New ICT infrastructure and reference architecture to support Operations in future PI Logistics NETworks

D2.8 Smart PI Containers – Tracking & Reporting as a Service – Final

Document Summary Information

Grant Agreement No	769119	**Acronym**	ICONET
Full Title	New **IC**T infrastructure and reference architecture to support **O**perations in future PI Logistics **NET**works		
Start Date	01/09/2018	**Duration**	30 months
Project URL	https://www.iconetproject.eu/		
Deliverable	D2.6 - Smart PI Containers – Tracking & Reporting as a Service - Final		
Work Package	WP2		
Contractual due date	30/09/20 (as per AMD)	**Actual submission date**	30/09/20
Nature	Other	**Dissemination Level**	Public
Lead Beneficiary	NGS		
Responsible Author	Claudio Salvadori - NGS		
Contributions from	Viet Phuong Dao, Marco Spinosa (NGS), Philippos Philippou (eBOS), Kieran Flynn (IBM), David Cipres (ITA)		

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement No 769119.

Disclaimer

The content of the publication herein is the sole responsibility of the publishers and it does not necessarily represent the views expressed by the European Commission or its services.

While the information contained in the documents is believed to be accurate, the authors(s) or any other participant in the ICONET consortium make no warranty of any kind with regard to this material including, but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Neither the ICONET Consortium nor any of its members, their officers, employees or agents shall be responsible or liable in negligence or otherwise howsoever in respect of any inaccuracy or omission herein.

Without derogating from the generality of the foregoing neither the ICONET Consortium nor any of its members, their officers, employees or agents shall be liable for any direct or indirect or consequential loss or damage caused by or arising from any information advice or inaccuracy or omission herein.

Copyright message

© ICONET Consortium, 2018-2021. This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.

Table of Contents

List of Figures

List of Tables

Glossary of terms and abbreviations used

Abbreviation / Term	Description
2G/3G/4G	The second, third and fourth generation cellular network technology respectively
API	Application Programming Interface
DI	Digital Internet
BLE	Bluetooth Low Energy
EU	European Union
FSM	Finite State Machine. A finite state machine is a computation model that can be implemented with hardware or software and can be used to simulate sequential logic and some computer programs.
Geo&time-referenced	An information is geo&time-referenced when it is completed with data regarding its position and time
GET/POST/PUT/DELETE	Methods of the HTTP API
Groupage	Groupage is the same as LCL (Less than Container Load). Transporting a shipment with other goods in the container is referred to as LCL. That means that multiple LCL shipments with different Bills of Lading and different owners can be loaded in a single container. Any space used in the container is subject to a charge.
HTTP	The Hypertext Transfer Protocol is an application protocol for distributed, collaborative, hypermedia information systems (e.g., Internet)
IC	Integrated circuit
IEEE802.15	A standardised IoT protocol
IoT	Internet of Things
JSON	JavaScript Object Notation is an open-standard file format that uses human-readable text to transmit data objects
LoRaWAN	Low Power Wide Area Network IoT protocol, property of Semtech (1)
LTE Cat-M / LTE-M	It is a type of low power wide area network radio technology standard developed by 3GPP to enable IoT communications exploiting the cellular networks. It is part of 5G framework.
M2M	Machine-to-Machine
NB-IoT	Narrowband Internet of Things. It is a Low Power Wide Area Network radio technology standardised by 3GPP to enable IoT communications exploiting the cellular networks. It is part of 5G framework.
PI	Physical Internet
QoS	Quality of Service
RFID	Radio Frequency ID
WSN	Wireless Sensor Network

1. Executive Summary

Goal of this deliverable is to report the final findings and results related to the development of the IoT components already introduced within D2.6, D2.7 ICONET deliverables, entitled "Smart PI Containers – Tracking & Reporting as a Service v1" and "Smart PI Containers – Tracking & Reporting as a Service v2" respectively (in the following called 'D2.6' and 'D2.7'), and capable to satisfy both business and technical requirements and materialize the architecture described within D1.6 ICONET deliverable, entitled "Requirements and high-level specifications for IoT-based smart PI containers" (in the following called 'D1.6').

This document sum-up all the innovative contributions coming from the findings related with the realisation of an IoT-enabled PI environment, and presented in D1.6, D2.6 and D2.7. These contributions derive from significant improvements on:

1. The **data collection**, envisioning a visionary and innovative IoT architecture capable to improve the pervasivity and the interoperability with added value IoT devices and with other domains linked with logistics (e.g., ITS).
2. **The enablement of proactive PI services**, allowing the realisation of real-time decision-making actions based on real data coming from the field.
3. **The logistics network characterisation**, computing anonymised information and features capable to characterise the whole supply chain, thus providing an important ground-truth to base the PI decisions.

The main outcome of this report relates to the definition of a blueprint on how to setup an IoT-enabled PI environment, capable to facilitate end-to-end supply chain visibility and allow all the PI-users secure, ad-hoc access to data and information collected/computed by the proposed infrastructure, concurrently enabling data-oriented and fact-based decision making.

To realise this outcome two main components are developed and analysed in this report: (i) the positioning of the ICONET work regarding the IoT architecture and components in the development standardisation scenario; (ii) the methodology for data sharing and making available the information collected to the PI world in a secure and ad-hoc access, thus supporting adaptive routing functionalities, fact-based decision making, consequently driving supply chain optimisation.

Starting from the first component, ICONET has based the proposed innovative IoT architecture in a contemporary standardised environment. Particularly, we have analysed the evolution of a typical logistics and transportation scenario and compared it with the architecture envisioned in D1.6. The outcome of this work has demonstrated that a standardised environment has to consider multiple protocols for data collection, to facilitate the prospect of composing and implementing value-added functionalities:

1. The optimisation of the data dispatchment (e.g., lower power consumption, lower cost), thus reducing the operative costs of maintaining logistics dedicated IoT devices on premises (e.g., on the container, on the warehouse(s) etc.).
2. The realisation of dedicated IoT networks interoperable with standardised protocols, ready to interact with special IoT devices and capable to provide added value information (e.g., the pollution).
3. The interaction with other related domain (as for example, smart cities or transport) collecting (e.g., traffic congestion information coming from trucks OBU, to support the routing operations as well as the decision making) or providing added value information (e.g., communicating to the OBU regarding the presence of harmful compounds generated by the transported goods).
4. The improved monitoring granularity, giving the possibility to track and collect information regarding encapsulated goods, enabling smart routing and decision making.

Furthermore, this deliverable also expands on the interaction with both the PI world and PI users, enforcing compliance with a standardisation framework related with logistics digitalisation. The Cloud IoT platform is the components in charge of implementing the IoT brokerage and the data persistence service as well as the interaction interfaces. In D2.8, it is described on its final release, providing a set of APIs, enabling the data sharing with the other components of the PI as well as with users exploiting human-readable interfaces (i.e., GUI and PDF reports). Data are presented as they are collected (i.e., raw data), as well as aggregated, exploiting the dedicated analytics engine. In this manner, the PI users have available an important instrument to completely monitor the supply chain and the goods flowing in it, enabling proactive data-oriented and fact-based decision making.

Summing up, the deliverable achieves the following key objectives:

O1. It provides an overview of the IoT enabled PI environment, summarising all the steps followed to arrive to its current design.
O2. It positions the ICONET project in an in-fieri standardisation framework.
O3. It describes a blueprint on how to set-up the IoT enabled PI environment, highlighting the interaction models between the components, and the API to interact with the PI users.
O4. It outlines results generated by tracking & tracing as well as analytics services related with the LLs.
O5. It reports the development status of each component identified within D2.6 and D2.7, to realise the architecture proposed in D1.6.
O6. It highlights the work shift and issues triggered by the COVID-19 pandemic, especially in regards with the physical deployment of the IoT devices.

This report is released, as per the recently submitted amendment, two months after the initially planned date, due to the COVID-19 pandemic. This pandemic has affected IoT development, but mostly the deployment of real devices in the Living Labs. Finally, this document also reports, some of actions put in place to mitigate the impact of the pandemic and effectively demonstrate the overall system performance.

2. Introduction

The main objective of this deliverable is to report on the development of advancements regarding the IoT infrastructure, following the guidelines defined in D.1.6 In fact, this deliverable aims at showing the implementation of the strong visionary innovative IoT architecture, compliant with well-recognised standards, out-of-the-box interoperable both with ICONET's PI Services as well as other existing services, allowing seamless and cost-efficient implementation of end-to-end visibility necessary for the establishment of any PI Network.

Moreover, this report and finalise the work described in D2.6 and D2.7, showing the importance of IoT as PI enabling technology and highlighting the solutions' development beyond:

1. The realisation of the complete visibility of the supply chain, implementing a hyper-connected and ubiquitous IoT environment.
2. The enablement of a continuous tracking and monitoring of goods, assets and means, equipping those with smart IoT devices and exploiting the hyper-connected and ubiquitous IoT environment.
3. The (real-time and on-demand) reporting for all the PI users, implementing ad-hoc and secure Internet transactions.
4. The provision of the system performance based on data retrieved by the deployed IoT devices and the simulator.

D2.8 considers the development of the devices capable of enabling the IoT connectivity within all the PI scenarios (containers, hubs, warehouses, …). These devices will be characterised in terms of:

1. Exploitation in operative environments, defining different business models to enable a ubiquitous monitoring of the whole supply chain.
2. The final hardware and software architecture of each components described in D2.6 and D2.7.

D2.8, originally due at M23 (end of July 2020), is finally released at M25 (end of September 2020) with a delay of 2 months due to the COVID-19 pandemic as explained further on.

2.1 Deliverable Overview and Report Structure

This report describes the completion of technical work done for implementing the specifications and the architecture described within D1.6 and concludes the work done in D2.6 and D2.7. Particularly, it will describe the complete architecture to implement the Cloud IoT service, thus all the functionalities to collect data from the field, toward its dispatchment to other systems, networks, decision making mechanisms, etc.. The document is structured as follows.

Section 3 - IoT enabled PI environment overview describes an overview of both the business requirements coming from D.16, the derived architecture and the components to satisfy these, highlighting the standardisation environment, the interoperability and the security issues.

Section 4 - Toward an innovative Supply Chain complete visibility, describes the innovative aspect derived by the realisation of the IoT enabled PI environment to implement the Supply Chain complete visibility.

Section 5 - How to set-up the IoT-enabled PI environment provides a blue-print on how to realise and manage a connected environment implementing the complete visibility of the supply chain.

The following sections regard the technical details related to the components developed within the ICONET project.

Section 6 - The Cloud IoT Platform, details the final details regarding the platform to collect and store the data, compute added value information and share reports.

Section 7 and 8 - The Tracker and the Smart Router and **The Smart Gateway** provide the technical descriptions regarding the final release of the IoT devices developed and considered within the ICONET project.

Conclusion ends the document.

2.2 COVID-19 pandemic delay and mitigation activities

COVID-19 pandemic and the consequent lock-down have caused several problems and delays in the realisation, but especially in the deployment of the IoT devices in the LLs.

The lock-down has imposed NGS to work at home and to organise on-the-fly the management strategies. For these reasons, the development activities related to T2.3 are delayed for almost 2 months.

The same problems and the derived lack of coordination with the large intermodal transportation company (ITC – selected outside the consortium partners), considered on supporting LL2 operations, affect the deployment of the IoT devices. The scenario was changed due to the pandemic:

1. The selected containers are left stopped at the ITC facility, leaving the devices on. The battery has terminated May 9th, 2020.
2. As consequence of the pandemic, because internal directions of the ITC, the preliminary agreement cannot be maintained. In this scenario, the two selected containers are left to move all along EU, with the condition that NGS and ICONET can exploit only the routes along the logistics network of a global consumer goods company covering Belgium, France, Italy, Switzerland and the United Kingdom. The data must be maintained confidential inside the project, and this cannot be disseminated.
3. The selected two containers start their trip all along EU with the IoT devices off, because the batteries were not charged.

Because all these delays, the IoT devices' deployment was disrupted and the latest conditions introduce delays in the installing process (recharging, setting up, configuring). As a result, we decide to accelerate the process to minimise the delays the following mitigation solutions:

1. Implementation, with the support of ITA, of a cloud component capable to generate coherent routes all along EU and to emulate the behaviours of an IoT device – i.e., implementing a secure transaction toward the Cloud IoT server. See Sec. 5.2.3 - Simulator. This work was done to support and accelerate the development tasks related with the realisation of the Cloud IoT platform.
2. Sending of other two devices for equipping other two different containers, to increase the amount of data collected from the supply chain real scenario (sent August 10th).

Finally, COVID-19 pandemic has blocked the development and the installation of a pollution measuring IoT devices in the LL1 context, to be deployed within the Port of Antwerp landside. The objective of these devices were to assist PoA decision making processes in the environmental impact reduction, providing pervasive and effective measurements of the pollutants concentration within the port landside.

3. IoT enabled PI environment overview

In this section, the final achievements to satisfy the outcome derived from D1.6 are detailed, completing the findings described in D2.6 and D2.7. The most important achievement we want to reach is the implementation of the supply chain complete visibility, allowing the continuous and real-time monitoring of goods, assets, means, and infrastructure related to the PI world. The ICONET project is demonstrating the PI performance considering 4 LLs summarised in Figure 1. The goods in output from the factories are encapsulated in PI Container and transported in a multimodal PI corridor (LL2, leads by INL) toward the PI hubs in charge of managing its dispatchment (LL1, leads by PoA). Finally, goods are stored in “proxy[1]” PI warehouses (LL4, leads by SB) and then dispatched toward the final destination, implementing efficient e-commerce services (LL3, leads by SON).

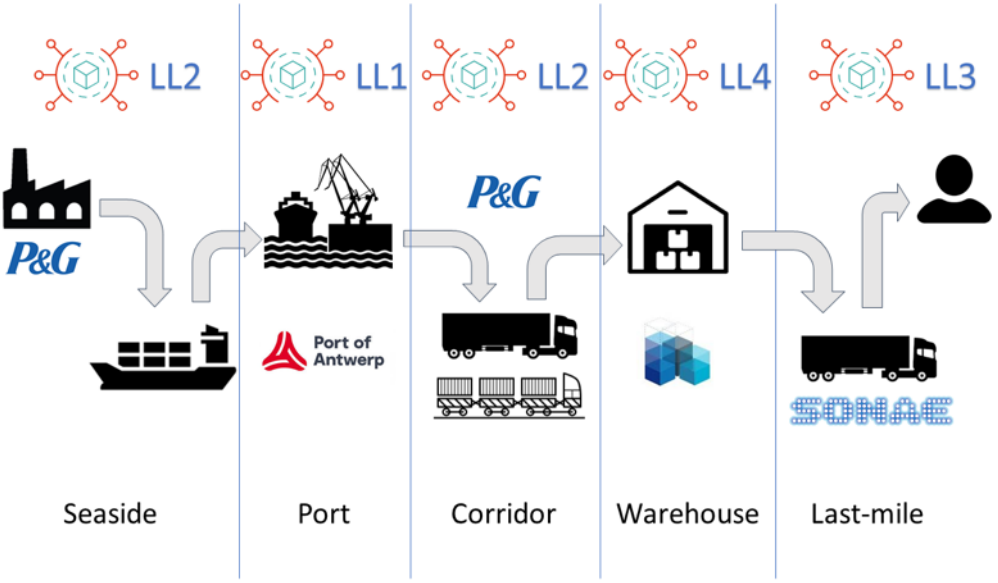


Figure 1 PI seen by the ICONET perspective

Transforming the supply chain in a fully monitored environment means having under remote control all the logistics environment: the PI packets (at different level of granularity, e.g., Smart PI Container, Smart PI Pallet, Smart PI packet), the PI assets (e.g., the container itself, the trucks, …), and the PI infrastructure (e.g., road congestion). In this scenario, all the business requirements elicited in D1.6 will be satisfied, thus providing to the PI world all the instruments to implement all the functionalities of an efficient, optimised and green logistics as well as better tractability of the products and events referring to the products

This section aims at summarising this environment providing a brief outline of the business and technical requirements, and of the architectural findings described in D1.6, suggesting their possible realisation exploiting the IoT enabling technology. Moreover, it will provide the positioning of these finding in the standardisation framework, discussing regarding interoperability issues and data usability.

3.1 Main business requirements overview

In Table 1, the summary of the most significative business requirements from D1.6 is provided, in line with UN/CEFACT guidelines (5). It is important to highlight how the **complete visibility of the supply chain** is the key requirements between the following, since it enables:

1. The implementation of more optimised and less expensive logistics transactions, also reducing their environmental impact. In this scenario, an effective resources and assets management will be enabled.

[1] We use the Digital Internet term of proxy. In DI, a proxy is a server application that behaves as an intermediary for requests from clients seeking resources from other Internet servers that provide those resources. Proxies aims at the simplification and optimisation of the complexity of requests.

2. The realisation of effective routing (or re-routing) and decision-making policies, toward an improvement of the determinism in the shipment, including facilitation of synchromodality and handling of sensitive or dangerous goods.
3. The implementation of interoperable added value services, to increase the customer satisfaction thus gaining competitive advantages.
4. Increase the productivity, optimising the supply chain thus the production lines.

Table 1 Business requirement summary

Requirement Name	Description
Affordable system and integrability	The IoT physical infrastructure has to be affordable for all the users. As a service approach has to be preferred to the device selling. The IoT service must provide interfaces that allows an affordable and secure integration with third parties' software (e.g., Port's RMS, Warehousing, Traffic control, Resources status, TMS etc.), respecting the privacy issues.
Easy and not invasive installation – Easy maintenance	The logistics operators are not expert in technology. The system has to have an easy installation process, in terms of physical deployment, maintenance and configuration.
Supply chain visibility at multiple layers - Supply chain digital twin	The IoT will provide a reliable, low-cost end-to-end and real-time visibility of the whole supply chain (corridors, warehouses, hubs). The visibility has to be implemented with an improved granularity for all the encapsulation layers (e.g., PI container, pallet, packet). In this scenario, each stakeholder involved in the logistics transaction can access to IoT devices deployed at every level of the encapsulation stack (e.g., PI container, pallet, packet).
Localisation and inventory of goods and products	The IoT will provide the position of certain goods at a certain time at multiple layers (e.g. PI container, pallet, packet). In this scenario, an advanced track&trace service can be provided, providing real-time encapsulation information up to the goods layer.
Localisation and monitoring of assets - Assets' management optimisation	The IoT will be provide the connectivity for monitoring assets connected to the logistics operations (e.g., shelf, pallets, crates, containers …), thus providing information regarding their position and status. In this scenario, an improved assets' management will support the optimisation of the operational efficiency.
Ensure goods integrity and safety	The IoT will monitor the status of the goods in terms of integrity or safety (e.g., cold chain monitoring) at multiple layers (e.g., PI container, pallet, packet).
Data-oriented and fact-based decision making and business intelligence	Exploiting the data retrieved the IoT environment, Big-data analysis techniques can be implemented to generate knowledge and models to support the decision-making implementing Data-oriented and fact-based business intelligence
Accurate ETA prediction	The storage of bigdata generated by the IoT devices, can support the computation of an accurate ETA prediction.
Maximise Operational Efficiency and Capacity, Revenue Generation	Improved efficiency improving the capacity and the efficiency of the intermodal corridors, hubs, and of the warehouses, thus reducing the costs and/or increasing the profits. Examples: (i) Increasing the transport load factor, thus avoid empty trips; (ii) implementing a reliable and up-to-date warehouse management; (iii) Optimised and deterministic scheduling and organisation uploading/downloading/crossdocking operations in PI hubs.
Increase Customer Satisfaction and Gain competitive advantage	Implementing a reliable, secure and less expensive logistics services, and offering an advanced information sharing service capable to provide an accurate end-to-end real-time visibility of the goods along the supply chain and accurate predictions. Moreover, it allows to gain competitive advantage by differentiating against similar supply chain actors
Fault liability	Exploiting the information coming from the IoT, the liability of certain event can bring back to the effective party.

The requirements analysis was the result of an extensive study of the recipients of this report and stakeholders/users in the supply chain arena and detailed work may be found in D1.6.

3.2 Main technical requirements and architecture overview

Summarising the outcomes derived from D1.6, to realise an open scalable, interoperable plug-and-play and easy-to-use IoT solution framework tailored for the PI, the following requirements must be satisfied:

- **IoT enablement**. The IoT devices and components must be capable of enabling IoT environments, thus connecting the physical world with the digital one, implementing the digital twins functionalities.

Composability and pervasiveness. IoT must allow the PI-packets encapsulation, implementing the Recursive Gateway-mediated Edge Connectivity and Management pattern, depicted Figure 2. This configuration must allow a hierarchical monitoring of PI-goods, PI-containers and PI means. IoT must allow a pervasive monitoring (e.g., also inside the container, where the wireless/5G connectivity are un-available), thus answering the questions “Where?”, “When?” and “How?” for all the encapsulation levels (see

- Table 2), and enabling the supply chain complete visibility.
- **Interoperability**. The IoT environment must be interoperable with different set of IoT protocols (i.e., technical interoperability) and with the stakeholders’ platforms (i.e., syntactic interoperability).
- **Easy use maintenance and integration.** The installation and the data access must be easy and plug&play, the maintenance reduced, thus the battery duration is an essential point.
- **Cloud based platform,** capable to implement the functionalities of data persistence and analysis, as well as a catalogue of APIs for easy integration into different platforms.

Table 2 “Where?”, “When?” and “How?”

Where?	Where are the goods located?	Providing information regarding the position of the goods, also adding information regarding the encapsulation.
When?	At what time?	Provide information regarding the date and the time of the sampling.
How?	How is their status?	Provide punctual added value measurements regarding the storage of the goods.

To realise such requirements, we have designed the architecture depicted in Figure 2, that allows the implementation of the following functionalities:

1. **Realisation of IoT networks of networks**. In this manner an IoT network can cooperate with another one, allowing the implementation of a hierarchical monitoring. This possibility allows the creation of local networks in charge of monitoring certain PI elements: for example, the internal monitoring of the goods encapsulated within the PI containers can be enabled, thus improving the supply chain visibility coverage and granularity (e.g., enabling the monitoring of connected pallets) and tracking the encapsulation evolution. For example, in Figure 3 a hierarchy of IoT devices is shown enabling the monitoring the encapsulation hierarchy of the goods.

