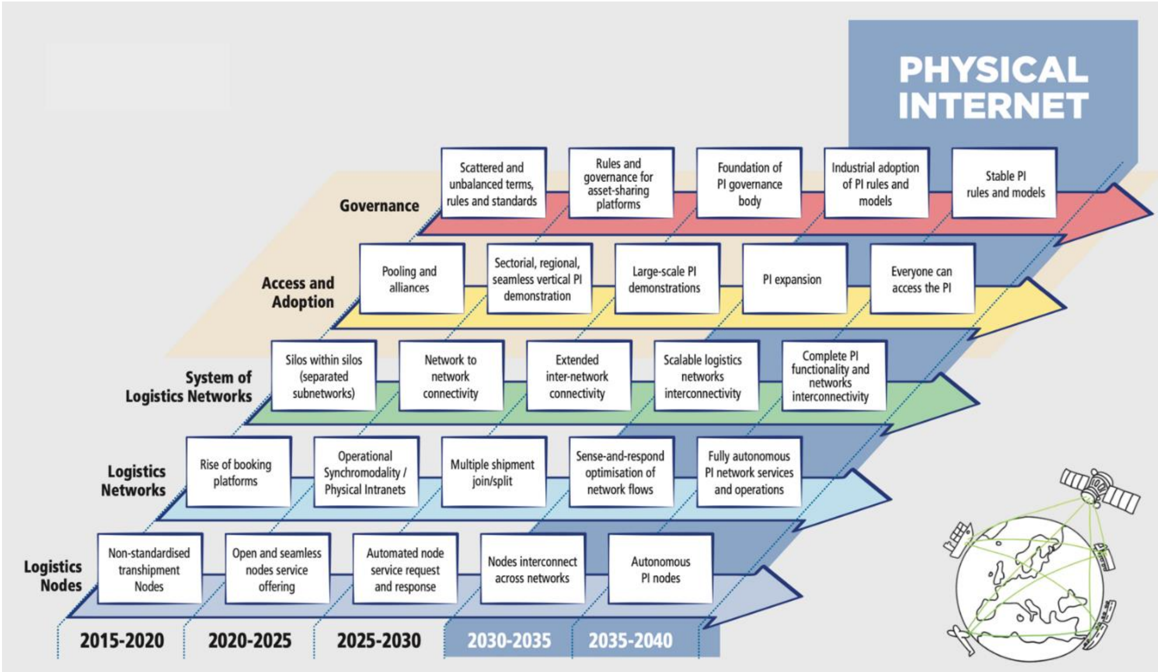

Figure 9: The Physical Internet roadmap [6].

The EGTN Smart Contracts, deployed in the context of the EGTN Interledger Service can be beneficial in the roadmap towards the PI (Figure 9) by fostering collaboration between all the different T&L stakeholders involved in the supply chain.

To sum up, as described throughout this report, these technological solutions offer several technical benefits such as the following:

- Integrity and immutability of the data throughout the entire supply chain.
- Data interoperability and interconnectivity of different Blockchain networks, thus breaking data and system silos.
- Automated and safe contract execution through the application of smart contracts.
- Reduction of overheads and time delays thanks to the digitisation of processes.
- A distributed and community-driven approach thanks to Blockchain inherent decentralised nature.

7 Conclusions

This report aims to inform the reader on the final work performed in *ST2.5.2 EGTN Smart Contracts*, which focuses on the structure of the smart contracts and associated PI motivated workflows in the context of SLA management.

The report presents in detail how SLA management is tackled in PLANET through the design and development of the EGTN Smart Contracts. These are employed to offer a technical solution to the PI workflows that were defined in close collaboration with the Living Labs. The terms and conditions of the smart contracts, as well as the data structures and functions developed in are also described. The overall solution is open-source, reusable, and can be found on Gitlab[4]. In this manner, it acts as a blueprint so that any interested parties can easily use or extend the EGTN Interledger Service or even the EGTN Platform.

The interconnection between the EGTN smart contracts and the IoT connectivity layer was presented which led to the development of automatically triggered smart contracts. Scenarios regarding the development of AI-enabled smart contracts were also developed, hoping to bring further technological and business value to the PLANET project.