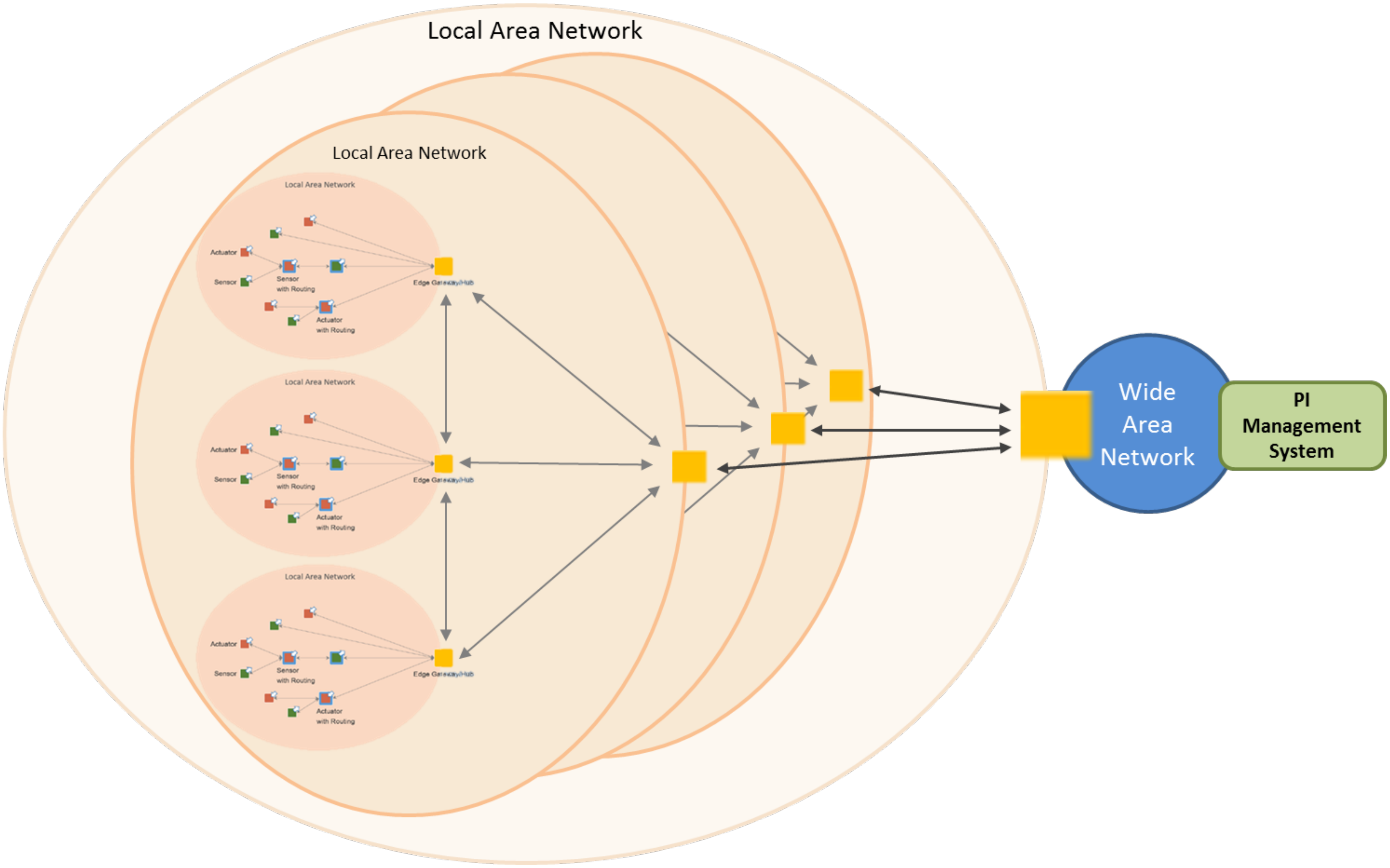
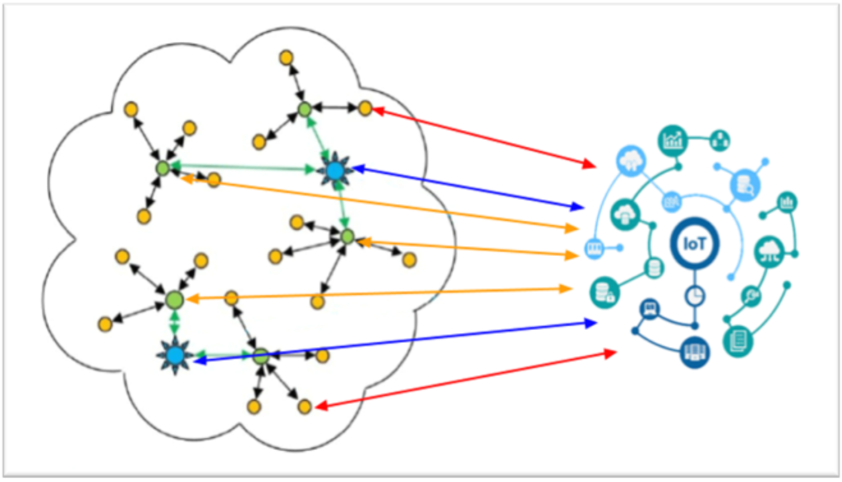


Figure 2 Recursive Gateway-mediated Edge Connectivity and Management pattern



🟠	PI-packet	IoT nodes
🟢	PI-container	Smart Router
🔵	PI-mean	Smart Gateway

Figure 3 Hierarchical monitoring: a hierarchy of IoT devices

2. **Opportunistic routing**. An IoT device, equipped of more than one network interfaces, can select between the available ones the most convenient. As matter of example, an IoT devices can select the cheapest or the less power intensive connection, or, inside the container (done in metal, thus behaving as a Faraday cage[2]), can cooperate with a higher hierarchy level device, capable to implement communication routes toward the Cloud. The exploitation of multiple IoT networks interfaces improve also the ubiquitousness characteristics of the system.

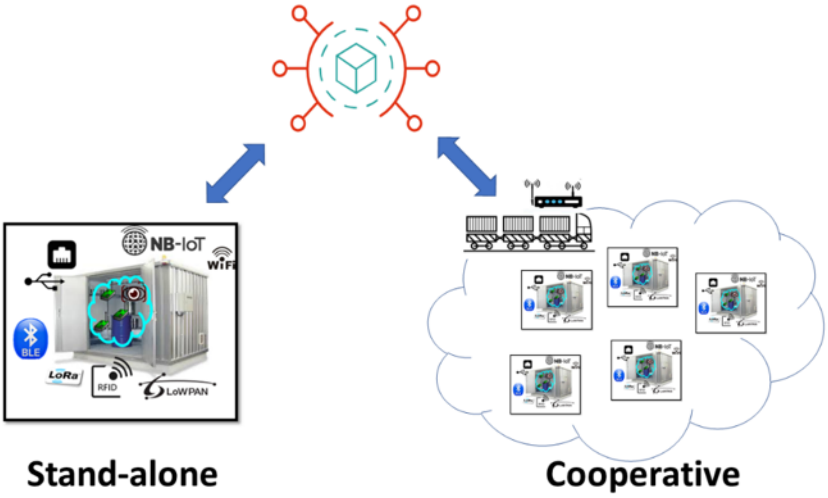


Figure 4 Opportunistic routing

As depicted in Figure 3, each element of the PI-enabled logistics (e.g., PI-packet, PI-container, PI-mean, …) can communicate exploiting the opportunistic routing, thus participating in the generation of IoT networks of networks. This approach improves the coverage and the pervasiveness of the IoT network allowing the implementation the complete visibility of the supply chain, communicating in real-time added value information regarding the goods and their encapsulation with improved granularity.

[2] A Faraday cage is an enclosure capable to block electromagnetic fields. Metal enclosure as containers are Faraday cages.

3.3 Standardisation reference framework

In this section, the reference standardisation scenario for the realisation of supply chain complete visibility is described. Particularly, we will consider the standardisation path implemented by DCSA, that has been released in May 2020, regarding the realisation of the IoT connected container, the ISO suite to define the communication stack for Intelligent Transport Systems (ITS), and the 5G. Both these standardisation experience are taken into account in the realisation of the Smart PI Container/Smart PI Pallet/Smart PI Mean (see Sec. 3.5) and the devices for implementing them (see Sec. 7 and 7).

3.3.1 DCSA standard for digital containers

DCSA (Digital Container Shipping Association) is a non-profit organisation established in 2019 by several of the largest container shipping companies with the mission to drive technology standards to enable carriers to bring innovative solutions to market. Their main objective is to define standardised approach to provide information throughout the container journey overtaking the lack of interoperability between different IoT solutions (6). In our point of view, the compliancy with this standard is important since it is released by the industry for the industry, to satisfy a real hot topic in the logistics domain (the complete visibility of the supply chain) toward its optimisation in terms of costs and environmental impact.

The “DCSA Gateway Connectivity Interfaces standard” (7) has been released at the beginning of Q2 2020 and represent the first step towards solving this lack of interoperability. In fact, its scope is to standardise a set of Physical and Media Access Control (PHY+MAC) layer protocols to interconnect containers, allowing the data collection and dispatchment toward remote control rooms. Because the considered environment (the logistics container world, so moving object detached from the infrastructures), the selected protocols will be wireless and low power consumption, allowing the autonomous supply exploiting batteries.

The standardisation document is organised in 3 Use Case (UC) listed below:

UC_1. Reefer container sensor & other data monitoring, tracking and remote control.
UC_2. Dry container sensor data monitoring and tracking.
UC_3. Automatic electronic container registration.

Particularly, UC1 and UC2 are related to the tracking and monitoring of containers as depicted in Figure 5. DCSA envisions the possibility of deploying gateway on vessels, on landside means and on ports/terminals/container yards capable to gather added value information for tracking and monitoring the containers.

The scenario proposed by DCSA suggest how to dimension the static gateways installed on the infrastructure, and, therefore, the IoT devices (for tracking and monitoring) to be installed on the containers. For this reason we can extrapolate that the envisioned devices compliant with the DCSA standard must consider almost one of the protocols described in Figure 6: cellular (2G, 4G, LTE-Cat. M, NB-IoT), LoRaWAN and BLE. Of course, the landside IoT devices equipped with cellular technologies can directly communicate toward the remote-control rooms exploiting the network made available by mobile network providers.

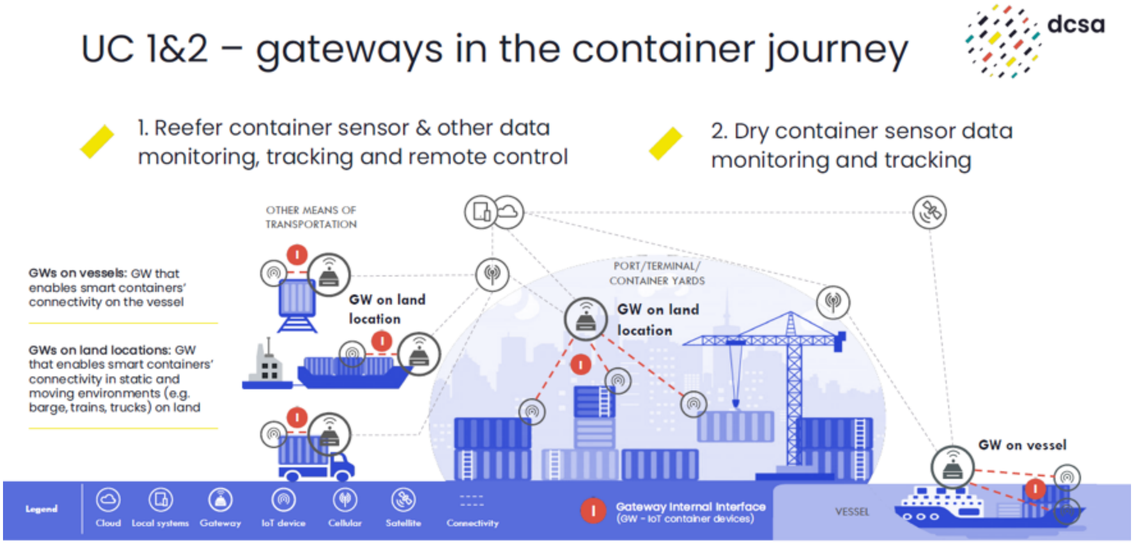
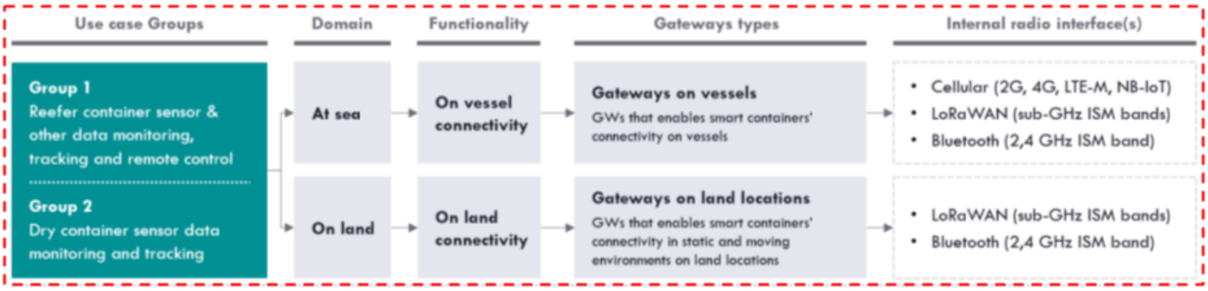

UC 1&2 – gateways in the container journey
dcsa
1. Reefer container sensor & other data monitoring, tracking and remote control
2. Dry container sensor data monitoring and tracking
GWs on vessels: GW that enables smart containers' connectivity on the vessel
GWs on land locations: GW that enables smart containers' connectivity in static and moving environments (e.g. barge, trains, trucks) on land
OTHER MEANS OF TRANSPORTATION
GW on land location
PORT/TERMINAL/ CONTAINER YARDS
GW on land location
GW on vessel
VESSEL
Legend
Cloud
Local systems
Gateway
IoT device
Cellular
Satellite
Connectivity
Gateway Internal Interface (GW - IoT container devices)

Figure 5 DCSA environment and connections



Use case Groups	Domain	Functionality	Gateways types	Internal radio interface(s)
Group 1 Reefer container sensor & other data monitoring, tracking and remote control	At sea	On vessel connectivity	**Gateways on vessels** GWs that enables smart containers' connectivity on vessels	• Cellular (2G, 4G, LTE-M, NB-IoT) • LoRaWAN (sub-GHz ISM bands) • Bluetooth (2,4 GHz ISM band)
Group 2 Dry container sensor data monitoring and tracking	On land	On land connectivity	**Gateways on land locations** GWs that enables smart containers' connectivity in static and moving environments on land locations	• LoRaWAN (sub-GHz ISM bands) • Bluetooth (2,4 GHz ISM band)

Figure 6 Considered IoT protocols in UC1 and UC2

3.3.2 ISO standard for ITS station

The ISO TC204 (8) WG16 has defined a standardised ITS communication stack, also known as CALM for C-ITS (Communications in Cooperative Intelligent Transport Systems), depicted in Figure 7 (to see the complete standardisation scenario in ITS check this link (9)). The proposed stack supports several physical communication media, that can be used simultaneously for different types of applications (ITS safety applications, ITS non-safety applications, and all legacy Internet applications). In this scenario, IoT can provide a great benefit in the ITS environment on integrating added value information sources as sensor nodes deployed along the road as well as on the vehicles. For this reason, two standardisation actions were done in 2016 to integrate IoT within the CALM stack: ISO 19079:2016 (10) and ISO 19080:2016 (11). In this view, it is possible to integrate in both Road Side Units (RSU) and On-Board Units (RSU) the 6LoWPAN+CoAP standard on top of IEEE802.15.X MAC and PHY layers (i.e., IEEE802.15.1-BLE and IEEE802.15.4) as depicted in Figure 8. Regarding the logistics and the PI domains, the interoperability with this standard suite is very important since it will be the base of the fully connected vehicle: in this manner the smart container can connect and share information with the OBU and then with the ITS infrastructure, thus using such infrastructure to dispatch the data collected with the remote Cloud IoT Platform, as well as feeding the transport side with added value data (e.g., notifying that the container is encapsulating dangerous compounds).

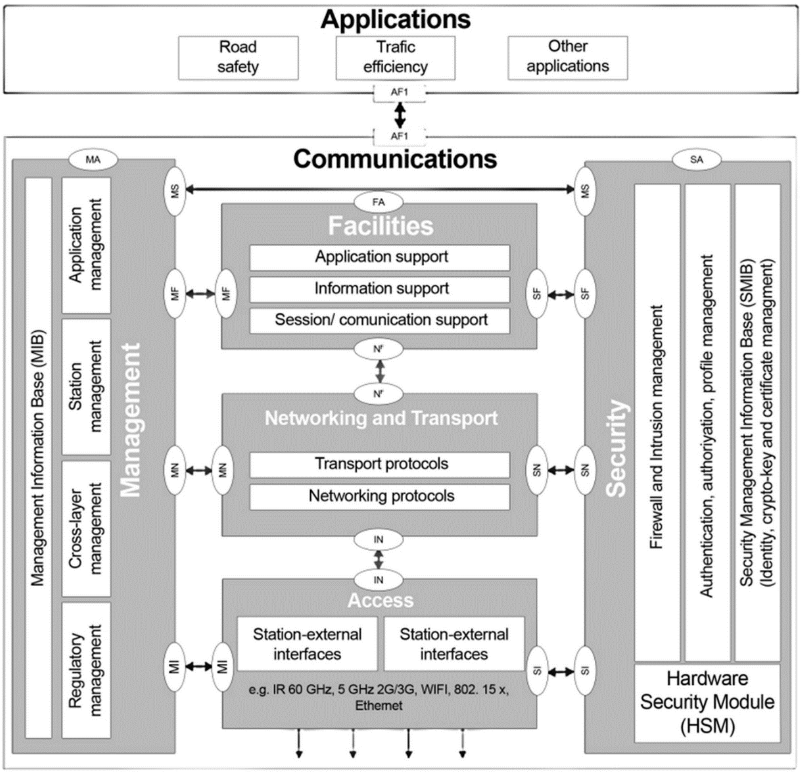
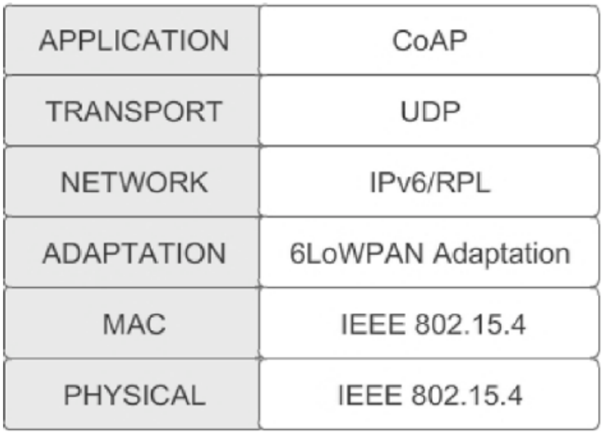

Applications
Road safety
Trafic efficiency
Other applications
AF1
AF1
Communications
MA
SA
MS
MS
FA
Facilities
Application support
Information support
Session/ comunication support
MF
MF
SF
SF
N'
N'
Networking and Transport
Transport protocols
Networking protocols
MN
MN
SN
SN
IN
IN
Access
Station-external interfaces
Station-external interfaces
e.g. IR 60 GHz, 5 GHz 2G/3G, WIFI, 802. 15 x, Ethernet
MI
MI
SI
SI
Management Information Base (MIB)
Application management
Station management
Cross-layer management
Regulatory management
Management
Security
Firewall and Intrusion management
Authentication, authoriyation, profile management
Security Management Information Base (SMIB) (Identity, crypto-key and certificate managment)
Hardware Security Module (HSM)

Figure 7 CALM C-ITS stack



APPLICATION	CoAP
TRANSPORT	UDP
NETWORK	IPv6/RPL
ADAPTATION	6LoWPAN Adaptation
MAC	IEEE 802.15.4
PHYSICAL	IEEE 802.15.4

Figure 8 6LoWPAN+CoAP reference stack

3.3.3 5G and NarrowBand-IoT

5G is the fifth-generation technology standard for cellular networks, thus the planned successor to the 4G networks which is the current connectivity exploited by most of the mobile-phones. Its main improvements compared with the previous protocols are (see also Figure 9):

1. **Mobile IoT/Massive IoT/LPWAN**. In fact, it aims at providing improved network coverage, long device operational lifetime (implementing improved power saving functionalities) and a high density of connections (improve the scalability managing a huge amount of IoT devices).
2. **Critical communications**, implementing high performance, ultra-reliable, low latency industrial IoT and mission critical applications.
3. **Enhanced Mobile Broadband**, implementing improved performance and a more seamless user experience accessing multimedia content for human-centric communications.

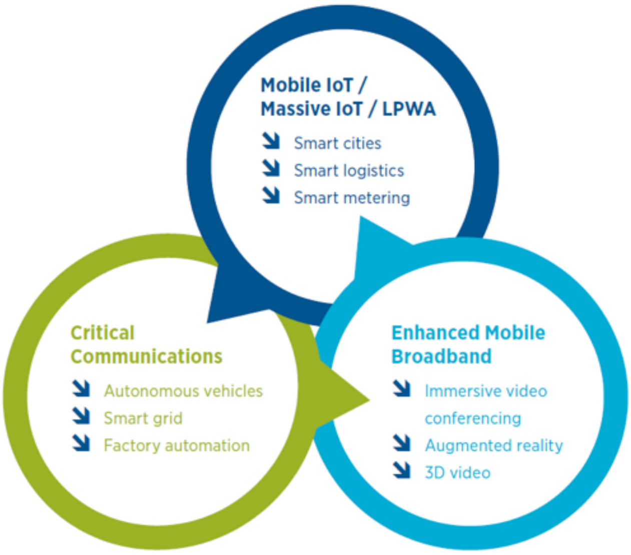


Figure 9 5G 3 directions (12)

NarrowBand-Internet of Things (NB-IoT) is a standards-based low power wide area (LPWA) technology developed to enable a wide range of new IoT devices and services. Its specification was frozen in 3GPP Release 13 (LTE Advanced Pro), in June 2016 (13) . NB-IoT significantly improves the power consumption of user devices, system capacity and spectrum efficiency, especially in deep coverage. Particularly, the energy saving functionalities are improved with respect to the standard mobile protocols, introducing Power Saving and Extended Discontinuous Reception Mode (14) .

In this scenario, considering the guidelines coming from the GSMA white paper of April 2018 (12), NB-IoT can be considered as forerunner protocol toward 5G. In fact, Figure 10 shows the various 5G network components that are built up and deployed over time. It highlights that NB-IoT, already (partially) operational, is considered (together with LTE-M) as the 5G technology capable to enable the **Mobile IoT/Massive IoT/LPWAN**, and it will coexist with the other 5G components (i.e., enhanced mobile broadband and critical communications) when these will be deployed.

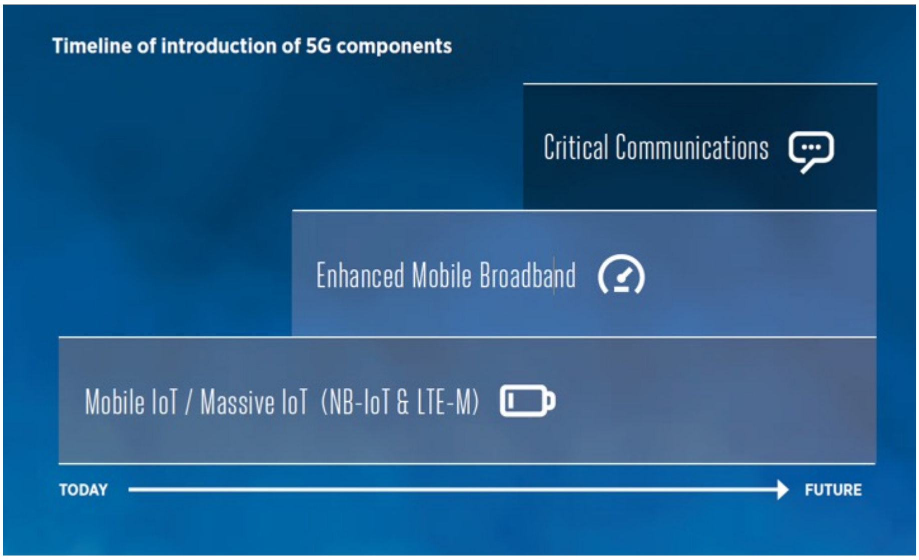

Timeline of introduction of 5G components
Critical Communications
Enhanced Mobile Broadband
Mobile IoT / Massive IoT (NB-IoT & LTE-M)
TODAY
FUTURE

Figure 10 Timeline of introduction of 5G components (GSMA, 2018)

3.3.4 Standardisation framework vs. ICONET IoT architecture

The ICONET IoT architecture complies with the standardisation framework presented in the previous sections.

5G connectivity boosts the proposed architecture realising the natural pervasive communication infrastructure for implementing stand-alone transactions toward the Cloud. In this scenario, the exploitation of NB-IoT and LTE Cat-M allows the devices installed on top of the Smart PI containers to implement transactions toward the remote platforms with a reduced the battery consumption (thus reducing the costs) comparing with the previous cellular standards.

On the other hand, the ICONET IoT Architecture comply with DCSA standardisation scenario. In fact, DCSA standard Release 1 defines the IoT connectors for the realisation of gateways capable to collect data from (Smart/Connected) Gateways, both in the seaside (i.e., deployed in the vessels) and in the landside (i.e., deployed in the tracks, trains, infrastructures). Particularly, it envisions the exploitation of mobile stand-alone communication (considering 2G, 4G, NB-IoT and LTE Cat-M) as well as cooperative communication (i.e., LoRaWAN and BLE). This scenario leaves the possibility to implement multi-protocol devices capable to compose and select network, as suggested in the proposed architecture.

The compliancy with ISO CALM stack, can be seen as a generalisation of the DCSA standard: connected vehicles (in this case, trucks) enables IoT interfaces to cooperate with, allowing an improvement of the performance of both the transport and the logistics systems (as well as smart cities systems).

3.4 Data collection and sharing: interoperability and data usability

The IoT environment proposed in ICONET must implement an interoperable framework where devices of different stakeholder can cooperate in an open and scalable scenario. As highlighted in D2.6 and D2.7, the IoT framework will be one of the contributors of the PI digital infrastructure, thus it will expose interoperable APIs capable to satisfy up to the syntactic interoperability level (see Figure 11).

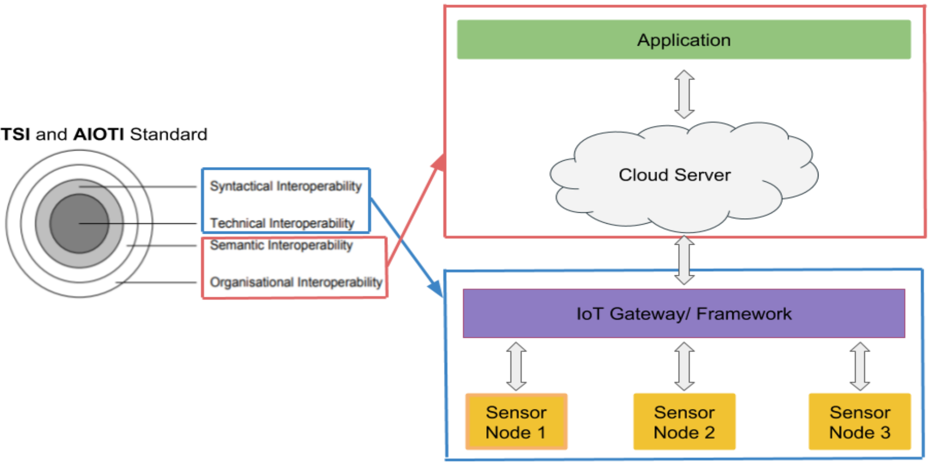

ETSI and AIOTI Standard
Syntactical Interoperability
Technical Interoperability
Semantic Interoperability
Organisational Interoperability
Application
Cloud Server
IoT Gateway/ Framework
Sensor Node 1
Sensor Node 2
Sensor Node 3

Figure 11 Mapping IoT framework on the ETSI and AIOTI Standard

In this scenario, the Cloud IoT is considered the main interoperability centre, since devices on the premises do not have available enough computational capacity to implement different interoperable APIs. In thi scenario, the Cloud IoT platform oversees:

1. Collecting data from devices of different parties in a secure and ad-hoc manner.
2. Dispatching data toward all the PI stakeholders involved in the PI transaction, in a secure and ad-hoc manner.

In the following sections, the interoperability issues are discussed focusing on:

1. How IoT devices will cooperate with the Cloud IoT platform interoperating up to the semantic level (see Sec. 3.4.1 and 3.4.2) considering a secure and ad-hoc approach (see Sec. 3.4.3).

2. How the Cloud IoT platform makes the data available to the PI world, in terms of interoperability and security (see Sec. 3.4.4), and the considered methodologies for data sharing and usability (see Sec. 3.4.4 and 3.4.5).

3.4.1 IoT devices interoperability – Technical and syntactic interoperability

As discussed in D1.6, while the direct Internet transactions (i.e., using Ethernet and Wi-Fi) implement the cross-platform interoperability design pattern (see Figure 12), while the mobile dispatchments consider the a Platform-to-Platform pattern, having in the middle the mobile network operators platform in charge of dispatching the data coming from the mobile network toward the IP address of the Cloud IoT platform (see Figure 13).

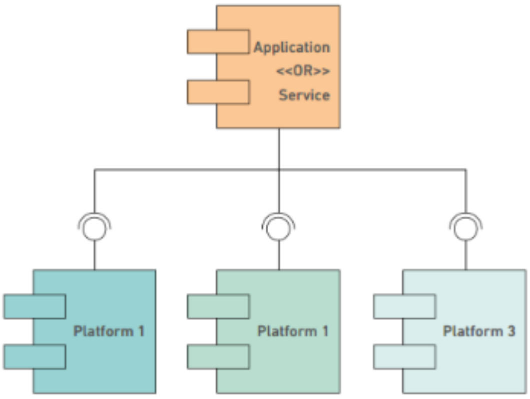


Figure 12 Cross-platform interoperability pattern

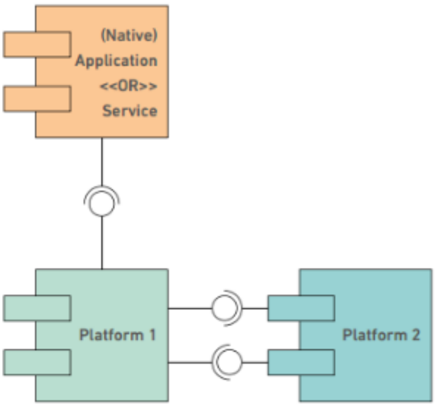


Figure 13 Platform-to-Platform interoperability pattern

The introduction of these approaches derives from the following considerations:

1. The IoT Device can select the best path to dispatch the data to reduce the power consumption, thus reducing its maintenance costs. In fact, usually mobile connectivity is more power consumption intensive with respect to the other IoT protocols considered by DCSA (i.e., BLE and LoRaWAN) or by ISO in the ITS domain (i.e., 6LoWPAN+CoAP over IEEE802.15.4).
2. Selecting a local gateway can also provide added value information regarding the encapsulation of the goods/pallets/containers.

3.4.2 IoT devices interoperability – Semantic interoperability

Following the guidelines of the deliverable D2.1 of the Horizon 2020 project CHARIOT (The CHARIOT project, s.d.), in D2.7 we have considered Sensor Markup Language (SenML) (RFC8428, August 2018) to improve the interoperability features of the proposed IoT environment toward the semantic level.

SenML is an emerging drafted standard from IETF for representing sensor measurements and device parameters. It defines a minimal and self-describing data-model capable to integrate data and meta-data (to describe the data), thus capable to satisfy the IoT efficiency constraints, as well as to realise the semantic interoperability among IoT devices.

On the other side, DCSA is going to publish the Release 2 of its IoT standards program to in depth analyse higher protocols layers than the one proposed in first release, as well as data structure and handling, as depicted in Figure 14 (DCSA, 2020) (planned to release in Q4 of 2020).

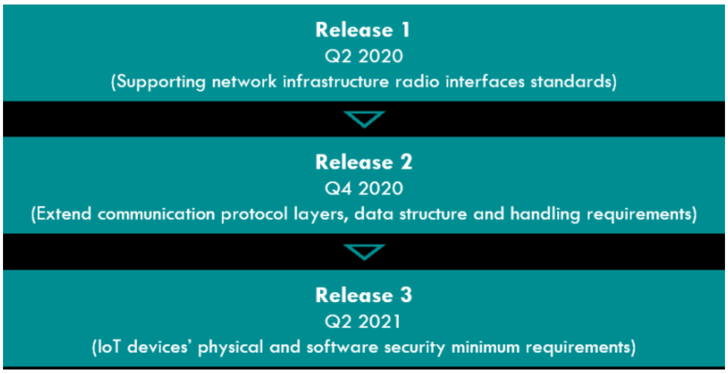

Release 1
Q2 2020
(Supporting network infrastructure radio interfaces standards)
Release 2
Q4 2020
(Extend communication protocol layers, data structure and handling requirements)
Release 3
Q2 2021
(IoT devices' physical and software security minimum requirements)

Figure 14 Overview of planned releases for DCSA's IoT standards program

3.4.3 IoT devices – Security

On the other side, the access the open, scalable, and interoperable IoT environment must be ruled with security constraints to collect reliable data coming from authenticated data sources (i.e., IoT devices). For this reasons, a secure and ad-hoc access is defined for IoT devices that want to cooperate in the ICONET IoT environment considering HTTPS transactions and the "standard basic authentication", that will be further detailed in Sec. 6.7 - Security and ad-hoc access.

Future investigations are considered after the conclusion of the ICONET project to be compliant with the release 3 of DCSA standards regarding the security requirements (planned to release in Q2 of 2021, see Figure 14).

3.4.4 The Cloud IoT Platform – Interoperability and security

To implement an interoperable Cloud IoT platform to cooperate with the PI environment as well as with the third parties' stakeholders involved in the shipments. For this reason, we have preferred to consider the main approaches and protocols available in the Internet world.

In this scenario, to implement the technical and syntactic interoperability level we have considered HTTPS protocol and JSON representation respectively, while to reach the semantic one we have agreed with the other project partners (see Sec. 5.1). Future improvement, after the finalisation of the ICONET project, are foreseen: we aim to be in line with the guidelines provided by the forthcoming contribution by DCSA on its standardisation, especially regarding Release 2 of its IoT standards program, planned to release in Q4 of 2020.

Regarding the security, the IoT Cloud platform is designed to implement secure and ad-hoc transaction with the users, considering HTTPS and "standard basic authentication" respectively, that will be further detailed in Sec. 6.7 - Security and ad-hoc access. Future investigations, to be done after the conclusion of the ICONET project, will consider the compliancy with the release 3 of DCSA standards regarding the security requirements (planned to release in Q2 of 2021).

3.4.5 The Cloud IoT Platform - data sharing

One of the main objectives of PI is to provide information to all the stakeholders involved in the supply chain, to support the improvement of the production processes connected to the goods dispatchment, but also to define the liability of certain events. The importance of the PI Tracking was extensively discussed in D2.6 being the first version of this deliverable. For these reasons, all the data collected by the IoT environment must be made available to all the users both in term of accessibility (exploiting standardised and interoperable interfaces, as described in the previous section), and of comprehensibility (agreeing on data models for data sharing). The latter imposes the transformation of the raw data represented in JSON in human comprehensible information requiring the following actions:

1. **Provide a users' profiling to generate ad-hoc access**. In this manner, the user will be capable to access a certain logistics transaction using private credentials. In this scenario, the privacy issues are maintained and the access to the data simplified.
2. **Generate events derived from the measurement gathered by the IoT devices**: e.g., bump detection, movement detection, threshold exceeding, etc.

3. Analyse the data retrieved by the IoT devices generating **high level statistics and metadata** (e.g., “how long the container stays at the customs?”). In this manner, raw data will be transformed into more comprehensible and aggregated information.
4. Realising an **ergonomic Graphical User Interface** (**GUI**), capable to visualise the route and the status of the container.

3.4.6 Generation of event, statistics, and meta-data

For the generation of events, statistics, and metadata, we will follow the following two approaches:

1. **Edge computing approach**, that allows the processing of certain raw data at the edge of the collection layer (e.g., in the IoT nodes) generating asynchronous event driven information. The information generated by the edge computing are called **events**. Of course, events can be computed at the Cloud side, but this type of information regards the punctual processing (only a set of samples related to the same time) of data gathered from the IoT devices.
2. **Cloud computing approach**, that allows the comprehension of long duration events, integrating along the time the raw data collected by the IoT devices. To implement such approach a data storage is required, and it will be realised at the Cloud/server level. In this scenario, **meta-data** defines the information that describes *the performance of a certain transaction* (e.g., time of arrival), while **statistics** define the information derived by aggregating meta-data and/or events (e.g., the average of time of arrival). Particularly, statistics are general parameters independent by the shipment, anonymised and *represent the performance of the PI environment*. These parameters are computed/updated (metadata/statistics) at the end of each shipment.

In the following table the events, statistics and meta-data are listed which we suggest implementing within the IoT environment. This list is consolidated in D2.8, where the complete set of event, statistics and meta-data will be defined and implemented, and their dispatchment methodology detailed.

Table 3 Events, statistics and meta-data

Name	Type	Approach	Description
Stop detection	Event	Edge	It detects whether the container is stopped.
Motion detection	Event	Edge	It detects whether the container is starting to move.
Bump detection	Event	Edge	It detects a bump.
Threshold exceeding	Event	Cloud	It compares whether a threshold is exceeded (e.g., regarding temperature data).
Start date&time	Event	Cloud	It represents the date and the time when the transaction is started.
Finish date&time	Event	Cloud	It represents the date and the time when the transaction is finished.
Travelled time	Meta-data	Cloud	Time to arrive from a point to another.
Travelled distance	Meta-data	Cloud	Travelled distance computed using a straight line distance metric
Time of stop	Meta-data	Cloud	Interval of time that a container stays stopped in a certain area (e.g., customs).
Avg. speed	Meta-data	Cloud	Average speed from a point to another.

Crossed countries	Meta-data	Cloud	Countries crossed during the shipment
Shipment "efficiency" – stop time	Meta-data	Cloud	The percentage of time, within the shipment, that a container stay stopped in a selected corridor.
Avg. duration	Statistics	Cloud	Average time to arrive from a certain point to a certain other.
Avg. Time of stop	Statistics	Cloud	Average time that a container stays stopped in a certain area (e.g., customs).
Avg. distance	Statistics	Cloud	Average distance between a point to another.
Corridor "efficiency" – Avg. stop time	Statistics	Cloud	The average of the shipment efficiency in a certain corridor
Container usage "efficiency" – used time	Statistics	Cloud	The percentage time that a container is booked in a PI orders with respect to a time period.

The computation and the sharing of these parameters will support all the PI users in the fact-based and data-oriented decision making process, simplifying the understanding how the goods dispatchments in implemented. Particularly:

1. **Events** can provide information regarding punctual facts happened during the goods dispatchment. The approach presented in ICONET allows the generation of **centralised generic events** computed on the stable IoT devices installed on the containers/truck trailers, as the one proposed in Table 3. However, it also enables the integration of **a scalable set of "special" pervasive functionalities**, exploiting interoperable "special" IoT devices thought for certain vertical functionalities and services (e.g., monitoring of expensive wines).
2. **Metadata provides an in depth characterisation of the considered shipment**, providing an ergonomic representation to analyse its performance.
3. **Statistics provides aggregated data regarding the supply chain**, providing a powerful instrument of analysis and comparation, to evaluate the performance of a shipment, as well as of the whole logistics networks.

3.5 Architecture realisation and main components

The proposed architecture depicted in Figure 2 is a general architecture that can be deployed all along the supply chain, thus enabling its complete monitoring and the realisation of the PI-digital-twin. In this, the standardised communication interfaces will be integrated on the goods (PI packets or PI pallets) and will communicate exploiting the available communication channels on the connected PI-means, PI warehouses, and PI container. Particularly, the smart PI container can be considered the cornerstone of the PI, allowing the interoperation (monitoring) with the connected goods in-movement without direct power supply (battery powered) and remote communication considering different possible technologies thought for different environments (e.g., the boat environment in the middle of the sea, the truck environment in the motorway…).

In the following, the realisation of the Smart PI Container is detailed. Moreover, also the definition of the concepts of Smart PI Pallet and Smart PI Packet is provided, improving the goods tracking and monitoring granularity, and enabling the instantiation of scalable set of special functionalities and services, tailored for the specific commodities sector.

3.5.1 The realisation of the Smart PI Container – The Smart PI Pallet and the Smart PI Packets

The Smart PI container is the cornerstone of the ICONET finding. In fact, it is the brick that allows the realisation of the IoT environment capable to answer to the three questions of

Table 2. A **Smart PI container** can be defined *as a connected PI container that can establish (inside and/or outside itself) local IoT networks for data collection, and it is capable to dispatch remotely the gathered information.* Such a solution can enable an improved scalability on the amount and types of IoT sensors capable to provide added value information.

In this scenario, the smart PI container will be capable, to monitor the goods encapsulate inside itself, and to be encapsulated in a Smart PI-mean (or another level in the hierarchy shown in Figure 3), as exemplified in Figure 15. It allows the data collection regarding:

1. The **general status of the container**, in term of position and time, as well as aggregated data as bump, T/H and light.
2. **A distributed set of devices capable to monitor in a pervasive manner inside and outside the container**. Added value functionalities can be added to monitor the presence or the punctual status of the goods, or the container itself status (door open/closed, sealed/not sealed). In the following the concepts of Smart PI Pallet and Smart PI packet will be defined.

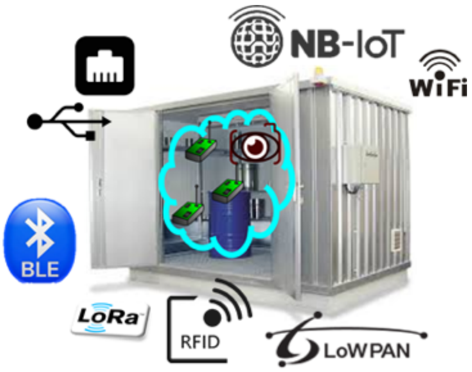


Figure 15 Smart PI-container

The smart PI-container must communicate remotely toward the IoT Cloud platform dispatching all the information collected. Following the DCSA guidelines the IoT device can exploit different paths toward the Cloud, selecting the most convenient between the available.

Figure 16 The Smart PI Pallet

The Smart PI Container allows the definition of **Smart PI Pallet**, see Figure 16. *The Smart PI pallet is a connected PI pallet equipped with sensors capable to implement special functionality and services measuring data from specific sensors, generating added value information regarding the encapsulated goods: for example, presence, bump, temperature. It can monitor, eventually,* **Smart PI Packets** *encapsulated on it and dispatch remotely the collected information participating of the proposed architecture.* Particularly, it can cooperate with higher level IoT devices or operate in stand-alone manner if equipped with mobile connectivity.

The concepts of Smart PI Pallet and Smart PI Packet introduce a set of functionalities capable to improve the performance of the supply chain and its related assets. In fact, in this manner several business requirements (summarised in Sec. 3.1) are realised, implementing:

1. **An improved granularity in the goods tracking and monitoring**. In this manner, the tracking and the monitoring services can be done at the level of the pallet or of the packet, supporting more punctual operations of routing and decision making. This service can, for example, support logistics operators that implement groupage policies (e.g., the transport of wine, where punctual temperature and bump is very important), or the improvement of last mile logistics services. Particularly, certain special edge functionalities can be implemented, only introducing a special IoT device though for that sector (no other configuration required).
2. **The possibility to implement improved assets monitoring and management service**. Having available the big picture of the availability of assets (e.g., container and pallet), the optimisation of their usage can improve their efficiency, thus reducing the costs.
3. **An enablement for the implementation of circular economy services**. This point is the direct consequence of the previous one. In fact, having the big picture regarding the position and the availability of the assets (in this case the attentions are focused on pallets, kegs, baskets, …) efficient

and low cost asset-as-a-service solutions can be implemented. In this scenario, the re-use of assets is encouraged, thus reducing the environmental impact.

In the ICONET project, the enabler of both the Smart PI Container and the Smart PI Pallet is the smart routers (for further technical details, see Sec. 7). This device, implemented in the ICONET project, allows the implementation of all the features previously described:

1. It can be positioned on the top of the container and behaves as tracker (battery powered, GPS and on board sensors, 2G/5G connectivity).
2. It can establish internal connectivity to gather information from Smart PI Pallets as well as from added value IoT devices (e.g., gas sensors, open/close door, …), considering 2 different short range IoT protocols.
3. It can establish external connectivity, to gather information from external added values IoT devices (e.g., smart seals, …), as well as cooperate with gateway compliant with DCSA (instead of using 2G/5G, thus reducing the power consumption and the costs) and with ISO CALM stack standards (see Sec. 3.3).

On the other side, for the implementation of the Smart PI Pallet we have considered commercial devices BLE beacons equipped with added values sensors (e.g., temperature and humidity, bumps, reed sensors, …). Such type of devices is selected since they have a cost comparable with the cost of the monitored assets.

The introduction of such device on the standard PI pallet **allows the implementation the encapsulation tracking**, enabling an **improved granularity in the monitoring task**. In this scenario a unique relation between the Smart PI pallet and the higher level gateway (installed, for example, on the Smart PI container or on a truck trailer) is established, allowing to have un-ambiguous goods-wise tracking. In the last mile logistics domain, it can support to monitor the operativity, e.g. understanding where the pallet is uploaded and downloaded. In the warehouse domain, it supports automatic inventory functionality and on the storage quality monitoring.

Moreover, **it allows to implement punctual, personalised, dedicated, and specialised monitoring functionalities** related with the goods encapsulated on it. For example, certain (expensive) wines need to monitor the temperature trends (they cannot to stay over a certain temperature for a certain period, to maintain their organoleptic characteristics): in this scenario special IoT devices can be installed in “special” Smart PI Pallets can be thought for these special functions and interoperate with the IoT environments (exploiting standardised approaches), without the need of reconfiguring the other IoT devices.

Asset management functionalities can be implemented having the position of “empty” pallets. These functionalities can improve the circular economy, reducing the environmental impact and allowing the implementation of “as-a-service” business models. This scenario fits perfectly with the realisation of specialised pallets (as previously mentioned), where the cost of the IoT device amortised, thus reducing the service costs and improving its performance.

Finally, other IoT devices can be added up to the Smart PI Container to implement important monitoring functionalities, as for example:

1. **Predictive maintenance** in reefer engines, to understand the health status of the motors to activate the refrigeration units.
2. **Gas monitoring.** Having information regarding the compounds’ concentration can support decision making (e.g., measuring the ethylene to understand the maturity level of the fruit) and improve the security (e.g., if inside the container a dangerous compound is generated, the operators can be preventively warned). Moreover, pollution smart sensors can provide important information to support a greener logistics.
3. **Consistency of the shipment.** A smart seal can notify when and where it is open or closed, providing added value information regarding the efficiency of the shipment, as well as detecting tampering actions.

3.5.2 Smart PI means and Smart PI Infrastructure

DCSA standardisation framework (see Sec. 3.3.1) foresees the presence of gateways on the PI means or on the PI infrastructure (i.e., PI hubs and PI warehouses) capable to collect data and information regarding the logistics equipment (i.e., the Smart PI Containers or the Smart PI pallets) as well as added value measurements from IoT sensors (e.g., pollution sensors, tyre pressure sensors in trucks). Particularly, DCSA Standard Release 1 (Digital Container Shipping Association (DCSA), 2020) suggest which IoT protocols must be used to be compliant with it. Figure 17 shows an example on how a train becomes a Smart PI mean considering the IoT technologies.

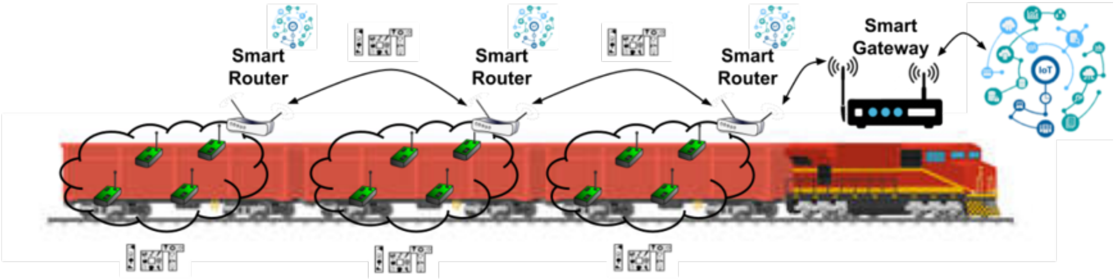


Figure 17 Train as a Smart PI-mean

To realise such solution, within the ICONET project we have implemented the Smart Gateway described in Sec. 8.

3.5.3 An IoT enabled PI environment

All the components described in the previous section allow the realisation of the supply chain complete visibility providing information regarding the goods at different granularity toward the PI users and the involved stakeholders. In fact, the concepts of Smart PI container, Smart PI Pallets, Smart PI Packet, as well as PI mean and PI infrastructure provide an ubiquitous connected environment capable to track and monitor the goods flows along the supply chain, from the sender toward the receiver, as depicted in Figure 1. The collected information will be dispatched exploiting the Cloud IoT platform (see Sec. 6) toward the higher OLI stack level, allowing proactive multimodal routing and fact-based and data-oriented decision making.

4. Toward an innovative Supply Chain complete visibility

In this section, we aim at wrapping up the innovation derived by the innovative approach used to realise the pervasive PI IoT environment. All these innovations come from the realisation of the visionary architecture presented in Sec. 3 (detailed in D1.6) capable to seamlessly interoperate with PI services, as well as with other domains (e.g., smart cities, ITS, smart factory/industry 4.0), providing added value information to support proactive and fact based actions. Having real time such variety of information, a real-time decision-making approach can be implemented toward the service optimisation, cost reduction and pollution reduction.

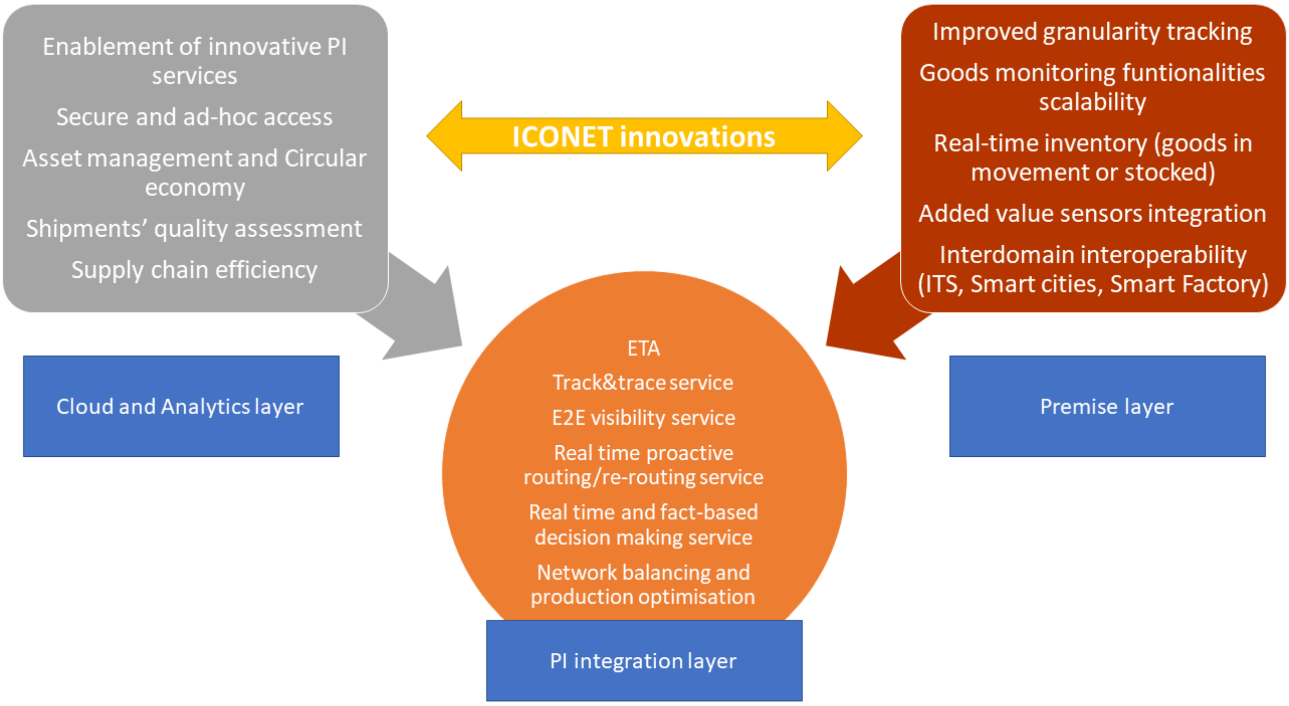

Enablement of innovative PI services
Secure and ad-hoc access
Asset management and Circular economy
Shipments' quality assessment
Supply chain efficiency
ICONET innovations
Improved granularity tracking
Goods monitoring funtionalities scalability
Real-time inventory (goods in movement or stocked)
Added value sensors integration
Interdomain interoperability (ITS, Smart cities, Smart Factory)
Cloud and Analytics layer
ETA
Track&trace service
E2E visibility service
Real time proactive routing/re-routing service
Real time and fact-based decision making service
Network balancing and production optimisation
Premise layer
PI integration layer

Figure 18 Innovation at different layers

In Figure 18, the direct innovations generated by the proposed approach are classified in 3 categories related with the different level of abstraction of the IoT environment:

1. The "**Premise Layer**" refers to the innovative services directly derived from the realisation of smart sensors deployed on the field.
2. The "**Cloud and Analytics Layer**", provide aggregated information capable to suggest the logistics network performance, thus supporting decision making and resources optimisation toward the cost reduction and the customers' satisfaction improvement.
3. The "**PI Integration Layer**" describes the functionalities and services derived from the realisation of an IoT-enabled PI environment. in fact, IoT is seen as enabling technology, providing raw and processed information supporting proactive actions/reactions, in the direction of resources' optimisation and cost reduction.