The EGTN Smart Contracts, deployed in the context of the EGTN Interledger Service, offer tremendous benefits to the entire T&L sector, by fostering collaboration between all the different actors involved. These technological solutions offer integrity and immutability of the data throughout the entire supply chain, automated and safe contract execution, reduction of overheads and time delays, using a distributed and community-driven approach.

[4] https://gitlab.com/planet-h2020/egtn-smart-contracts

8 Bibliography

[1] N. V. E. B. Bertrand Copigneaux, "Blockchain for supply chains and international trade," European Parliamentary Research Service Scientific Foresight Unit, 2020.

[2] P. Kochovski, V. Stankovski, S. Gec, F. Faticanti, M. Savi, D. Siracusa and S. Kum , "Smart Contracts for Service-Level Agreements in Edge-to-Cloud Computing," *Journal of Grid Computing,* vol. 18, p. 673–690, 2020.

[3] W. Tan, H. Zhu, J. Tan, Y. Zhao, L. D. Xu and K. Guo, "A novel service level agreement model using blockchain and smart contract for cloud manufacturing in industry 4.0," *Enterprise Information Systems,* pp. 1-26, 2021.

[4] R. B. Uriarte, H. Zhou, K. Kritikos, Z. Shi, Z. Zhao and R. De Nicola, "Distributed service-level agreement management with smart contracts and blockchain," *Concurrency and Computation Practice and Experience,* vol. 33, no. 14, 2020.

[5] R. D. M. a. E. B. B. Montreuil, "Physical Internet Foundations," *Studies in Computational Intelligence,* vol. 472, pp. 151-166, 2013.

[6] M. K. a. E. H. T. Meyer, "Blockchain technology enabling the Physical Internet: A synergetic application framework," in *Computers & Industrial Engineering*, 2019.

[7] K. S. R. J. I. Y. a. M. O. H. Hasan, "Blockchain Architectures for Physical Internet: A Vision, Features, Requirements, and Applications," *Networks,* vol. 35, no. 2, pp. 174-181, 2021.

[8] T. Henry, R. Beck, N. Laga, W. Gaaloul and S. Pan, "Decentralized procurement mechanisms for efficient logistics services mapping - a design science research approach," *IEEE Computer Society Press,* 2022.

[9] S. a. T. D. a. M. D. a. M. B. a. B. E. a. H. G. Q. Pan, "Digital interoperability in logistics and supply chain management: state-of-the-art and research avenues towards Physical Internet," *Computers in Industry,* vol. 128, p. 103435, 2021.

[10] S. B. B. v. B. F. L. J. R. F. D. ‘. H. A. N. P. P. L. A. T. Eric Ballot, "Roadmap to the Physical Internet," ALICE-ETP, 2020.

[11] C. W. a. L. Graf, "Disrupting Logistics: Startups, Technologies, and Investors Building Future Supply Chain," Springer, 2021.

[12] Y. H. J. a. B.-D. P. Wang, "Understanding blockchain technology for future supply chains: a systematic literature review and research agenda," *Supply Chain Management,* vol. 24, no. 1, pp. 62-84, 2019.

[13] H. Z. K. K. Z. S. Z. Z. a. R. D. N. R. B. Uriarte, "Distributed service-level agreement management with smart contracts and blockchain," *Concurrency and Computation Practice and Experience,* vol. 33, no. 14, 2020.

[14] "Expedite trade with complete visibility and security," Tradelens, [Online]. Available: https://www.tradelens.com/tradelens-use-cases/trade-document-collection. [Accessed 2022].

[15] H. Treiblmaier, "Combining Blockchain Technology and the Physical Internet to Achieve Triple Bottom Line Sustainability: A Comprehensive Research Agenda for Modern Logistics and Supply Chain Management.," *Logistics,* vol. 3, no. 10, 2019.

[16] W. Tan, H. Zhu, J. Tan, Y. Zhao, L. D. Xu and K. Guo, "A novel service level agreement model using blockchain and smart contract for cloud manufacturing in industry 4.0," *Enterprise Information Systems,* pp. 1-26, 2021.