4.1 The Premise Layer

The premise layer examines innovation as relates to the potential value-added data as retrieved from the field. The proposed IoT architecture enables improved scalability, both in terms of quantities (number) and in terms of functionalities. In this context, Table 4 outlines the relevant innovations:

Table 4 The premise layer innovations

Inn. #	Title	Description	Ref. sections
INN_01	**Improved granularity in the goods tracking and monitoring**	This innovation allows the definition of Smart PI Container, Smart PI Pallet, and Smart PI Packet, installing IoT devices on top of these. In this scenario,	Sec. 3.5

		an improved tracking and monitoring of the goods is enabled.	
INN_02	**Real time inventory of the whole supply chain**	The real time inventory of the whole supply chain is enabled by both the Smart PI Pallet and Smart PI Packet realisation and the pervasiveness of the monitoring service, coming from the results related with INN_01.	Sec. 3.5
INN_03	**Scalability in terms of functionality**	The results related with INN_01 enable the realisation and the deployment of specialised devices to monitor certain types of goods (e.g., perishable goods, wines).	Sec. 3.5
INN_04	**Asset tracking and monitoring**	The device deployed on assets allows to track and monitor themselves, supporting **asset management and optimisation**, and **circular economy**.	Sec. 3.5
INN_05	**Value-Adding monitoring sensors to be installed on the whole supply chain**	Predictive maintenance for reefer motors, gas monitoring (e.g., ethylene sensors for monitoring fruit, harmful compounds monitoring), pollution monitoring, smart container seal can be seen as exhaustive set of examples capable to support the supply chain improvement toward its optimisation and improvement	Sec. 3.5
INN_06	**Interoperability with other domains**	Easy interoperability with other domains allows seamless exchange of value-adding information facilitating service optimisation and improvement.	Sec. 3.3 and 3.4

4.2 The Cloud and Analytics layer

The development of **the Cloud IoT platform enables efficient storage, processing and access to the data (**and meta-data), provisioned the IoT infrastructure. It also allows the accelerated implementation of a Track & Trace Service, and enables the innovations shown below:

Table 5 The Cloud and Analytics layer innovations

Inn. #	**Title**	**Description**	**Ref. sections**
INN_07	**Ad-hoc Secure access to data and information**	All authorized PI users involved in a given transaction will be aware in real-time of all the events related to the shipment, supporting fact-based and data-oriented decision making.	Sec. 3.5
INN_08	**Supply Chain Assessment**	Extracting, aggregating and anonymising IoT data and seamless storage and computation over the Cloud IoT platform, introduces better understanding of the Supply Chain behaviour providing deeper performance insights in terms of time, quality and efficiency.	Sec. 3.5
INN_09	**Circular Economy Business Models**	Easy to install and integrate IoT devices, engaged for asset monitoring (see INN_04) enable low-cost asset management consequently promoting **circular economy business models**.	Sec. 3.4 and 3.5

4.3 The PI Interaction Layer

The development of **the Cloud IoT platform enables the integration and the realisation of the PI innovative functionalities and services**. This layer enables the implementation of proactive and effective actions within the PI, integrating different information coming from several services (especially the Cloud IoT services) and influencing the physical logistics world. Table 6 shows the innovative PI services enabled by the integration with the IoT services.

Table 6 Innovative PI service enabled by IoT

Inn. #	**Title**	**Description**
INN_10	**Proactive decision-making functionalities**	IoT sensors track and monitor the goods providing real-time information in regards with their current status facilitating proactive decision making, either through direct human intervention or automatised by the PI itself to effectively optimise the transactions[3].
INN_11	**Real-time, adaptive, proactive routing/re-routing**	Real-time notifications regarding the position of the goods, the status of both the goods (i.e., quality rerouting[4]) and the network (e.g., forming queues) are provided with reduced latency, allowing to the realisation of proactive routing of the goods, capable to adapt to the configuration changes happened during the shipment.
INN_12	**Supply Chain organisation**	Having available a pervasive and interoperable world of IoT devices, the PI can be seen as an omniscient entity capable to organise all the supply chain from the producer to the costumer, maintain balanced and optimised transactions.

4.4 Innovations Application

In Table 9 the innovative contributions enabled and realised by the instantiation of the described innovative IoT services. Particularly, the last column of Table 9 aims at mapping the enabling innovations derived from the IoT services.

Finally, these contributions derive from the direct integration of the PI Shipping Service and PI Routing Service, capable to retrieve and infer on the collected/computed data and information, thus implementing proactive and adaptive actions to optimise the shipment itself.

Table 7 Innovative contributions of the proposed IoT environment within the ICONET project

LL	**Description**	Innovation
LL1	**Pollution reduction.** The monitoring of the pollution allows to understand critical area, supporting decision making. Its integration with the PI shipment service has the functions, in one hand, to understand the impact generated by PI, in the other to support the optimisation of the logistics service toward a greener logistics[5].	**INN_05, INN_08, INN_10, INN_12**
LL2	**Intermodal Tracking** enabling end-to-end visibility through the entire corridors interfacing with third parties back-end system. The ad-hoc access to the collected/computed information enable a secure switching of visibility to the logistic actor involved currently in the shipment.	**INN_01, INN_02, INN_03, INN_04, INN_07, INN_09**
LL2	**Events monitoring and reaction**: An IoT Sensor's accelerometer installed on a container carrying fragile goods detects a vibration level exceeding pre-set threshold and share with the PI. This automatically consumed to severe an SLA, ordering the redirection of the shipment to a warehouse instead of the intended delivery point.	**INN_01, INN_03, INN_07, INN_08, INN_10, INN_11, INN_12**

[3] A motivating example regards the logistics of fruits: introducing ethylene concentration sensors in a fruit container and evaluating its trend, the understanding the level of maturity of the fruits is enabled. This information can support action as blocking the shipment (e.g., rotten) or re-routing it (e.g., too mature to reach the real destination).

[4] Considering the fruits logistics, if the fruits are too mature the reach the final far destination, the container can be re-routed to other closer markets, attenuating the risks/costs, and optimising the transactions

[5] This task was disrupted by the COVID-19 pandemic, blocking its development and its integration with the LL1 environment.

LL2	**Dynamic Rerouting**: The Networking Service monitoring the Corridor detects a disruption of the network over the Alps with long waiting times due to a road accident. In parallel a Smart Container on transit carrying products sensitive to prolonged exposure to low temperatures, transmits rapid decline of the temperature within the container. The PI Routing Service utilizing those data, recommends to the truck to go out of the motorway, reach a PI hub and pass the goods on a train that is leaving in few hours later.	**INN_01, INN_03, INN_06, INN_07, INN_08, INN_10, INN_11, INN_12**

5. How to set-up the IoT-enabled PI environment

As describe in D2.7, in the ICONET project, PI is seen as a distributed and interoperable network of PI Nodes capable to organise autonomously the logistics network toward its optimisation. In this scenario, a PI node is depicted in Figure 19 and provide the following services:

- **Logistics Web Service**: This service exports the shipping instruction to the encapsulation service in order to bundle the shipping items into a PI-units and to enable an efficient and reliable planning of the shipment.
- **Shipping Service**: provides the functional and procedural means for enabling the efficient and reliable shipping of PI-containers. It sets, manages and closes the shipment between the shipper and each recipient. It defines the type of service to be delivered (normal, express, etc.) and insures the management of receipt acknowledgements. It establishes and rules the procedures and protocols for monitoring, verifying, adjourning, terminating and diversion of shipments.
- **Encapsulation Service**: provide the means for efficiently encapsulated products of a user in uniquely identified PI-containers before accessing the PI-networks. It allows linking product supply, realization, distribution and mobility taken at the Logistics Web Service.
- **Network Service**: provides the functional and procedural means for ensuring that PI-containers can be routed within a PI-network and across PI-networks while maintaining the quality of service given by the routing service.
- **Routing Service**: provides functional and procedural means for getting a set of PI-containers from its source to its destination in an efficient and reliable manner. It enables and controls the efficient and reliable inter-node transport and handling services to other services.

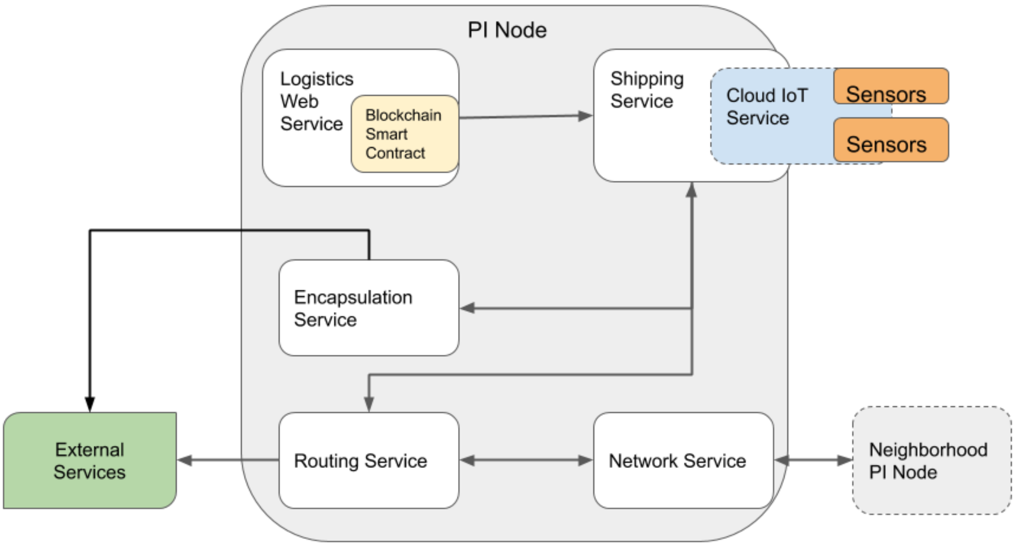


Figure 19 PI Node Block Diagram

Figure 20 shows a workflow diagram capable to highlight the interaction between the different services within the PI Node.

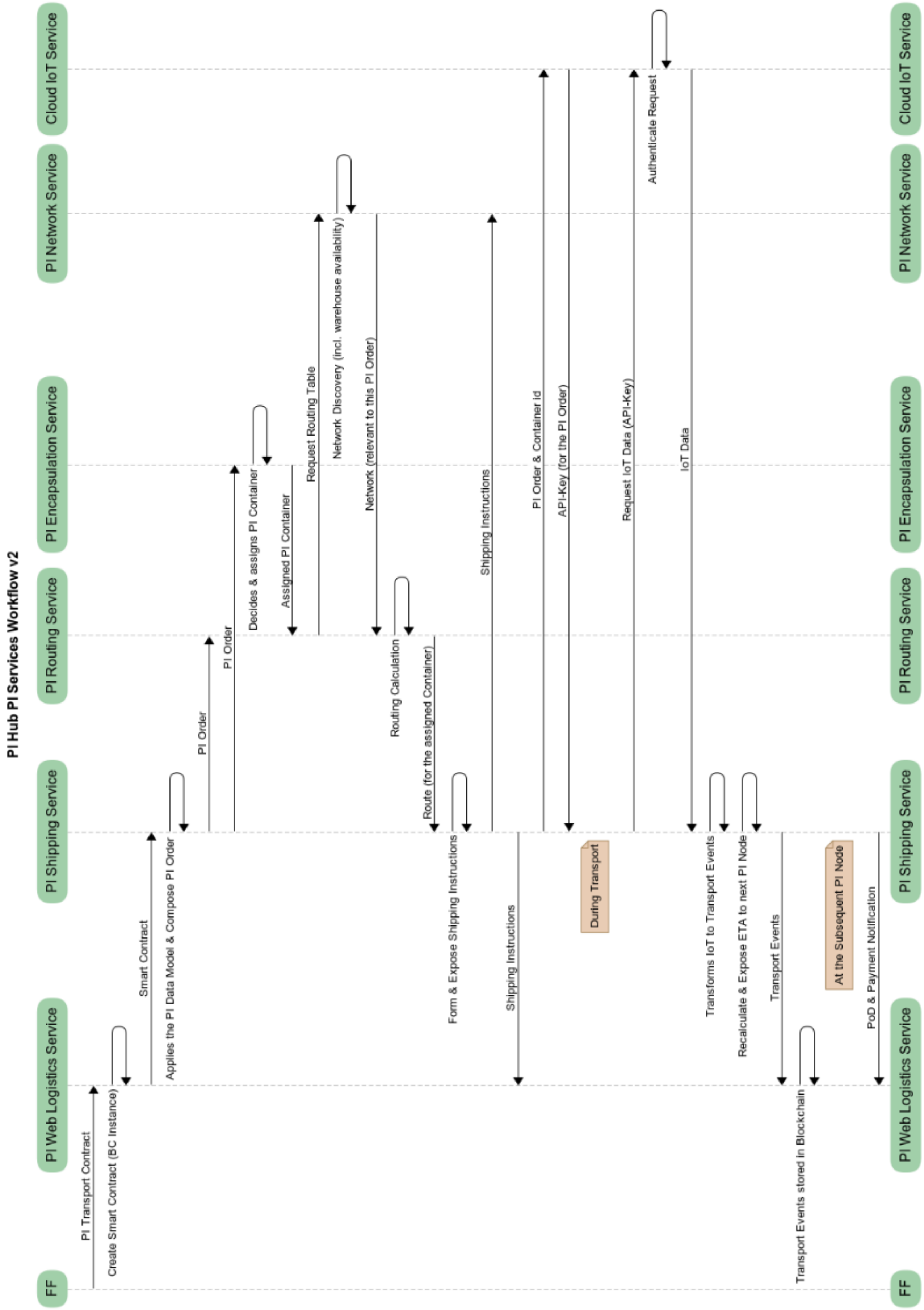

PI Hub PI Services Workflow v2
FF
PI Web Logistics Service
PI Shipping Service
PI Routing Service
PI Encapsulation Service
PI Network Service
Cloud IoT Service
PI Transport Contract
Create Smart Contract (BC Instance)
Smart Contract
Applies the PI Data Model & Compose PI Order
PI Order
PI Order
Decides & assigns PI Container
Assigned PI Container
Request Routing Table
Network Discovery (incl. warehouse availability)
Network (relevant to this PI Order)
Routing Calculation
Route (for the assigned Container)
Form & Expose Shipping Instructions
Shipping Instructions
Shipping Instructions
PI Order & Container id
API-Key (for the PI Order)
During Transport
Request IoT Data (API-Key)
Authenticate Request
IoT Data
Transforms IoT to Transport Events
Recalculate & Expose ETA to next PI Node
Transport Events
Transport Events stored in Blockchain
At the Subsequent PI Node
PoD & Payment Notification
FF
PI Web Logistics Service
PI Shipping Service
PI Routing Service
PI Encapsulation Service
PI Network Service
Cloud IoT Service

Figure 20 PI Services Workflow

In this scenario, the Encapsulation Service and the Shipping Service create the PI Order related to the shipment that contains several logistics information (e.g., the container id), as well as all the requirements to configure the data acquisition (e.g., set some thresholds). For this reason, the PI order is processed by the Cloud IoT Service to configure and to initialise the data acquisition service.

As introduced in D2.7, the interaction between the Cloud IoT service and the PI node is mapped in the block diagram shown in Figure 21 (regarding the initialisation of the Cloud IoT Service) and Figure 22 (the sharing of the data collected during the shipment).

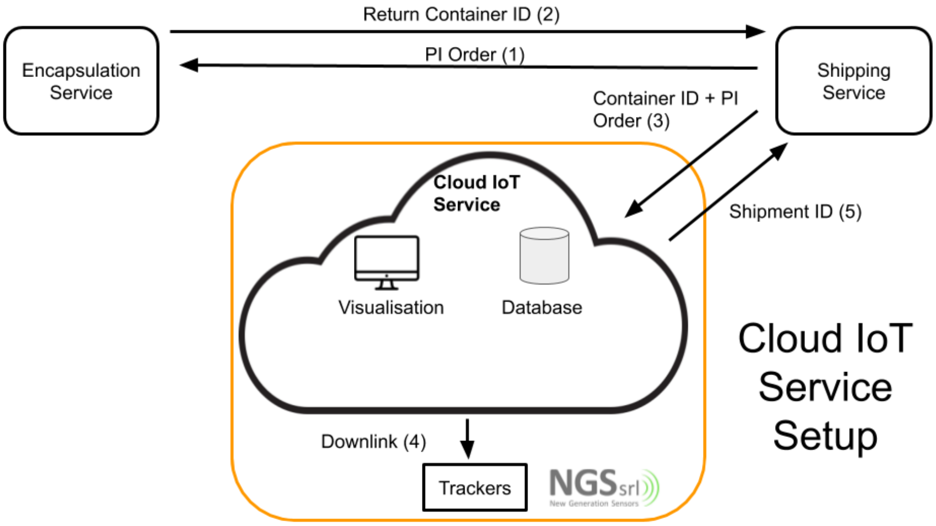


Figure 21 Cloud IoT Service Initialisation

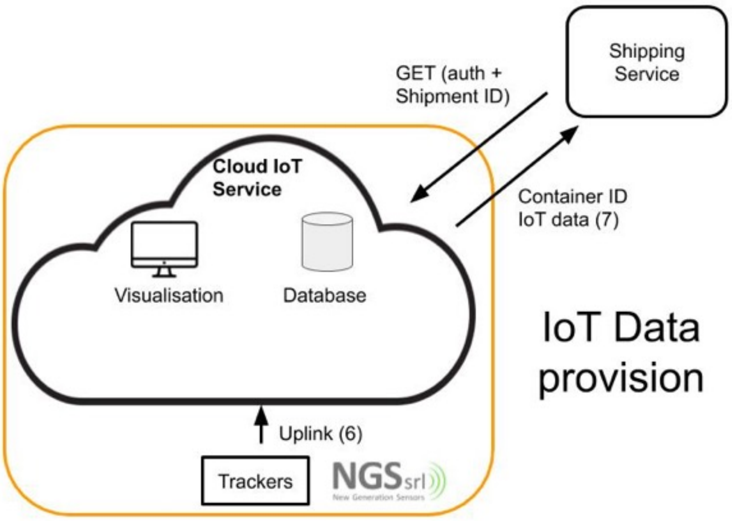


Figure 22 IoT data sharing

5.1 Interactions and data-models

In this section the interaction of the platform with the users, the shipping services, as well as with the IoT devices and the related data-models are described in their final release.

5.1.1 Cloud IoT Service Initialisation service and related data models

According to Figure 21 the following data models are required during the setup phase to initialise the Cloud IoT Service:

- Cloud IoT Service configuration data model included in PI Order from Shipping Service.

- The configuration of the IoT devices data model (transaction 4 in Figure 10), that configure the nodes for a new shipment.
- The returned data model from the Cloud IoT Service to the Shipping Service (transaction 5 in Figure 10). It contains the shipmentID that uniquely represents the considered logistics transaction.

In Figure 23 and Figure 24 the sub-data model of the PI-order regarding the Cloud IoT Service is depicted (transaction 3 in Figure 21). Particularly, Figure 24 depicts the complete data-model in form of JSON capable to configure the IoT Cloud Platform for enabling/disabling the sensors data collection and the thresholds to enable the generation of events.

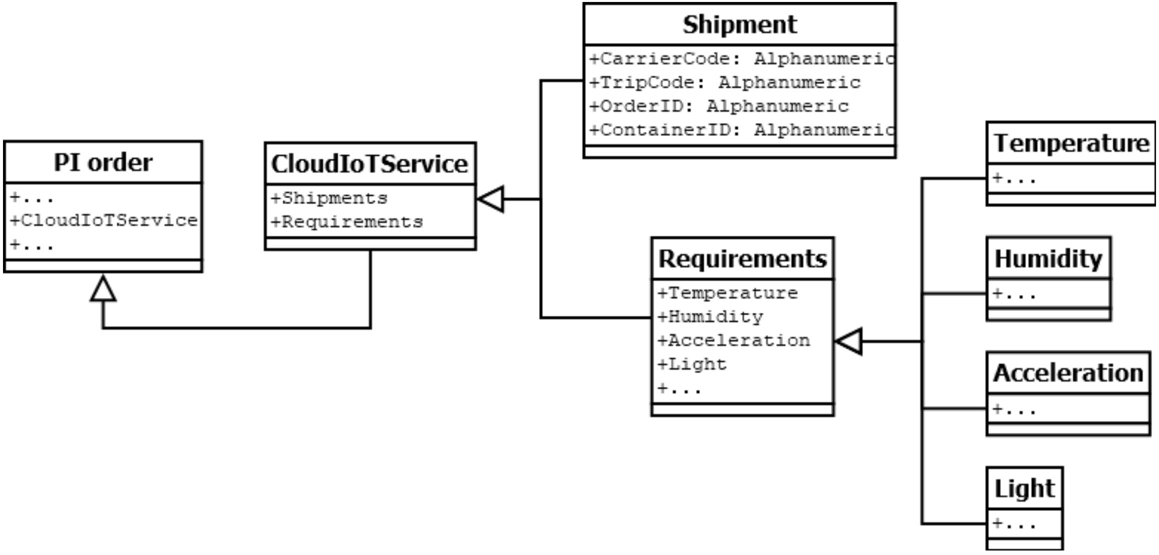
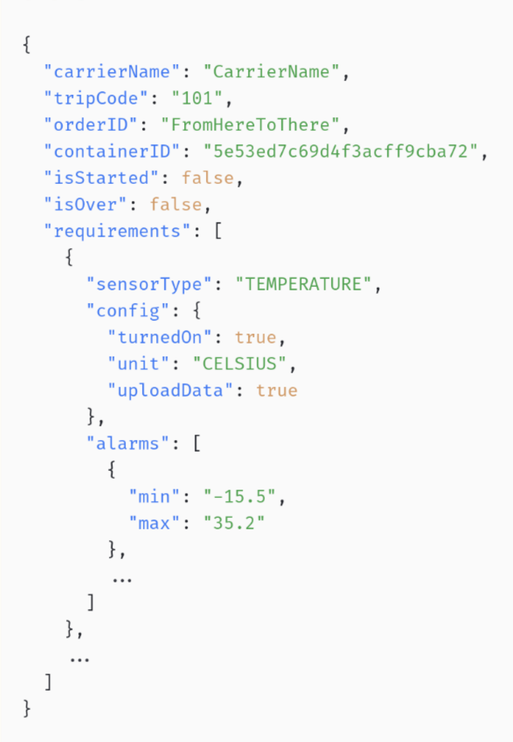


Figure 23 Cloud IoT Service Data Model Inside a PI Order (diagram)



```
{
  "carrierName": "CarrierName",
  "tripCode": "101",
  "orderID": "FromHereToThere",
  "containerID": "5e53ed7c69d4f3acff9cba72",
  "isStarted": false,
  "isOver": false,
  "requirements": [
    {
      "sensorType": "TEMPERATURE",
      "config": {
        "turnedOn": true,
        "unit": "CELSIUS",
        "uploadData": true
      },
      "alarms": [
        {
          "min": "-15.5",
          "max": "35.2"
        },
        ...
      ]
    },
    ...
  ]
}
```

Figure 24 Cloud IoT Service Data Model Inside a PI Order

The proposed data model is basically divided into two parts.

- **Shipment section**: includes all information about the carrier and container ID. This field is mandatory.
- **Requirements section**: includes all optional needs about the shipment and the transported goods. Particularly, this part oversees configuring the sensors for generating events thus meta-data. This field is optional.

In Table 8, all the field of the considered data-model are briefly described.

Table 8 Description of Cloud IoT Service Data Model Inside a PI Order fields

Field	Description
“carrierName”	Mandatory field for specifying the carrier company
“tripCode”	Mandatory field specifying the trip code
“orderID”	Mandatory field indicating the order ID
“containerID”	Mandatory field specifying a unique container ID or container’s ISO Alpha Code
“isStarted”	Mandatory field notifying that the container is transferred
“isOver”	Mandatory field notifying that the container reaches its destination
“requirements”	Optional field specifying the requirements for a specific shipment
“sensorType”	Optional field indicating the type of measurement
“config”	Optional field for setting up a specific sensor
turnedOn	Optional field for switching on/off a specific sensor
“unit”	Optional field for setting up the measurement unit of a sensor
“uploadData”	Optional field for sending data
“alarms”	Optional field for setting up thresholds for a specific sensor
“min”	Optional field for setting up minimum threshold
“max”	Optional field for setting up maximum threshold

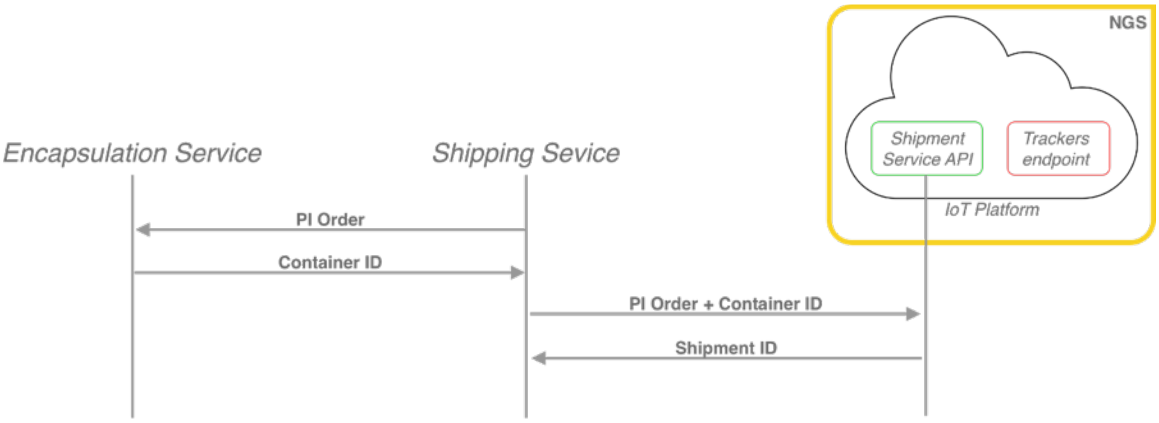


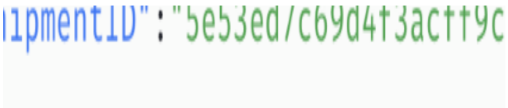
Figure 25 Transactions for configuring Cloud IoT Platform

As depicted in Figure 25, after receiving a PI order containing the data model specified above, the Cloud IoT Service parses and processes the data, and performs the following actions:

1. Setting-up of the Cloud IoT Service, configuring all the internal parameters. Particularly, the Cloud IoT Service considers the requirements from the PI order to set up properly the Cloud IoT Platform

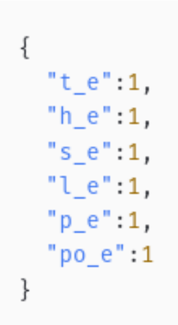
parameter for configuring notification and/or alarm whenever there is a problem occurring during transportation.

2. The Cloud IoT Server answer sharing a unique identifier, called "Shipment ID", represented in the data-model of Figure 26.
3. Optionally, it will also be in charge of configuring the corresponding tracker/smart router devices. The downlink transaction to configure the IoT tracker/smart router (transaction 4 in Figure 21), is defined in the data-model of Figure 27 (the description of the downlink data model fields is shown in Table 9), however its implementation will not be realised in ICONET project, since not essential for its objectives.



```
ipmentID":"5e53ed7c69d4f3acff9c
```

Figure 26 ShipmentID data-model



```
{
  "t_e":1,
  "h_e":1,
  "s_e":1,
  "l_e":1,
  "p_e":1,
  "po_e":1
}
```

Figure 27 Downlink Data Model

Table 9 Description of Downlink Data Model fields

Field	Description
"t_e"	Enable/Disable temperature measurement
"h_e"	Enable/Disable humidity measurement
"s_e"	Enable/Disable shock detection
"l_e"	Enable/Disable light measurement

5.1.2 Operativity interactions and related data-models

After the configuration phase described in the previous section, the Cloud IoT Service is configured to operate within the PI environment. Accordingly with Figure 21, the Cloud IoT Service will be in charge of:

1. Collecting the data from the IoT devices (transaction 6 in Figure 21), exploiting the data-model in Figure 28 (in Table 10 each field of the data-model is detailed). In fact, the IoT devices dispatch periodically toward the Cloud IoT Service the collected data and saved in the platform database. When a transaction is recognised (i.e., between the PI order reception and the end of transaction notification) this data is linked with the shipment, otherwise the data are stored to track the container (supporting the asset management).

2. Sharing the data gathered with all PI users involved in the shipment (transaction 7 in Figure 21), implementing the interaction depicted in Figure 28. The data-model shown in Figure 30 is used to share this information with the PI users exploiting a secure and ad-hoc transaction, as described in Sec. 5 (in Table 11 each field of the data-model is detailed).

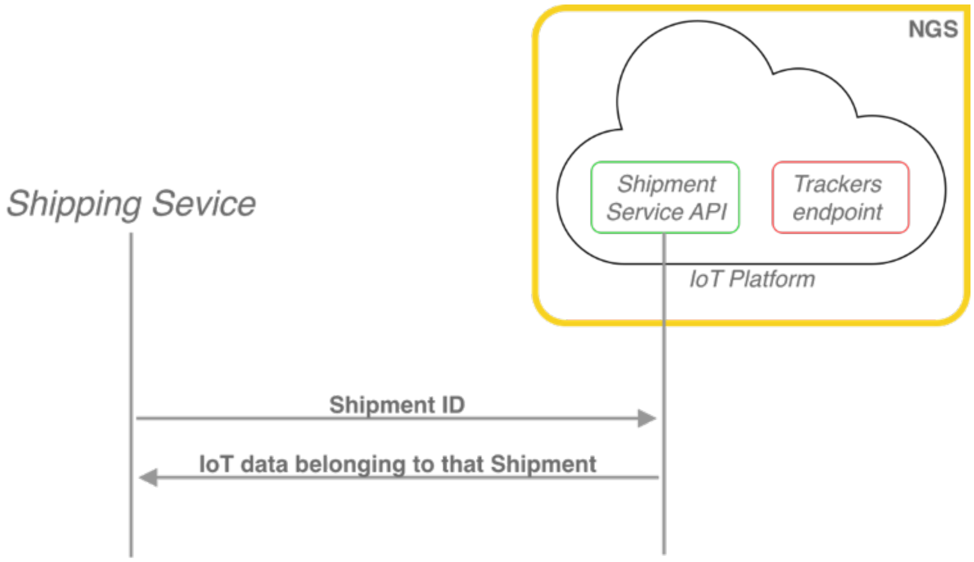
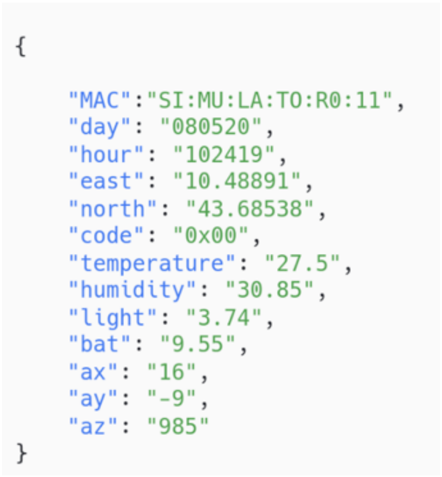


Figure 28 Transactions during IoT Provision



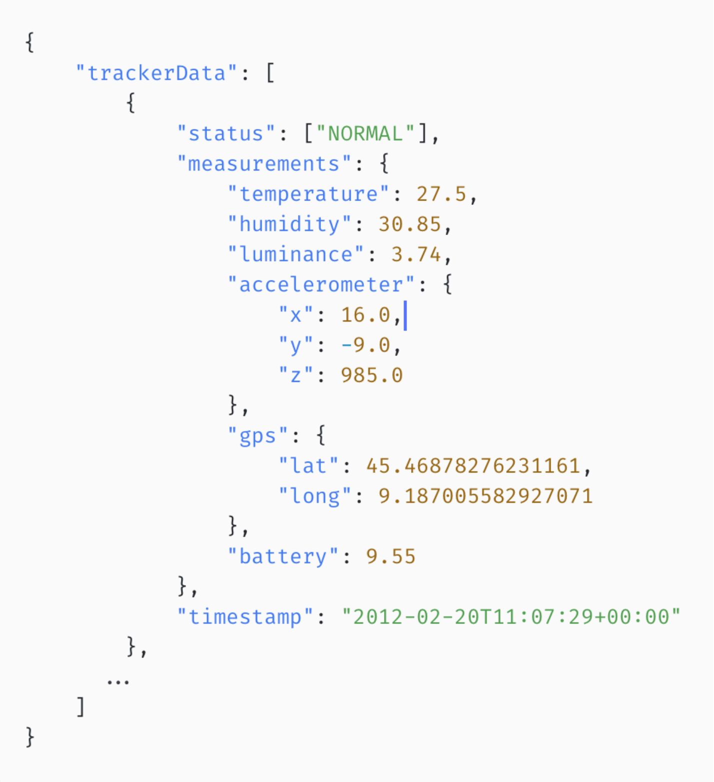
```
{

    "MAC":"SI:MU:LA:TO:R0:11",
    "day": "080520",
    "hour": "102419",
    "east": "10.48891",
    "north": "43.68538",
    "code": "0x00",
    "temperature": "27.5",
    "humidity": "30.85",
    "light": "3.74",
    "bat": "9.55",
    "ax": "16",
    "ay": "-9",
    "az": "985"
}
```

Figure 29 Uplink Data Model

Table 10 Description of Uplink Data Model fields

Field	Description
“MAC”	Unique MAC address of a tracker
“day”	date field specified at UTC+0
“hour”	hour field specified at UTC+0
“east”	Longitude
“north”	Latitude
“code”	Code indicating system status and events
“temperature”	Temperature in C degree
“humidity”	Humidity in Percentage
“light”	Luminance in lux
“bat”	Battery level in V
“ax”	Acceleration along x axis measured in mg
“ay”	Acceleration along y axis measured in mg

“az”	Acceleration along z axis measured in mg



```
{
    "trackerData": [
        {
            "status": ["NORMAL"],
            "measurements": {
                "temperature": 27.5,
                "humidity": 30.85,
                "luminance": 3.74,
                "accelerometer": {
                    "x": 16.0,
                    "y": -9.0,
                    "z": 985.0
                },
                "gps": {
                    "lat": 45.46878276231161,
                    "long": 9.187005582927071
                },
                "battery": 9.55
            },
            "timestamp": "2012-02-20T11:07:29+00:00"
        },
       ...
    ]
}
```

Figure 30 IoT Cloud Service Data Model

Table 11 Description of the IoT Cloud Service Data Model fields

Field	Description
“trackerData”	Mandatory field returning a list of trackers data
“status”	Mandatory field indicating trackers status
“measurements”	Mandatory field including measurement information from a tracker
“temperature”	Optional field for temperature measured in C degree
“humidity”	Optional field for humidity measured in percentage
“luminance”	Optional field for luminance measured in lux
“accelerometer”	Optional field for acceleration in three axes x, y, z
“gps”	Mandatory field including gps data
“lat”	Mandatory field for latitude in decimal degree
“long”	Mandatory field for longitude in decimal degree
“battery”	Optional field for battery measured in Volt
“timestamp”	ISO format timestamp at UTC+0

5.1.3 Events, metadata and statistics gathering and data models

A set of RESTful APIs is implemented to share events and metadata related to a certain shipment, exploiting the handshakes depicted in Figure 31 and Figure 32, and the data models in Figure 33 and Figure 34 respectively (their fields are described in Table 12 and Table 13).

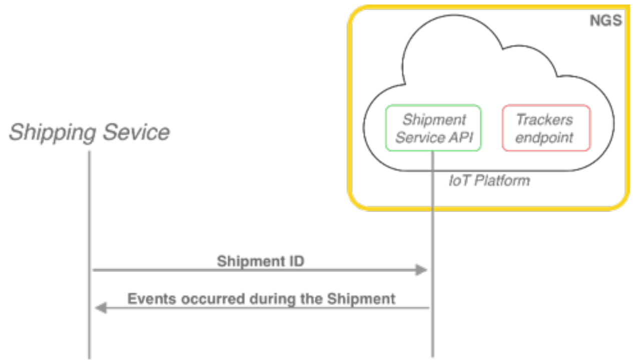


Figure 31 Handshake to retrieve the events (GET method)

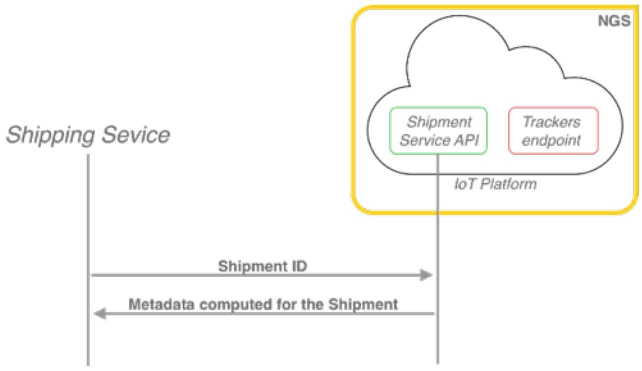
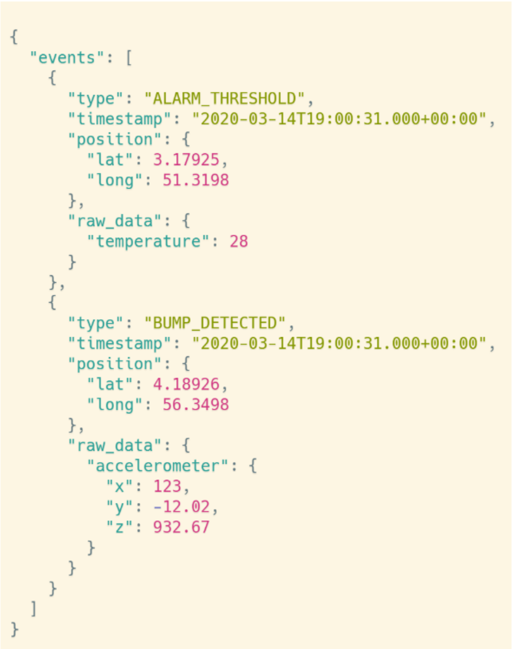
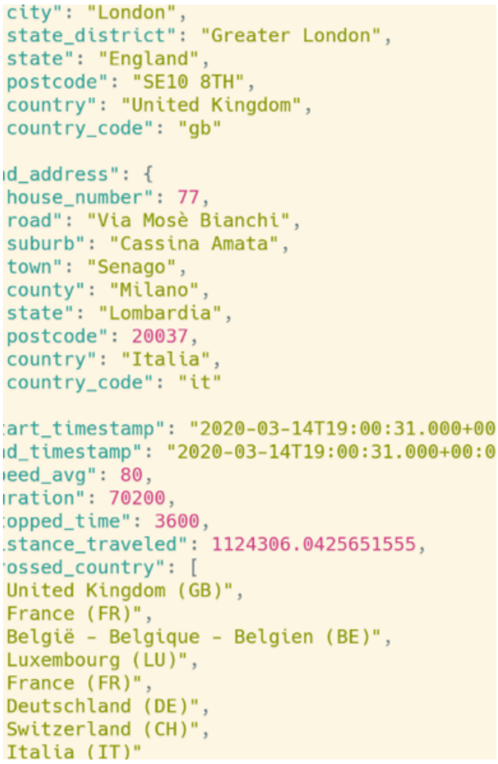


Figure 32 Handshake to retrieve the metadata (GET method)



```
{
  "events": [
    {
      "type": "ALARM_THRESHOLD",
      "timestamp": "2020-03-14T19:00:31.000+00:00",
      "position": {
        "lat": 3.17925,
        "long": 51.3198
      },
      "raw_data": {
        "temperature": 28
      }
    },
    {
      "type": "BUMP_DETECTED",
      "timestamp": "2020-03-14T19:00:31.000+00:00",
      "position": {
        "lat": 4.18926,
        "long": 56.3498
      },
      "raw_data": {
        "accelerometer": {
          "x": 123,
          "y": -12.02,
          "z": 932.67
        }
      }
    }
  ]
}
```

Figure 33 Data model to share events



```
city": "London",
state_district": "Greater London",
state": "England",
postcode": "SE10 8TH",
country": "United Kingdom",
country_code": "gb"

d_address": {
house_number": 77,
road": "Via Mosè Bianchi",
suburb": "Cassina Amata",
town": "Senago",
county": "Milano",
state": "Lombardia",
postcode": 20037,
country": "Italia",
country_code": "it"

art_timestamp": "2020-03-14T19:00:31.000+00
d_timestamp": "2020-03-14T19:00:31.000+00:0
eed_avg": 80,
ration": 70200,
opped_time": 3600,
stance_traveled": 1124306.0425651555,
ossed_country": [
United Kingdom (GB)",
France (FR)",
België - Belgique - Belgien (BE)",
Luxembourg (LU)",
France (FR)",
Deutschland (DE)",
Switzerland (CH)",
Italia (IT)"
```

Figure 34 Data model to share metadata

Table 12 Description of the Events Data Model fields

Field	Description
“events”	Field returning a list of events happened during the shipment
“type”	Type of event – e.g., “ALARM THRESHOLD” when a threshold is exceeded
“timestamp”	ISO format timestamp at UTC+0
“position”	GPS coordinates where the event happened, represented as latitude (“lat”) and longitude (“long”)
“raw_data”	Reference to the raw data collected followed by the raw data representation (e.g., “temperature”, “humidity”, “acceleration”

Table 13 Description of the Metadata Data Model fields

Field	Description
“start_address” **“end_address”**	Field returning the address where the shipment is started/ended
“city_district”, “city”, “state_district”, “state”, “postcode”, “country”, “country_code”, “house_number”, “road”, “suburb”, “town”	Addressed fields
“start_timestamp” **“end_timestamp”**	Field returning the timestamp (ISO format UTC+0) when the shipment is started/ended
“speed_avg”	Average speed in a shipment.
“duration”	Duration of the trip in second
“stopped_time”	It measures the time that a container stay stopped within a shipment in seconds
“distance_travelled”	Distance travelled in meters
“crossed_country”	List of crossed countries

On the other hand, statistics provide the general overview of the PI - for example suggesting the average duration of a trip along a corridor. For this reason, all the requests will be considered set of parameters to uniquely define conditions to retrieve the information, as depicted in Figure 35. As matter of example, in Figure 36 to retrieve the average distance between two points, the GPS coordinates of the start and the end points must be provided as parameter in the GET HTPPS request.

In Figure 37, the generic data model to share the statistics information. It is a very simple data model data and returns the required “<statistics>”, where “<parameter>” is related with the requested statistics (e.g., “distance”, “time”, “efficiency”, …)

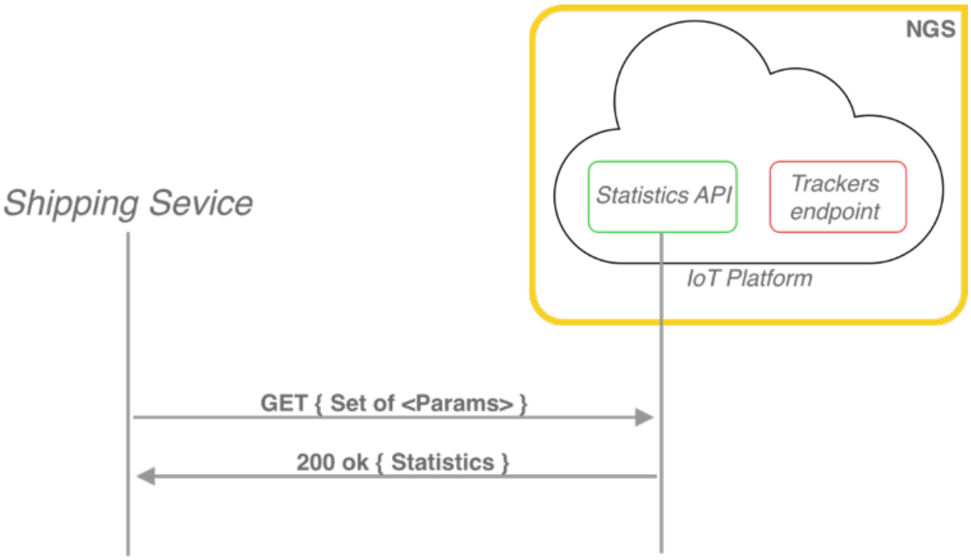


Figure 35 Generic handshake to retrieve statistics information

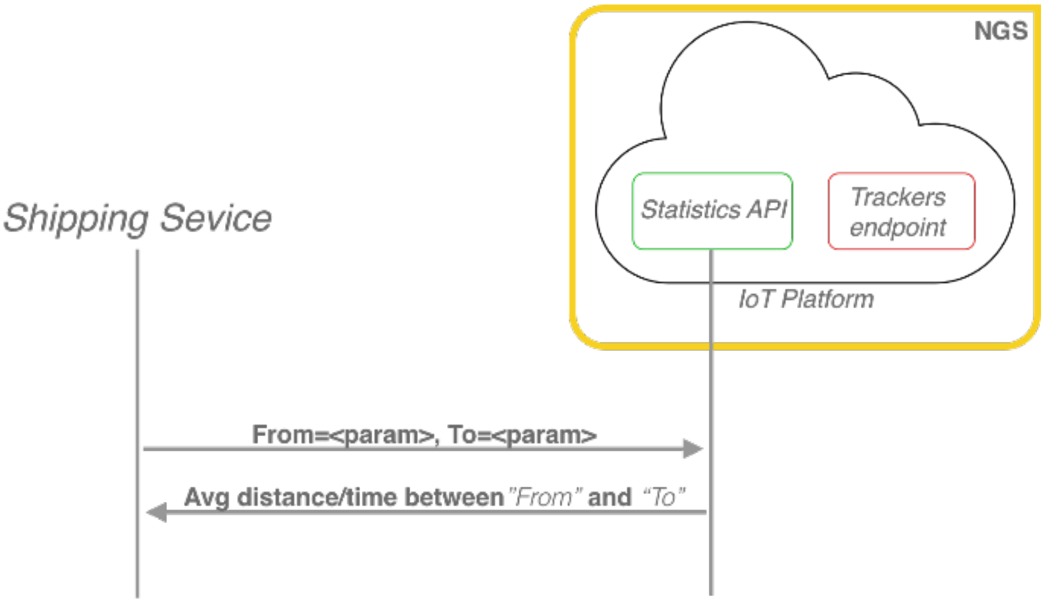
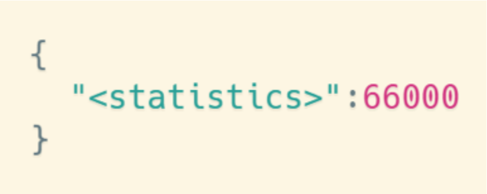


Figure 36 Handshake to retrieve the Avg. distance (GET method)



```
{
  "<statistics>":66000
}
```

Figure 37 Generic data model to share statistics

5.2 Data collection

5.2.1 How to authenticate a device

All the IoT devices must be authenticated (ad-hoc access control) to join and participate to the activities of the Cloud IoT platform. In fact, the data gathered by the IoT devices are the basis for implement the routing algorithms and the data-oriented and fact-oriented decision making policies, so authentication procedures must be implemented to guarantee the authenticity of the data. To realise such procedure, we use the "standard basic authentication" of http that implement the IETF standard RFC 7617 (IETF, 2015). In this scenario, each IoT device must provide an authorised username and password when it is making a request to access the Cloud IoT platform. Further technical details will be provided in Sec. 6.7.

5.2.2 Data collection from real devices

The IoT devices considered are installed in 4 containers, as well as installed in some private vehicles of NGS people. The device is sending every 10 minutes the information collected regarding the associated container, but it is configured to be triggered if a certain event happens. In this scenario, the information collected and shared with the Cloud IoT Platform can be classified as follow (see also Figure 38):

1. **Synchronous measurements, or Time Driven Acquisition**. These data are collected synchronously with the sampling period. Data collected with the Time Driven paradigm is position and time, and data gathered from sensors' ICs (i.e., T/H, acceleration, battery). This information is immediately dispatched toward the Cloud IoT Platform.
2. **Asynchronous measurements/events**. These measurements happen when certain events (i.e., bump, movement after a stopping period) trigger the device to collect data. In this scenario, the event data is collected, tagged with position and time information, and stored to be sent with the following Synchronous measurement.

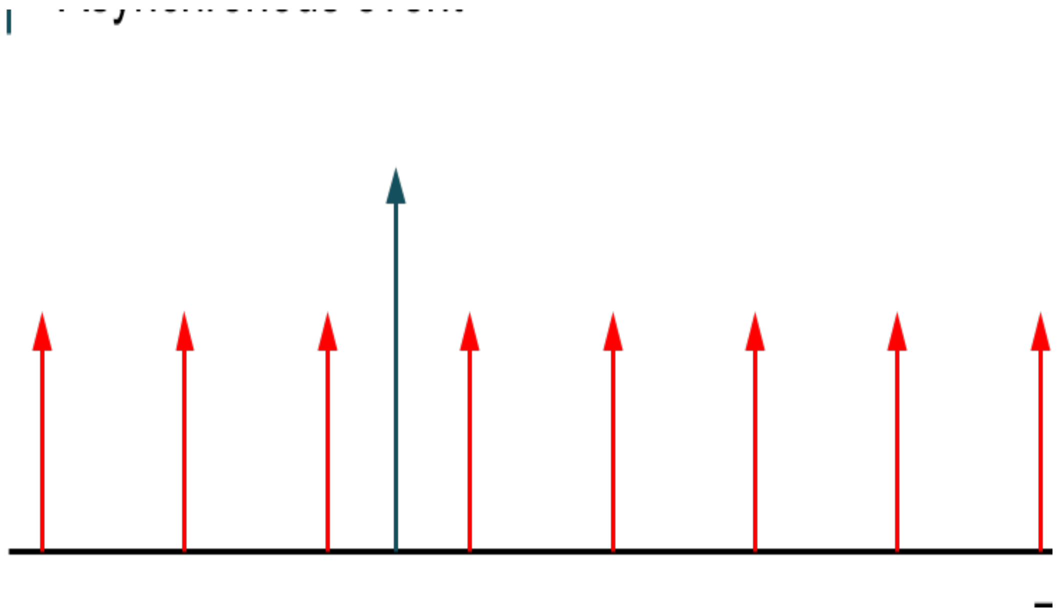


Figure 38 Time Driven Acquisition vs. Asynchronous events

The system can communicate remotely selecting between the enabled communications protocols, following one of the following approaches:

1. **Stand-alone approach**. The system will exploit the mobile connectivity enabled by the IoT devices to dispatch remotely the collected data as depicted in Figure 39. In this case, GPRS, NB-IoT and LTE Cat-M can be used to implement the communication transaction. In the ICONET project, only the former protocol is considered, since the limitation of NB-IoT and LTE Cat-M described in D2.6 and D2.7. This approach is in line with the DCSA protocol guidelines, described in Sec. 3.3.1.

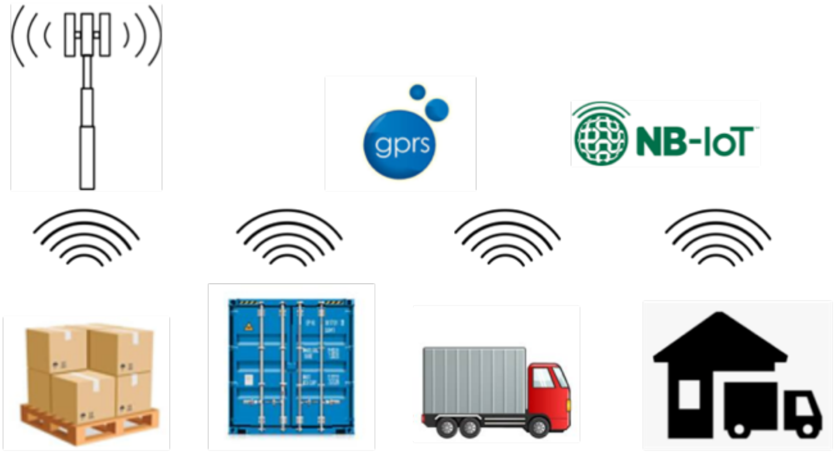


Figure 39 Stand-alone approach

2. **Cooperative approach**. The single IoT Device can select to communicate with a local gateway (in line with DCSA approach) exploiting low power consumption protocols, thus reducing the battery power consumption and the maintenance costs of the devices. In this scenario, all the IoT device will authenticate with an IoT gateway installed in PI-means or PI-Hub, that will be in charge of dispatching the data toward the Cloud IoT Platform, as depicted in Figure 40. The considered gateway is based on the hardware described in Sec. 7, connected toward the Cloud IoT platform using different protocols to connect with the Internet: i.e., Wi-Fi, Ethernet (implementing a direct transaction toward the Cloud IoT Platform) and mobile.

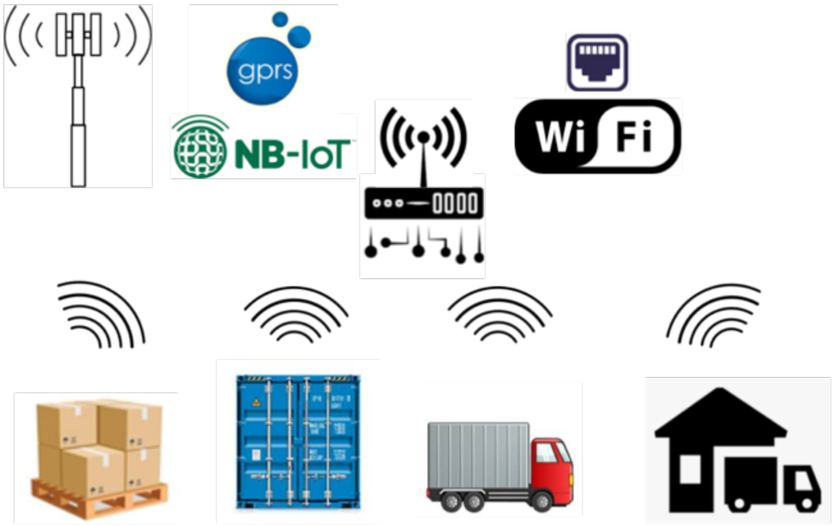


Figure 40 Cooperative approach

In the ICONET project, we present a real deployment of the IoT devices in LL2 and a lab demo to demonstrate the feasibility of the proposed approach, being completely in line with the open architecture proposed by DCSA.

5.2.3 Simulator

The simulation of the LL2 is focused on the evaluation of the performance of the IoT and the Cloud-based ICT Infrastructure for PoC Integration to foster the intermodal transportation. In this environment, due to the situation caused by the COVID-19, the objective was to develop a simulation model that would feed position data and movement events to the IoT service, as the sensors would do.

In the simulation model, nodes, containers and transports have been used as the basic elements. The nodes correspond to physical locations where trucks can make decisions and log additional events. Trucks are in charge of moving the containers with the sensors, following defined behaviour rules, defined by other ICONET services. Some example of the simulator output are shown in Figure 42, Figure 43 and Figure 44.

APIs are used to connect to services and exchange information. In the endpoints of the APIs are hosted web services, these services are made with the https protocol, through asynchronous requests. The information is transmitted through JSON by using Restful requests. Four types of messages are exchanged with the IoT tracking API (see also Figure 41):

1. **New shipment** (PUT): the shipment is registered within the ICONET IoT Cloud platform with the identifiers of the container, the carrier, the order, the origin and the destination. The service returns a unique identifier for the shipment.
2. **Status started notification** (POST): before starting to log the location and trip events, it must be initialized. The endpoint includes the shipment identifier.
3. **Tracker message** (POST): The simulated IoT sensor periodically records the location as well as the conditions of light, humidity, acceleration on three axes... When a sudden movement occurs, a message is also sent with the code corresponding to the direction of the movement.
4. **Status over notification** (POST): once the trip is over, the sending is closed at the same endpoint that uses the message 2.

This model is the first step to develop a Physical Internet Simulator in which the different ICONET services can be tested and integrated together.

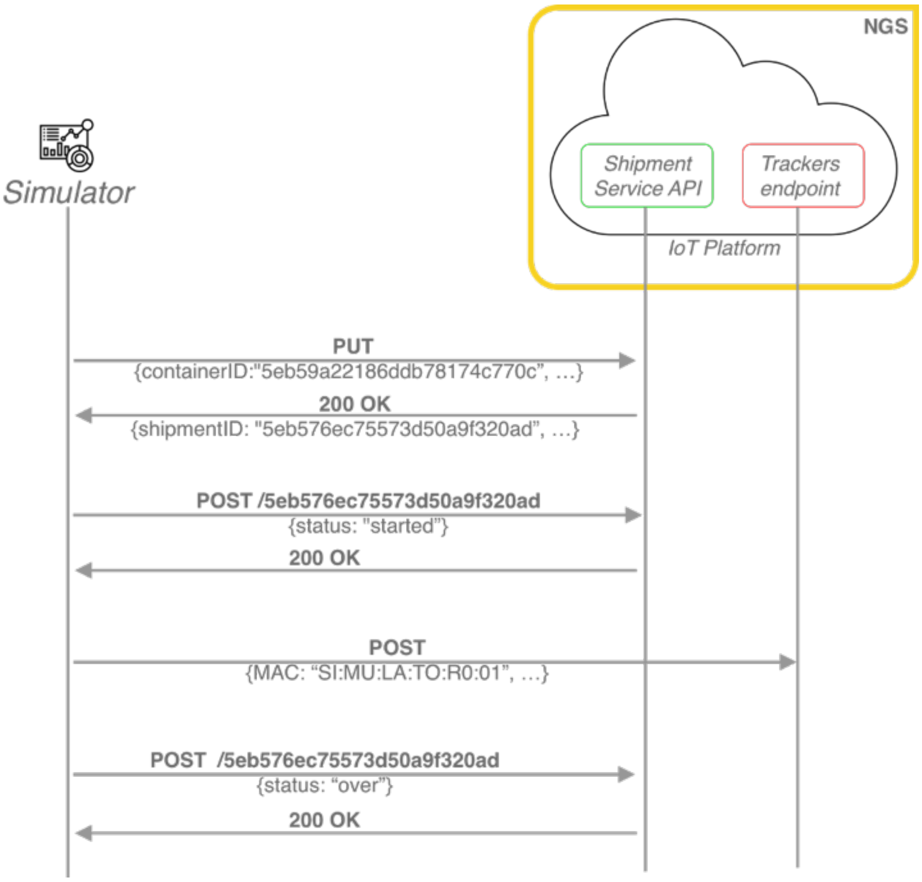

NGS
Simulator
Shipment Service API
Trackers endpoint
IoT Platform
PUT
{containerID:"5eb59a22186ddb78174c770c", …}
200 OK
{shipmentID: "5eb576ec75573d50a9f320ad", …}
POST /5eb576ec75573d50a9f320ad
{status: "started"}
200 OK
POST
{MAC: "SI:MU:LA:TO:R0:01", …}
POST /5eb576ec75573d50a9f320ad
{status: "over"}
200 OK

Figure 41 Handshake to set-up the simulated environment

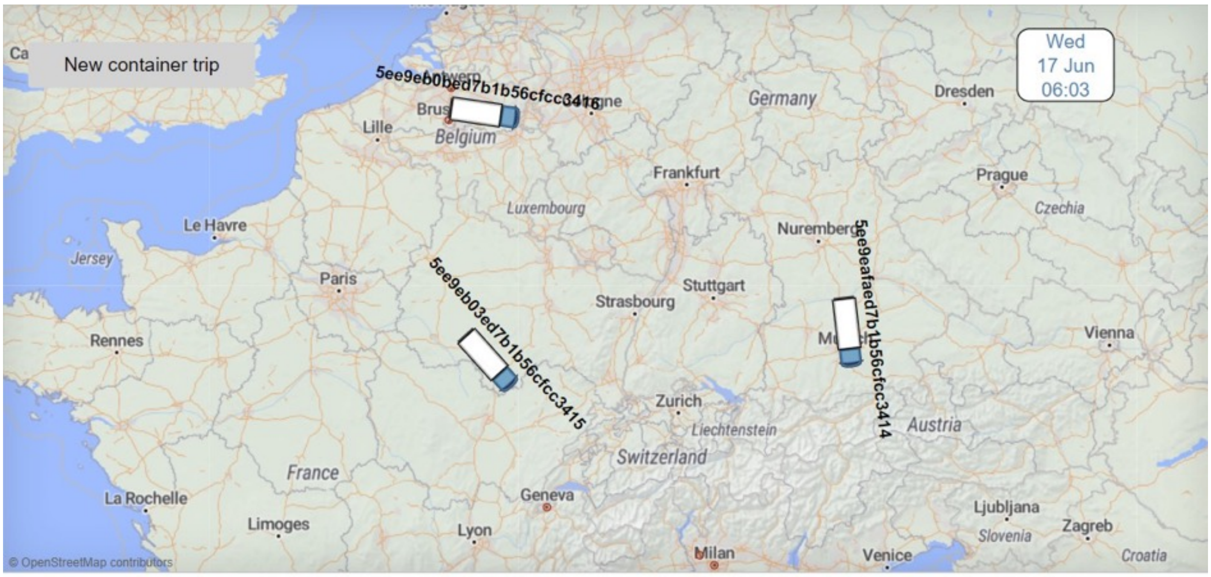


Figure 42 Simulator example

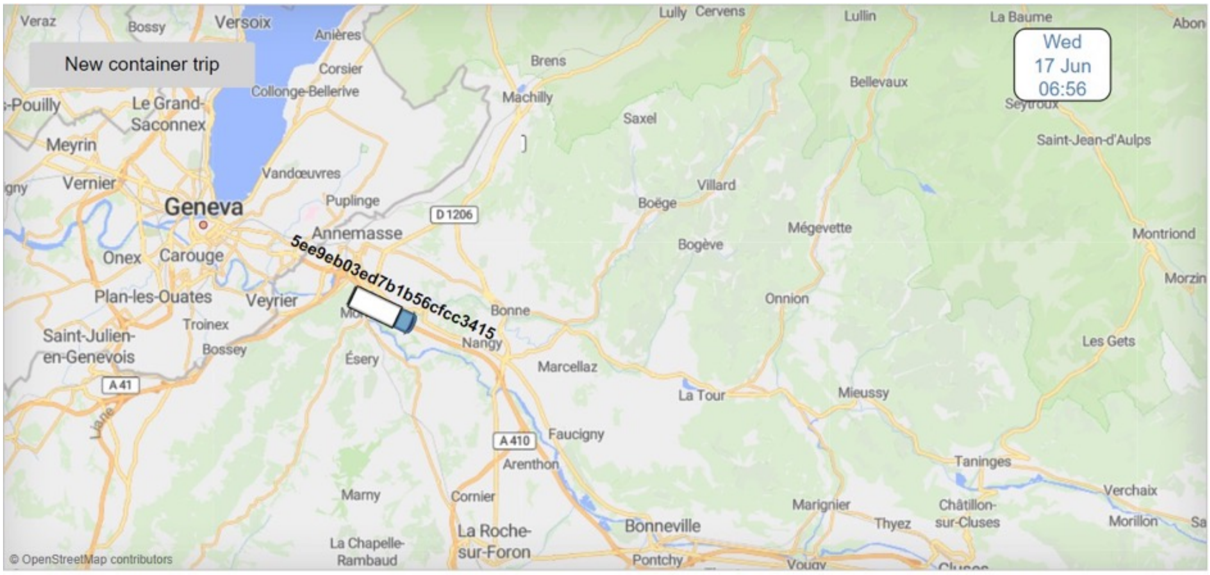


Figure 43 Simulator example

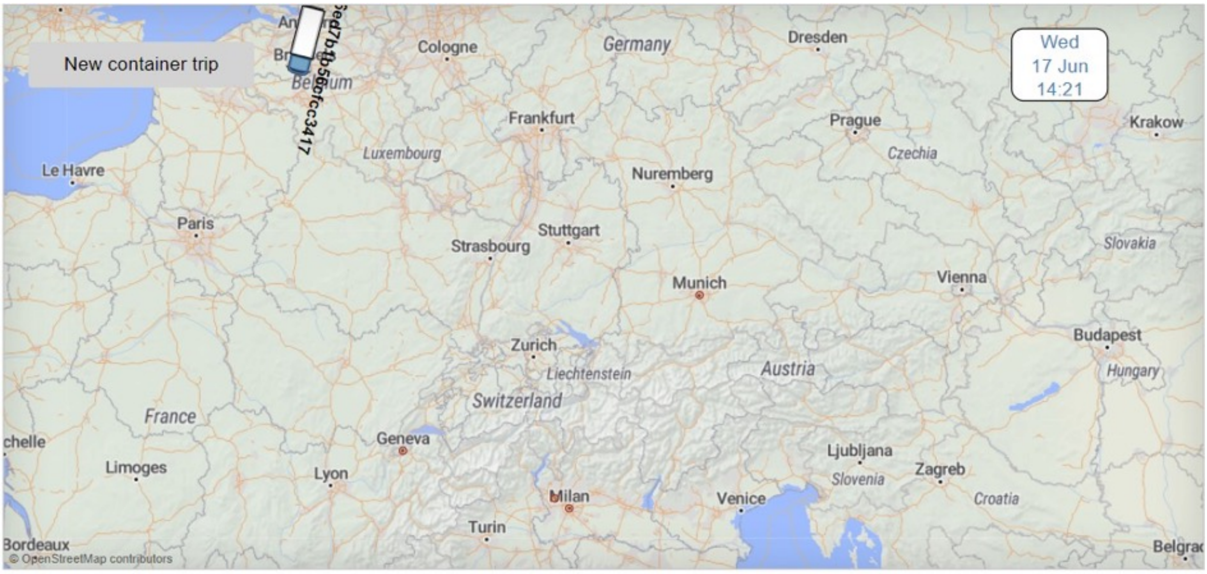


Figure 44 Simulator example

5.3 How to set up a PI order

Setting up a PI order in the PI environment means to define the rules of engagement of the shipment, thus configuring the data to be collected and the related events, as well as providing the start of the shipment. At this stage, a unique relation between the goods, the container and the devices installed on it must be generated, thus associating the involved entities, and allowing the ad-hoc sharing of the information regarding the shipment. The handshake shown in Figure 45 allows the setting-up of the PI order, implementing the following actions (all these actions are implemented following the guidelines defined in Sec. 5.1):

1. The PI order is sent by the PI Shipping Service and it is received by the Cloud IoT Platform allowing the unique association between it, the selected container and the connected IoT device. Moreover, the PI order is in charge of configuring the monitoring rules of the IoT device.
2. The platform set up the event manager with the received requirements (if any), see Sec. 5.1.1.
3. The platform acknowledges the PI reception and the correct configuration returning toward the PI Shipping Service a unique token, the “**ShipmentID**”, that identify uniquely the new shipment. This token will be used by the PI-users to retrieve in an ad-hoc manner the data associated to the considered shipment (for further details, look at Sec. 5.4.1).
4. The PI Shipping Service notifies the Cloud IoT Platform the start of the shipment: only at this time all the data collected by the related IoT device (exploiting the HTTPS “POST” method) are uniquely linked with the shipment.
5. The data are collected continuously and asynchronously by the IoT device and stored in the IoT platform. At this time, having available the ShipmentID and the authorisations to monitor the shipment, it is possible to retrieve real-time the information exploiting HTTPS APIs or the GUIs (described in Sec. 6 - The Cloud IoT Platform).
6. At the end of the shipment, the PI Shipping Service notifies this information the Cloud IoT Platform, thus “detaching” the information gathered by the IoT device installed on top of the container from the considered shipment. Of course, having available the ShipmentID and the correct authorisations to access, the information regarding the shipment can be retrieved also its end.

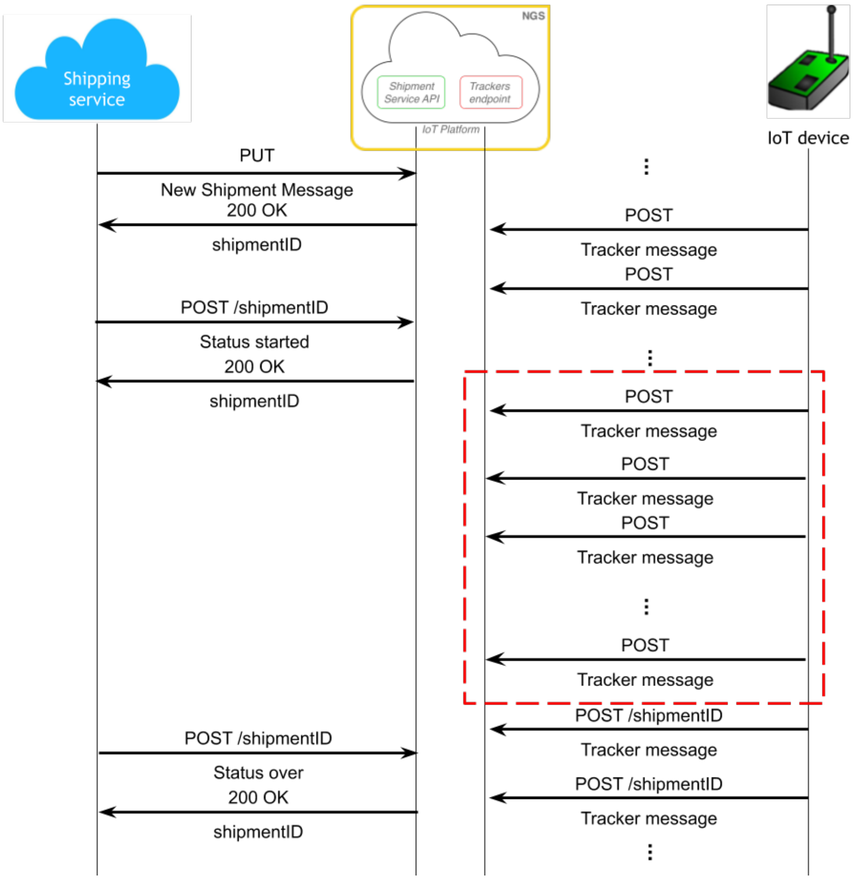


Figure 45 Handshake to set up a PI order

5.4 How to access the data, events, metadata and statistics

After the reception of the PI order and of the start of the shipment notification from the PI Shipping Service, data collected can be shared with both the Shipping Service module and all the authorised users in a secure and ad-hoc manner. Particularly, user-name and the password related to a certain user and the unique identifier "ShipmentID" must be authorised to collect data from the considered shipment. In this scenario, the following reporting operation can be implemented:

1. **Retrieving the data and the information gathered from the IoT device** deployed on the container exploiting REST interfaces, to feed PI environment as well as to integrate with third parties' platforms. This data and information can be retrieved during (real-time) and after the end of the shipment.
2. **Retrieving visual reports.** All the authorised users can require visual reports during (real-time) and after the end of the shipment.
3. **Retrieving events, metadata and statistics**. All the authorised users can require added value information regarding the considered shipment (i.e., events and metadata). Moreover, aggregated and anonymised statistics can be generated by the platform to dimension the performance of the specific routes or corridors.
4. **Pdf Shipment report**. After the end of the shipment, a complete pdf report containing all the information gathered from the field and computed (i.e., events, metadata and statistics) can be generated.

5.4.1 PI users' authentication

As for IoT devices (see Sec. 5.2.1), also PI users must authenticate within the Cloud IoT platform, thus implementing an ad-hoc access: the same standardised approach is implemented considering "standard basic authentication" of http that implement the IETF standard RFC 7617 (IETF, 2015).

However, while the IoT device must provide data continuously to the platform, the PI-users must access only to the information related to the shipments where it is directly involved, thus guaranteeing the privacy issues. As depicted in Figure 46, the PI users' involvement is communicated to the Cloud IoT service by the PI order, sent at the beginning of the transaction to set-up the shipment. In this scenario, the Cloud IoT platform generated a group of PI users identified by the retrieved "ShipmentID" (see Sec. 5.3), that allows them the ad-hoc access to the data and information gathered or computed.

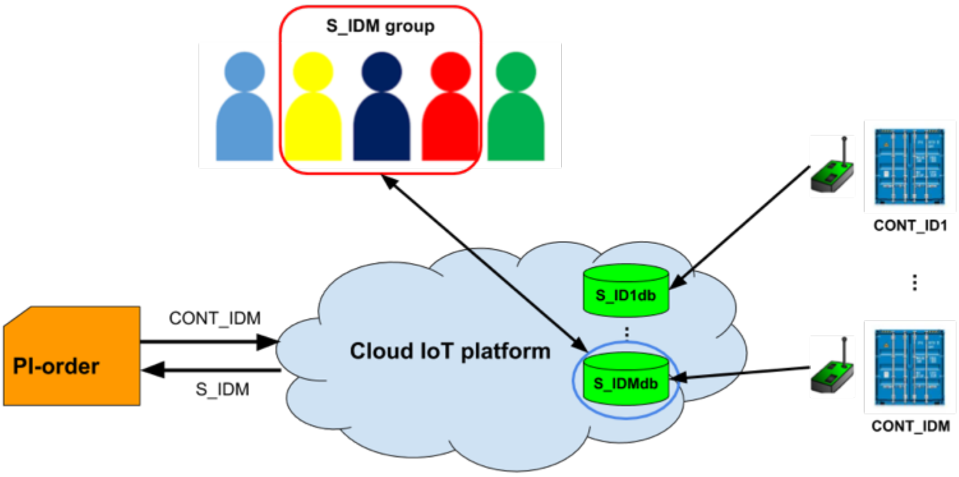


Figure 46 PI user ad-hoc access

5.4.2 Retrieving the data collected

To retrieve the data the PI Shipping Service must require the data related to the considered shipment using the unique "ShipmentID" exploiting the HTTPS "GET" method, as depicted in Figure 47: the platform will answer with the data-model defined in Sec. 5.1.2. All authorised PI users (see the previous section) can access in this manner, but also dedicated REST interfaces can be integrated on the Platform.

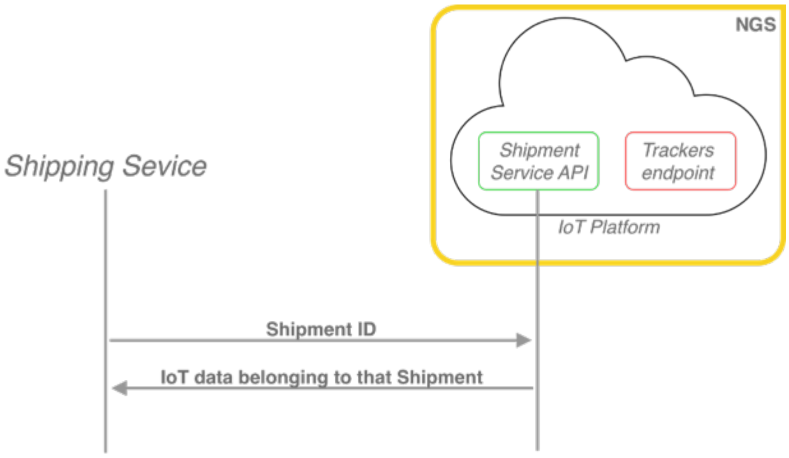


Figure 47 Handshake to retrieve the data (GET method)

5.4.3 Retrieving events, metadata and statistics

To retrieve events, metadata, and statistics a set of specific APIs will be developed considering the data-models described in Sec. 3.4.6. The information sharing can be done in an ad-hoc manner, as described in Sec. 5.4.1, following the rules described above:

1. **Events** related to a certain PI order can be shared with all the stakeholder authorised to access to both the platform and the considered shipment ID. As defined in Sec. 3.4.6, events are punctual so can be provided real time, during the shipment. Figure 48 shows the handshake to retrieve all the events happened during the shipment.

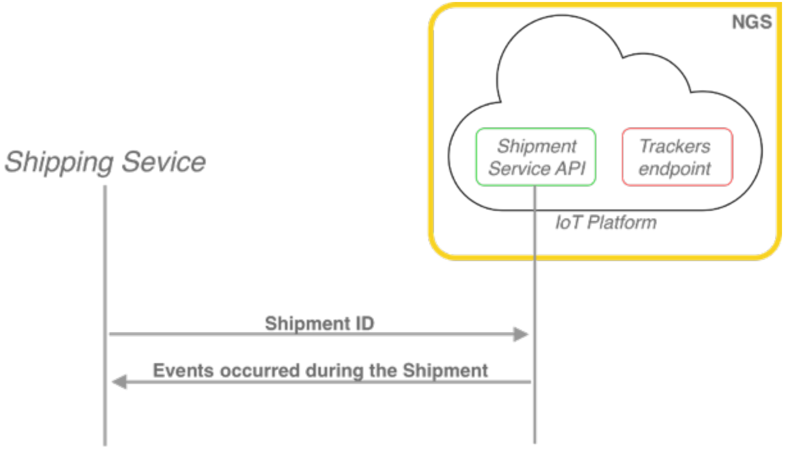


Figure 48 Handshake to retrieve the events (GET method)

2. **Metadata** related to a certain PI order can be shared with all the stakeholder authorised to access to both the platform and the considered shipment ID. Metadata are the output of the processing procedure, so it can be computed and shared only when the shipment is finished (i.e., status over notification is received) by the Cloud IoT service. Figure 49 shows the handshake to retrieve the metadata related to the shipment.

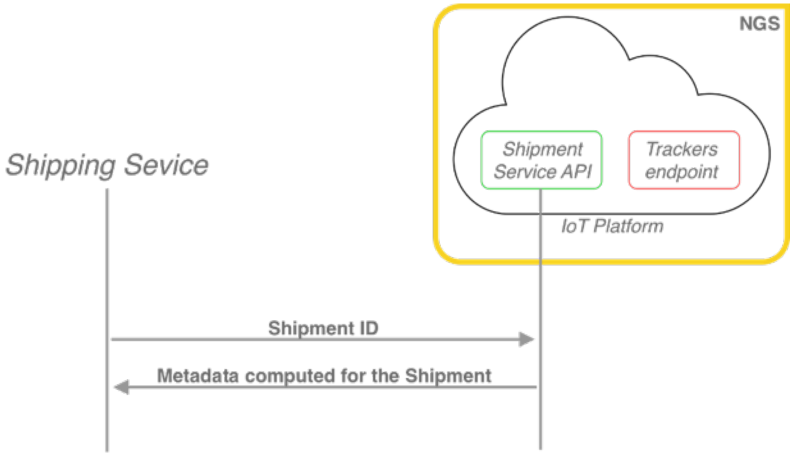


Figure 49 Handshake to retrieve the metadata (GET method)

3. **Statistics** represent aggregated data related with the analysis of multiple shipments (e.g., the average duration of the shipments in a certain corridor). In this case the handshake foresees parametric request (e.g., which corridor must be considered) as depicted in Figure 50.

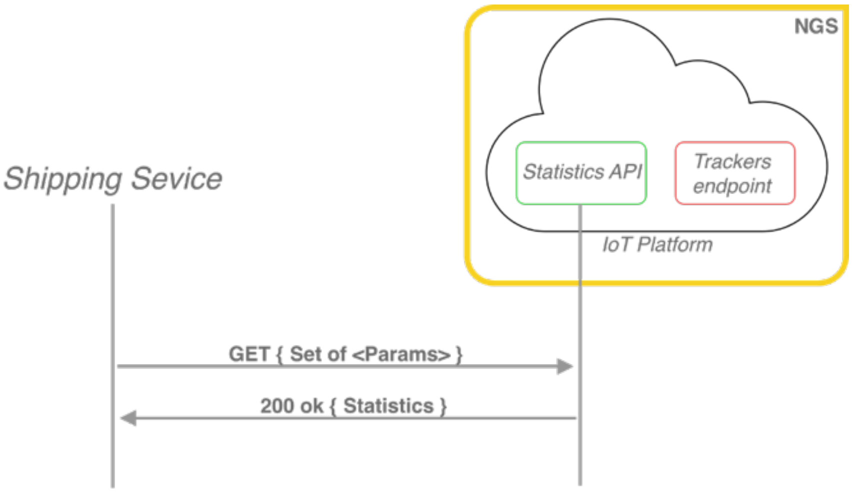
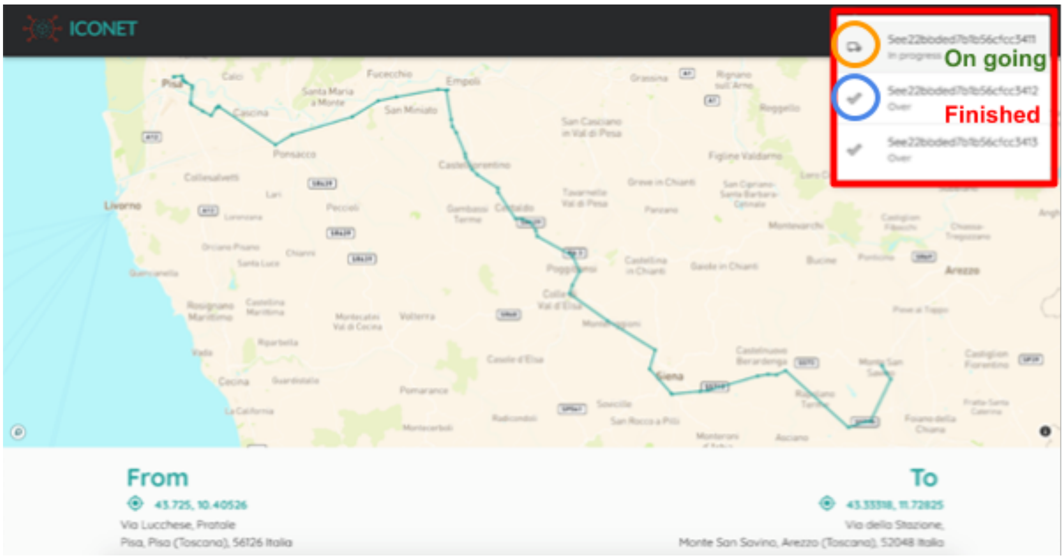


Figure 50 Handshake to retrieve the Avg. distance (GET method)

5.4.4 Retrieving visual reports

The data gathered and the information extracted can be represented as two different graphical reporting interfaces described below:

1. The **dashboard** (web page) where each PI-user can enter, authenticating its credentials (see Sec. 5.4.1), and visualise both (also during the shipment) all the shipments authorised (i.e., only the shipments related with the user) both in real-time and terminated shipments. The dashboard oversees visualising all the data and the information collected during the shipment, as well as all the metadata computed at the end of the transaction.



5ee22bbded7b1b56cfcc3411
In progress
On going

5ee22bbded7b1b56cfcc3412
Over
Finished

5ee22bbded7b1b56cfcc3413
Over

From
43.725, 10.40526
Via Lucchese, Pratale
Pisa, Pisa (Toscana), 56126 Italia

To
43.33318, 11.72825
Via della Stazione,
Monte San Savino, Arezzo (Toscana), 52048 Italia

Figure 51 Dashboard and ad-hoc access

2. A PDF report, automatically generated at the end of the shipment that contains all the data gathered during it, as well as all the events, metadata and statistics computed buy analytics engine. The pdf file can be retrieved implementing an http request to the Cloud IoT platform, of course providing the needed authorisation credentials (username, password, shipmentID).

5.5 Toward the supply chain complete visibility

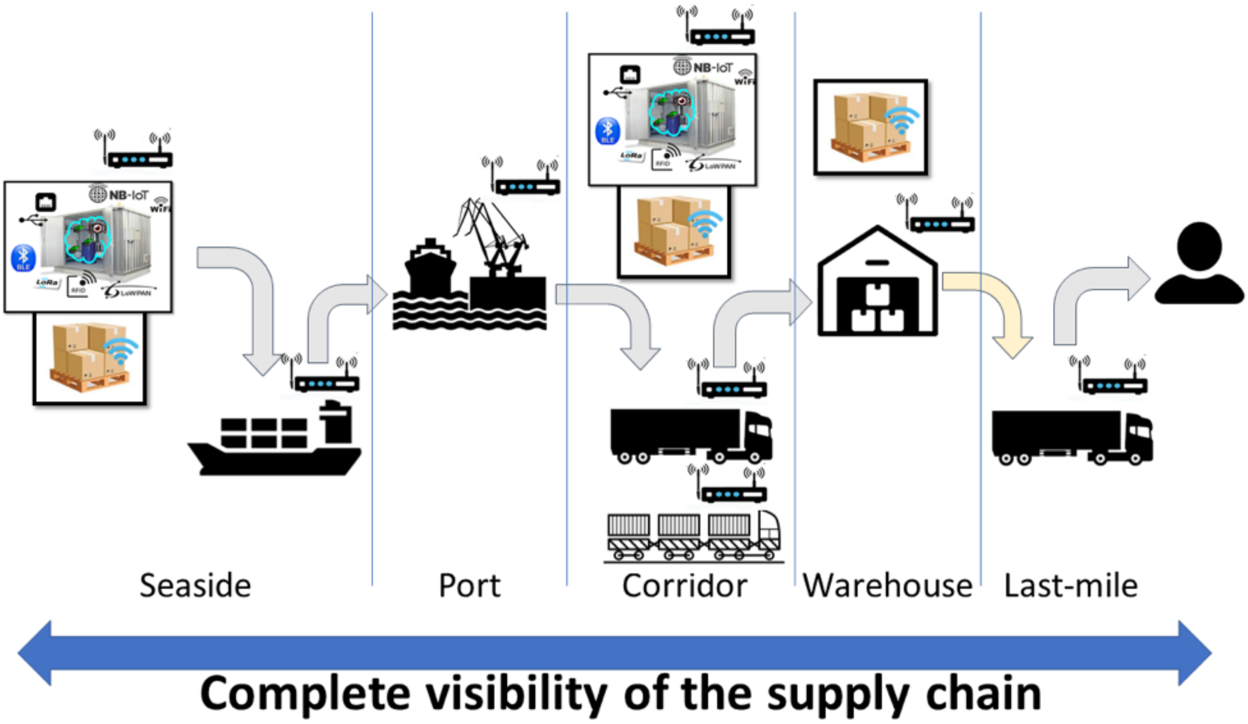


Figure 52 Supply chain complete visibility

Figure 52 represent how we want to transform the current supply chain scenario (depicted in Figure 1), implementing its complete visibility: an environment ubiquitously connected where goods can be tracked and monitored at different granularity, thus providing added value information from the sender facility toward the final used, passing through all the ICONET LLs. In this scenario, the Smart PI Pallets is assembled within the sender facility and encapsulated in the Smart PI Container. At this point their shipment is started, and the tracking and monitoring service is instantiated (as described in the previous sections). Knowing the correct position of the container/pallet, the goods are routed all along intermodal PI corridors and PI hubs, until the arrival at the “proxy” warehouse. The shipment is tracked and monitored all along the logistics network, exploiting the added value information coming from the IoT devices. This allows the implementation of adaptive routing actions, effective and fact-based operations, and the enablement of a deterministic scheduling of the activities. When the container, reach the “proxy” warehouse it is disassembled and the Smart PI Pallet can be tracked and monitored exploiting the IoT gateways installed on the PI infrastructure, as well as in the PI Means in the last-mile logistics operations.

5.5.1 LL2 implementation and the COVID-19 pandemic

In LL2 – Corridor Centric PI Network the ICONET Project investigates the progressive transformation of a typical transport corridor into a PI corridor. LL2 objectives regard:

1. The realisation of an IoT system capable to implement the goods’ tracking and the monitoring functionalities all along the PI corridors.
2. The improvement of the efficiency of the supply chain, toward the realisation of an optimised logistics network, less expensive and greener.
3. The increase of the reliability of multimodal transports, therefore facilitating syncro-modality at operational level and enabling proactive, data-oriented, and fact-oriented decision making.

IoT’s application in the real-life transport corridors is investigated through the installation of trackers in containers and reporting information exploiting GUIs and PDF reports, enhancing visibility and container’s location awareness, expected to lead in optimized utilization of both transport and logistics resources. All the shipments considered in LL2 consist of fully loaded containers starting and ending from warehouses, and only tracking, and on-board sensors’ monitoring services are required. No other sensors inside or outside (the smart seal) are required and allowed.

As per the project parameters and LL2 set up, the tracking and monitoring of the containers for transporting the goods on two multimodal corridors from Belgium to UK and to Italy, shown in Figure 53 and Figure 54 was established. To implement this service, two containers by the ITC had to be dedicated for this task, and two trackers were to be prepared, installed and activated in accordance with the LL2 plan.

The pandemic took a toll on the supply Chain activities and understandably the deployment of the trackers was hugely delayed. The disruptions in the traffic across Logistics corridors created in the line of service, delayed the whole deployment process in terms of the setup, charging, interfacing, testing and installing of the devices. Moreover, the integration activities with external platform was also affected. In an effort to mitigate the negative impact of the conditions (as discussed previously in this document) in close collaboration with the partner and the ITC, specific data sets were used as per agreed scenarios and respecting confidentiality conditions certain evaluation and data exploitation resulted thereon. Furthermore, even more devices were deployed to alleviate the situation and accelerate the data recording of the devices in the field.

Other issues caused by the COVID-19 pandemic related to LL2 and the derived mitigation activities are detailed in Sec. 2.2.

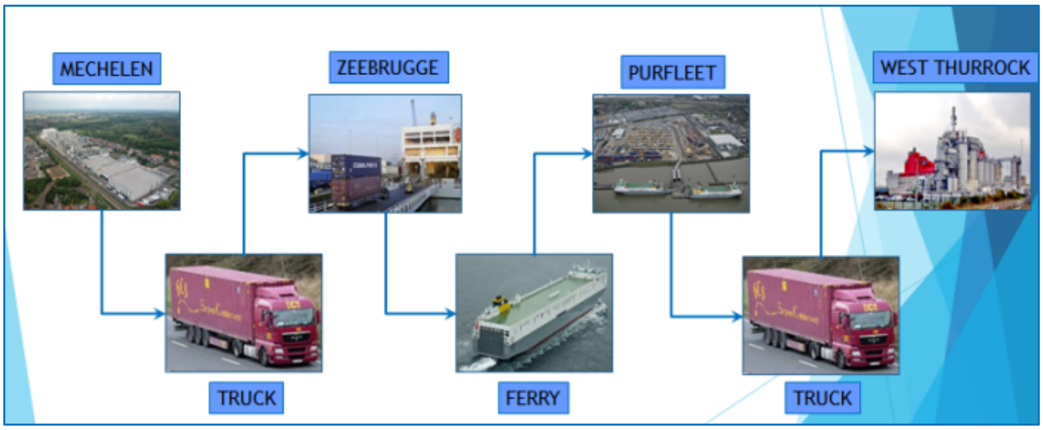


Figure 53 Corridor Mechelen (B) – West Thurrock (UK)

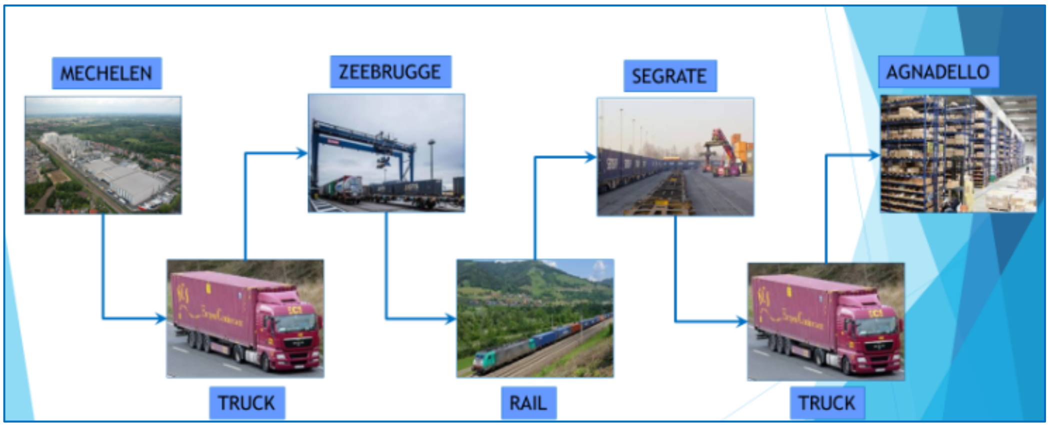
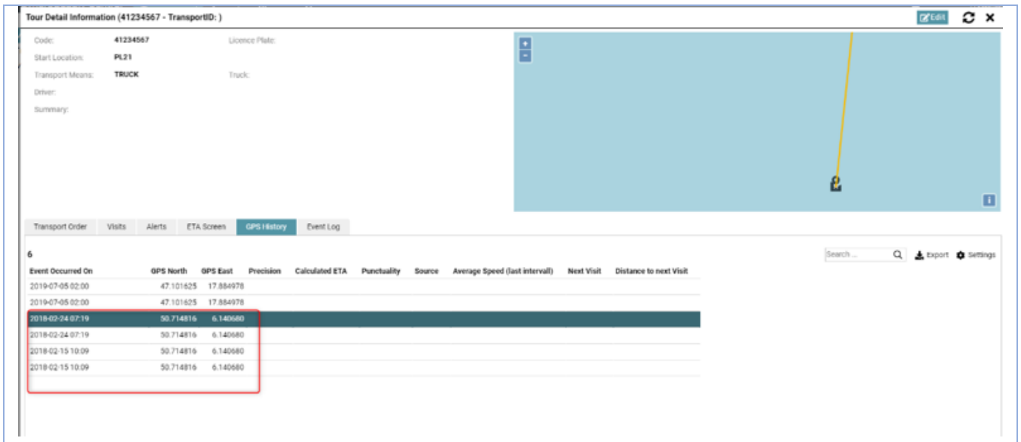


Figure 54 Corridor Mechelen (B) – Agnadello (I)

5.5.2 Challenges faced and results for LL2

In the following table, the challenges faced and the achieved results to realise the first prototype the LL2 are listed.

Challenge	***Description and results***
Exploitation of NB-IoT realise an energy efficient tracker	Currently, NB-IoT is not available for international logistics transactions (no international roaming, for more details see D2.6). For this reason, GPRS is considered and an energy saving protocol based on duty cycle is implemented (data are sampled and dispatched every 10 min). To obtain improved results on battery duration, a new board is developed to optimise the power consumption
Interoperability	The standardisation scenario is analysed to be compliant with the standardisation paths related to the logistics environment. All the works described in this deliverable are designed to be compliant with this standardisation framework (related to the technical and syntactic interoperability), as detailed in Sec. 3.3. Future developments of the prototypes described in this report will follow the guidelines of the forthcoming standards, to reach higher levels of interoperability. This approach has allowed on one hand to ease the integration with other partners modules, on the other to dispatch the data gathered by the IoT Cloud Platform toward third parties platform, such as Synchrosupply by Inform. In the latter case, a preliminary integration through a secure POST transaction is done, integrating with Synchrosupply (see Figure 55, where the screenshot of visualisation of the received messages is shown). However, further and more detailed tests will be realised in the next months.



Tour Detail Information (41234567 - TransportID:)

Code:	41234567	Licence Plate:	
Start Location:	PL21		
Transport Means:	TRUCK	Truck:	
Driver:			
Summary:			

6

Event Occurred On	GPS North	GPS East	Precision	Calculated ETA	Punctuality	Source	Average Speed (last interval)	Next Visit	Distance to next Visit
2019-07-05 02:00	47.101625	17.884978							
2019-07-05 02:00	47.101625	17.884978							
2018-02-24 07:19	50.714816	6.140680							
2018-02-24 07:19	50.714816	6.140680							
2018-02-15 10:09	50.714816	6.140680							
2018-02-15 10:09	50.714816	6.140680							

Figure 55 Preliminary integration test with Synchrosupply

Resiliency on network leaks	The system is configured to store in a limited buffer the data collected when the connectivity is missing (e.g., in the middle of the sea).
On-board sensors sampling	The system collect measurement coming from the on-board sensors, dispatching those remotely. Some sensors are configured to trigger events, also during their sleep state.
Events' notifications	We have focused on the acceleration analysis, implemented following an asynchronous approach (i.e., interrupts). These analyses have allowed to implement: 1. The motion detection is based on thresholds obtained following an empirical approach. The motion detection is a useful data to determine how long a container stays stopped in a certain zone, as well as an important information to improve the power saving policy. 2. The bump detection is based on the results coming from some scientific papers and it allows the determination of the bumps that affect the container. Finally, configurable threshold exceedance functionalities are implemented to generate events related with the temperature and/or the humidity and/or light. These functionalities can be enabled and configured by the PI order during the shipment setting-up procedures (see Sec. 5.3).

5.5.3 LL2 System performance

To demonstrate the working principle of the proposed solution, in the following a container is followed in his trips all along EU. We consider data gathered from the simulator (see Sec. 5.2.3), to have a general vision of all the trips of a container, thus demonstrating the logistics network performance in terms of efficiency in both the goods dispatchment and the container usage.

In this scenario, we have generated 17 trips for the container “5eb59a6fcfd1014c38d15688” as detailed in Table 14, where the start and the end of the shipment and the “shipmentID” assigned by the Cloud IoT services to each PI order.

Table 14 Lists of trips for the container "5eb59a6fcfd1014c38d15688"

shipmentID	containerID	From	To
5f44bce0ec0aa4db149695ac	5eb59a6fcfd1014c38d15688	Agnadello, Milano (IT)	West Thurrock, Kent (GB)
5f44bf0dec0aa4db149695ad	5eb59a6fcfd1014c38d15688	West Thurrock, Kent (GB)	Agnadello, Milano (IT)
5f44c0d8ec0aa4db149695ae	5eb59a6fcfd1014c38d15688	Agnadello, Milano (IT)	Rumst, Antwerpen (BE)
5f44c29fec0aa4db149695af	5eb59a6fcfd1014c38d15688	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)
5f44c2f1ec0aa4db149695b0	5eb59a6fcfd1014c38d15688	West Thurrock, Kent (GB)	Agnadello, Milano (IT)
5f44c384ec0aa4db149695b1	5eb59a6fcfd1014c38d15688	Agnadello, Milano (IT)	West Thurrock, Kent (GB)
5f44c477ec0aa4db149695b2	5eb59a6fcfd1014c38d15688	West Thurrock, Kent (GB)	Agnadello, Milano (IT)
5f44c53bec0aa4db149695b3	5eb59a6fcfd1014c38d15688	Agnadello, Milano (IT)	Rumst, Antwerpen (BE)
5f44c598ec0aa4db149695b4	5eb59a6fcfd1014c38d15688	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)
5f44c60fec0aa4db149695b5	5eb59a6fcfd1014c38d15688	West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)
5f44c647ec0aa4db149695b6	5eb59a6fcfd1014c38d15688	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)
5f44c6ccec0aa4db149695b7	5eb59a6fcfd1014c38d15688	West Thurrock, Kent (GB)	Agnadello, Milano (IT)
5f44c89bec0aa4db149695b8	5eb59a6fcfd1014c38d15688	Agnadello, Milano (IT)	Rumst, Antwerpen (BE)
5f44c8e2ec0aa4db149695b9	5eb59a6fcfd1014c38d15688	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)
5f44c945ec0aa4db149695ba	5eb59a6fcfd1014c38d15688	West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)
5f44c971ec0aa4db149695bb	5eb59a6fcfd1014c38d15688	Rumst, Antwerpen (BE)	Agnadello, Milano (IT)
5f44c9bdec0aa4db149695bc	5eb59a6fcfd1014c38d15688	Agnadello, Milano (IT)	West Thurrock, Kent (GB)

When each shipment starts (after the reception of the start signal), all the data gathered from the IoT devices related to the considered container are made available to the group of stakeholders identified by the assigned "shipmentID": this data can be gathered by those exploiting the APIs described in Sec. 5.4 or the GUI (raw data and events can be gathered real-time, metadata only after the end of the shipment). In Figure 56 the screenshot of the GUI related of the shipment "5f44c0d8ec0aa4db149695ae" route is depicted, highlighting the point of sampling (black dots) as well as where events happens (red dots, in this case only where the shipment was stopped). Figure 57 shows the real time measurements during the shipments, in this case the temperature inside the container.

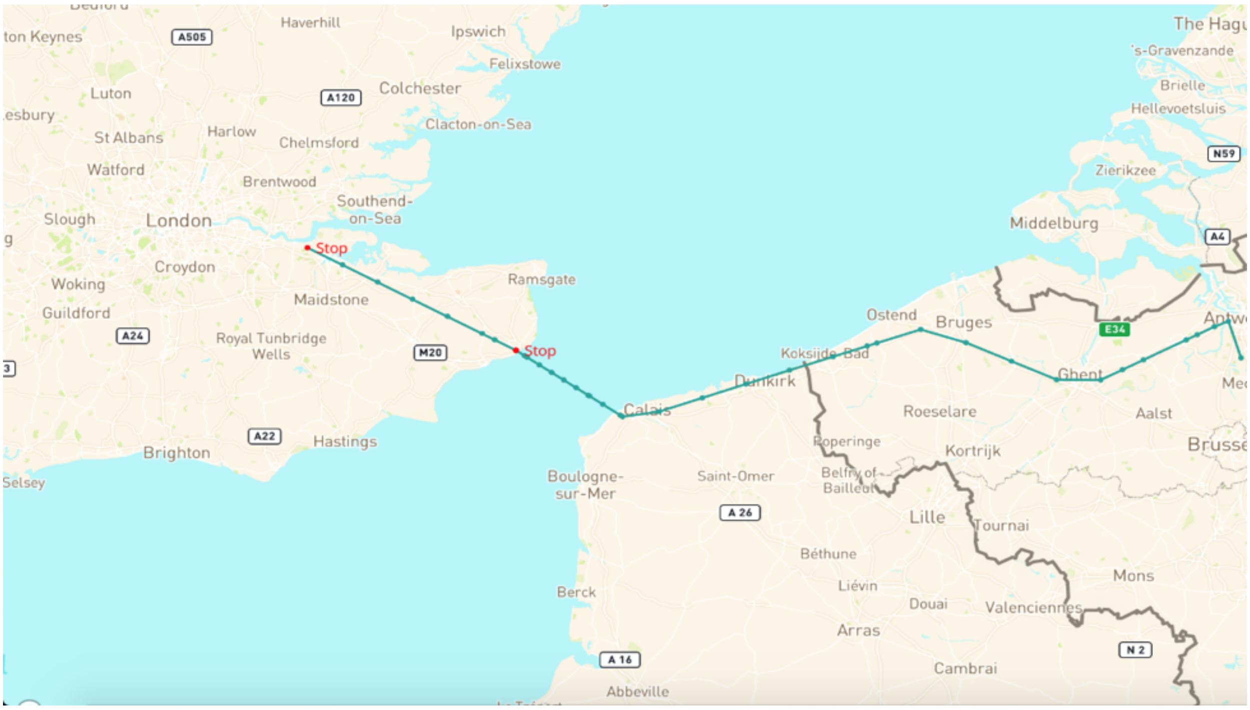


Figure 56 Route related at the shipment 5f44c0d8ec0aa4db149695ae

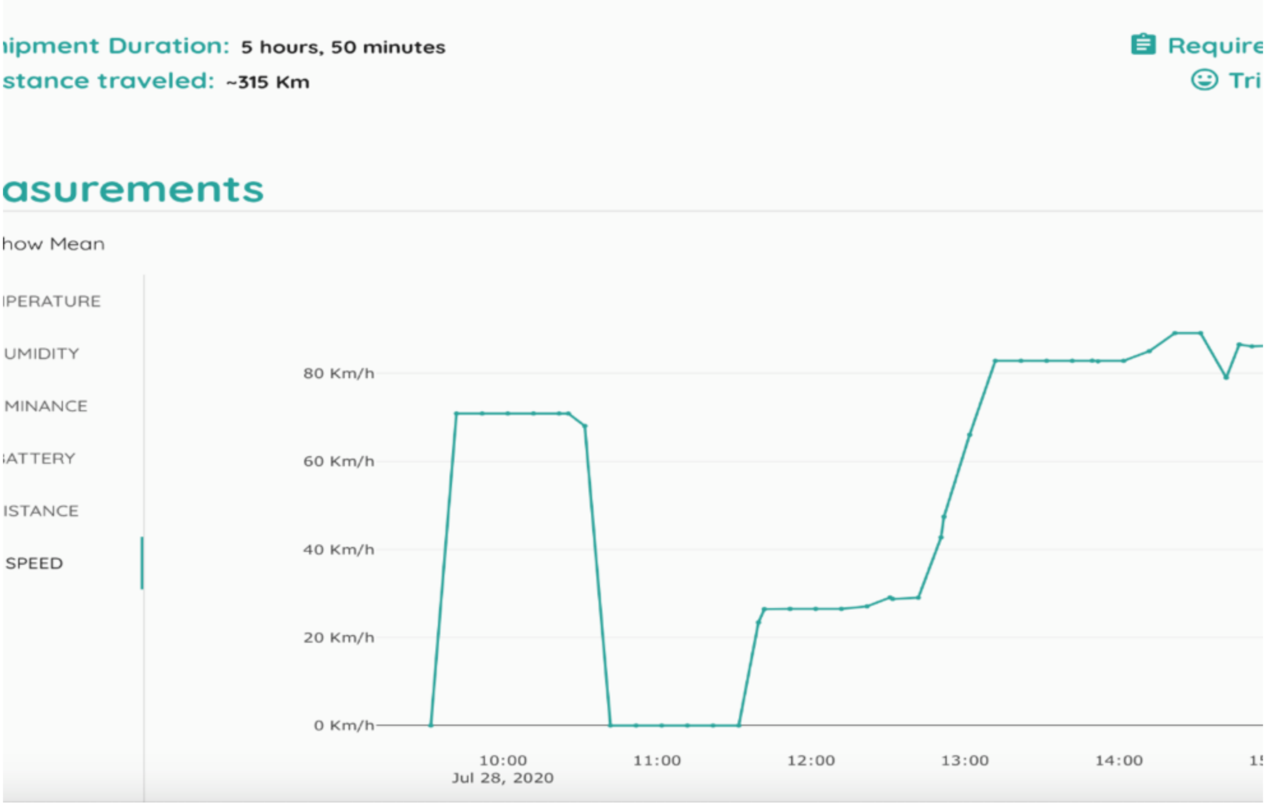
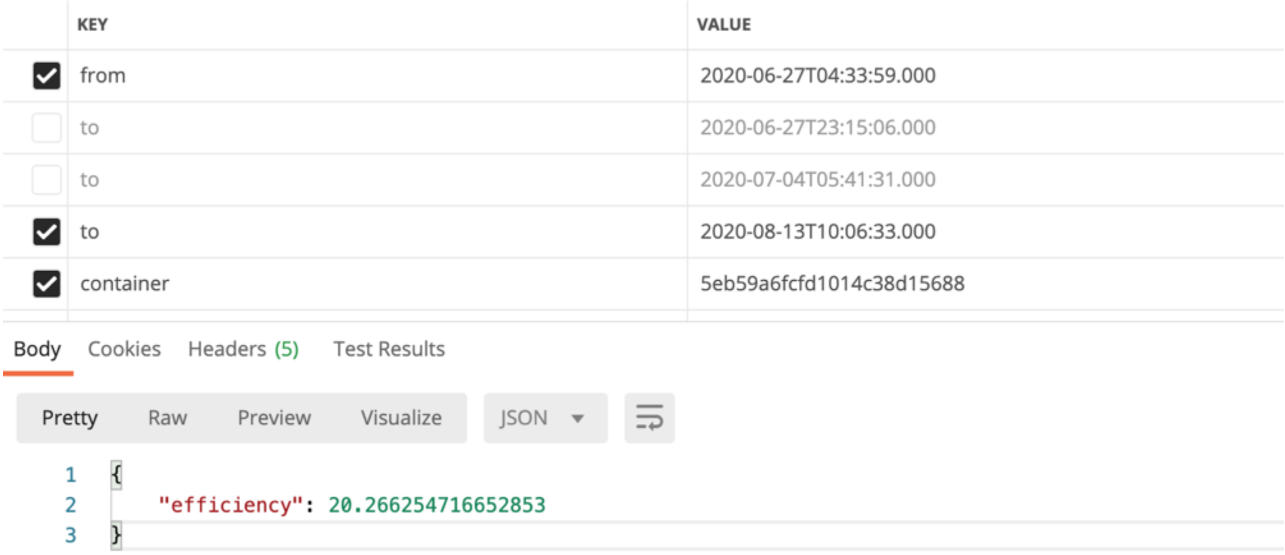


Figure 57 Shipment real time measurements

Finally, when the shipment is ended, the container is disassociated to the transaction, all the metadata are computed, and all the statistics updated. Table 15 shows the summary of the statistics parameters computed for each shipment. These parameters support the logistics decision making operations to understand the efficiency of the transaction, analysing the efficiency information provided. Particularly, analysing the Shipment efficiency in Table 15, the stop time with respect of the moving time of the shipment can be evaluated, while the raw data regarding the routes (e.g., Figure 56 and Figure 57) can support the understanding of the reasons of this stops (e.g., delay on crossdocking, queues, etc.). This analysis can be supported by the comparison between the average speed and the average speed without stops. The statistics, shown in Table 16, allow to provide a good comparison floor to understand the performance of the considered shipment. As matter of example, we consider the shipment identified with the ID 5f44c60fec0aa4db149695b5 from Rumst, Antwerpen (BE) to Rumst, Antwerpen (BE). It is the less efficient shipment in the selected, having a short trip and a long stop for waiting the vessel to pass the English Channel (see Figure 56 and Figure 57). However, the duration of the trip is the minimum computed, as detailed in Table 16. As in this scenario, such instruments can support the decision making and the proactive routing.

Finally, Figure 58 show the efficiency of exploitation of the container in the period related with the shipment detailed in Table 15. It provide an import instruments to evaluate the real usage of the containers, supporting asset management functionalities toward the service costs' reduction.



	KEY	VALUE
☑	from	2020-06-27T04:33:59.000
□	to	2020-06-27T23:15:06.000
□	to	2020-07-04T05:41:31.000
☑	to	2020-08-13T10:06:33.000
☑	container	5eb59a6fcfd1014c38d15688

```
{
    "efficiency": 20.266254716652853
}
```

Figure 58 Container Efficiency

Table 15 Shipments summary table

shipmentID	containerID	startsAt	endsAt	Duration [hh:mm:ss]	From	To	Distamce [Km]	Stop Time [min]	Efficiency[6] [%]	Speed Avg. [Km/h]	Speed Avg. NoStop [Km/h]
5f44bce0ec0aa4db149695ac	5eb59a6fcfd1014c38d15688	2020-06-27T04:33:59.000Z	2020-06-27T23:15:06.000Z	18:41:07	Agnadello, Milano (IT)	West Thurrock, Kent (GB)	~1138.8	01:00:00	94.65	~62.4	~65.2
5f44bf0dec0aa4db149695ad	5eb59a6fcfd1014c38d15688	2020-07-01T07:22:00.000Z	2020-07-02T02:02:06.000Z	18:40:06	West Thurrock, Kent (GB)	Agnadello, Milano (IT)	~1131.0	00:40:00	96.43	~61.8	~63.7
5f44c0d8ec0aa4db149695ae	5eb59a6fcfd1014c38d15688	2020-07-03T10:11:31.000Z	2020-07-04T05:41:31.000Z	19:30:00	Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	~1160.4	01:00:00	94.87	~60.0	~62.7
5f44c29fec0aa4db149695af	5eb59a6fcfd1014c38d15688	2020-07-13T19:06:12.000Z	2020-07-14T02:16:12.000Z	07:10:00	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	~315.52	00:50:00	88.37	~44.5	~49.8
5f44c2f1ec0aa4db149695b0	5eb59a6fcfd1014c38d15688	2020-07-14T16:04:05.000Z	2020-07-15T07:34:05.000Z	15:30:00	West Thurrock, Kent (GB)	Agnadello, Milano (IT)	~1073.1	00:50:00	94.62	~70.4	~73.5
5f44c384ec0aa4db149695b1	5eb59a6fcfd1014c38d15688	2020-07-20T13:28:09.000Z	2020-07-21T08:38:09.000Z	19:10:00	Agnadello, Milano (IT)	West Thurrock, Kent (GB)	~1123.1	01:00:00	94.78	~59.4	~62.2
5f44c477ec0aa4db149695b2	5eb59a6fcfd1014c38d15688	2020-07-21T14:10:41.000Z	2020-07-22T07:30:41.000Z	17:20:00	West Thurrock, Kent (GB)	Agnadello, Milano (IT)	~1236.9	01:00:00	94.23	~72.5	~76.2
5f44c53bec0aa4db149695b3	5eb59a6fcfd1014c38d15688	2020-07-24T10:57:38.000Z	2020-07-25T03:17:38.000Z	16:20:00	Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	~952.0	01:00:00	93.88	~58.9	~61.8
5f44c598ec0aa4db149695b4	5eb59a6fcfd1014c38d15688	2020-07-27T08:59:11.000Z	2020-07-27T13:39:11.000Z	04:40:00	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	~307.74	00:30:00	89.29	~65.9	~73.3
5f44c60fec0aa4db149695b5	5eb59a6fcfd1014c38d15688	2020-07-28T09:31:46.000Z	2020-07-28T15:21:46.000Z	05:50:00	West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	~316.0	01:00:00	82.86	~55.9	~65.0
5f44c647ec0aa4db149695b6	5eb59a6fcfd1014c38d15688	2020-07-28T21:21:56.000Z	2020-07-29T02:27:57.000Z	05:06:01	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	~320.41	00:40:00	86.93	~65.2	~72.9
5f44c6ccec0aa4db149695b7	5eb59a6fcfd1014c38d15688	2020-08-03T06:34:25.000Z	2020-08-04T00:54:25.000Z	18:20:00	West Thurrock, Kent (GB)	Agnadello, Milano (IT)	~1087.6	00:40:00	96.36	~59.9	~61.7

[6] Shipment efficiency

5f44c89bec0aa4db149695b8	5eb59a6fcfd1014c38d15688	2020-08-05T08:04:17.000Z	2020-08-05T20:44:17.000Z	12:40:00	Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	~816.8	01:00:00	92.11	~65.0	~70.0
5f44c8e2ec0aa4db149695b9	5eb59a6fcfd1014c38d15688	2020-08-06T06:01:08.000Z	2020-08-06T12:27:30.000Z	06:26:22	Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	~313.8	00:50:00	87.06	~50.9	~57.1
5f44c945ec0aa4db149695ba	5eb59a6fcfd1014c38d15688	2020-08-10T11:27:48.000Z	2020-08-10T18:47:48.000Z	07:20:00	West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	~316.7	01:10:00	84.09	~47.1	~54.2
5f44c971ec0aa4db149695bb	5eb59a6fcfd1014c38d15688	2020-08-11T13:06:01.000Z	2020-08-12T06:56:01.000Z	17:50:00	Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	~1165.0	00:50:00	95.33	~65.3	~68.0
5f44c9bdec0aa4db149695bc	5eb59a6fcfd1014c38d15688	2020-08-12T14:56:33.000Z	2020-08-13T10:06:33.000Z	19:10:00	Agnadello, Milano (IT)	West Thurrock, Kent (GB)	~1145.1	01:10:00	93.91	~60.0	~63.2

Table 16 Statistics

From	To	Avg Duration [hh:mm:ss]	Min Duration [hh:mm:ss]	Max Duration [hh:mm:ss]
Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	16:10:00	12:40:00	19:30:00
Agnadello, Milano (IT)	West Thurrock, Kent (GB)	19:00:22	18:41:07	19:10:00
Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	17:50:00	17:50:00	17:50:00
Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	05:50:36	04:40:00	07:10:00
West Thurrock, Kent (GB)	Agnadello, Milano (IT)	17:27:31	15:30:00	18:40:06
West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	06:35:00	05:50:00	07:20:00
From	**To**	**Avg Stop Time [hh:mm:ss]**	**Min Stop Time [hh:mm:ss]**	**Max StopTime [hh:mm:ss]**
Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	01:00:00	01:00:00	01:00:00
Agnadello, Milano (IT)	West Thurrock, Kent (GB)	01:03:20	01:00:00	01:10:00
Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	00:50:00	00:50:00	00:50:00
Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	00:42:30	00:30:00	00:50:00
West Thurrock, Kent (GB)	Agnadello, Milano (IT)	00:47:30	00:40:00	01:00:00
West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	01:05:00	01:00:00	01:10:00
From	**To**	**Avg Efficiency [%]**	**Min Efficiency [%]**	**Max Efficiency [%]**
Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	94	92	95
Agnadello, Milano (IT)	West Thurrock, Kent (GB)	94	94	95
Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	95	95	95
Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	88	87	89
West Thurrock, Kent (GB)	Agnadello, Milano (IT)	95	94	96
West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	83	82	84
From	**To**	**Avg Distance Traveled [Km]**	**Min Distance Traveled [Km]**	**Max Distance Traveled [Km]**
Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	976	817	1160
Agnadello, Milano (IT)	West Thurrock, Kent (GB)	1136	1123	1145
Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	1165	1165	1165
Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	314	308	320
West Thurrock, Kent (GB)	Agnadello, Milano (IT)	1132	1073	1237
West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	316	316	317
From	**To**	**Avg Avg. Speed [Km/h]**	**Min Avg. Speed [Km/h]**	**Max Avg. Speed [Km/h]**
Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	61	59	65
Agnadello, Milano (IT)	West Thurrock, Kent (GB)	61	59	62
Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	65	65	65
Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	57	44	66
West Thurrock, Kent (GB)	Agnadello, Milano (IT)	66	60	73
West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	52	47	56
From	**To**	**Avg Avg. Speed NoStop [Km/h]**	**Min Avg. Speed NoStop [Km/h]**	**Max Avg .Speed NoStop [Km/h]**

Agnadello, Milano (IT)	Rumst, Antwerpen (BE)	65	62	70
Agnadello, Milano (IT)	West Thurrock, Kent (GB)	64	62	65
Rumst, Antwerpen (BE)	Agnadello, Milano (IT)	68	68	68
Rumst, Antwerpen (BE)	West Thurrock, Kent (GB)	63	50	73
West Thurrock, Kent (GB)	Agnadello, Milano (IT)	69	62	76
West Thurrock, Kent (GB)	Rumst, Antwerpen (BE)	60	54	65

6. The Cloud IoT Platform

The Cloud IoT platform will be implemented to realise a set of seamless interfaces with the digital world components, thus providing reports to the legacy systems, as well as the PI components developed within the ICONET project, as depicted in Figure 59. In this section, the final architecture and implementation of Cloud IoT platform will be completely described.

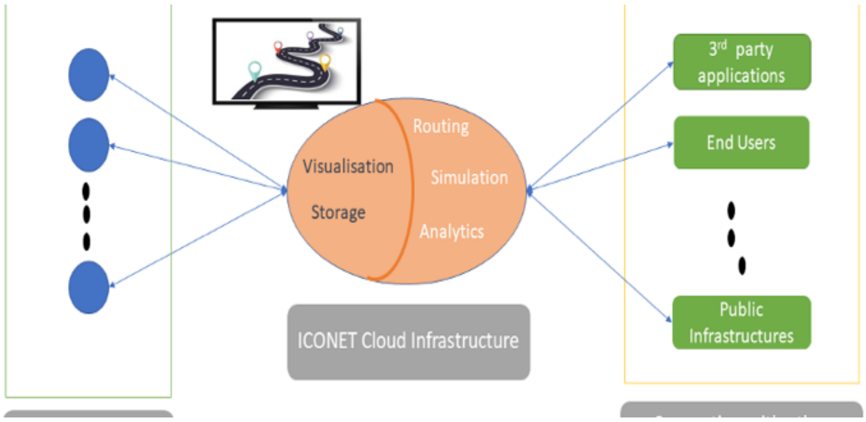


Figure 59 The Cloud IoT Platform

Figure 62 depicts the final architecture of the Cloud IoT platform developed within ICONET. It implements essentially the following functionalities:

1. Interoperate with a scalable set of IoT devices (trackers, sensors, smart routers, and smart gateways) capable to collect data from the field.
2. Store the data collected from the field in a database.
3. Report the data to other stakeholders' digital systems with a scalable and interoperable set of APIs, as well as with the visualisation web portal.

The Cloud IoT Platform will be based on the improvement of the platform by NGS developed for the smart factory and predictive maintenance, called Smart PlantOne (NGS, s.d.). This solution is stable and tested in real industrial environments (e.g., ENEL Federico II power plant). For this reason, we have preferred to improve this already available platform domain, instead of re-implementing a new one based on other technologies (as for example FIWARE).

Within ICONET, the PlantOne platform is improved is restructured to support a scalable set of IoT devices and of requests (users), thus implementing new interfaces (i.e., the Shimpment Service and the Shipment Statistics APIs) and new functionalities (i.e., the analytics engine).

In the following the final release of the IoT Cloud Platform is described, in terms of its integration within the ICONET Cloud PoC as well as on its components.

6.1 The ICONET Cloud PoC overview

The main objective of the ICONET PoC environment is to facilitate technical development and collaboration between the WP2 partners, as well as providing instruments for the realization of the LLs environments (WP3). The PoC environment is considered to be a valuable area for all assets developed through the project and also for testing multiple deployments and scenarios. From a multi-component integration perspective, multiple integration scenarios can be securely supported by the PoC.

The PoC is deployed within an AWS space made available by IBM. AWS allows for the deployment of Virtual Private Clouds (VPCs) and creation of many VPCs in a single AWS account, but still provides secure and separate networks facilitating inter-VPC network communication through appropriate configuration. The internal layout of the first VPC can be presented in Figure 60. Though there are different configured VPCs for a variety of goals created for WP2, the first VPC is noteworthy due to its adoption of a broad range of possible configurations, assets, and frameworks. The VPC divides into one public and two private subnets for security-enhancing purposes. A NAT Gateway is configured for the private subnets to allow network traffic to leave but block any external traffic inbound. Besides, an Internet Gateway is exploited for the public subnet to allow both inbound and outbound traffic. In order to support developers' access in a secure manner, a VPN has been configured to enable developers to access their own and other services deployed within those subnets.

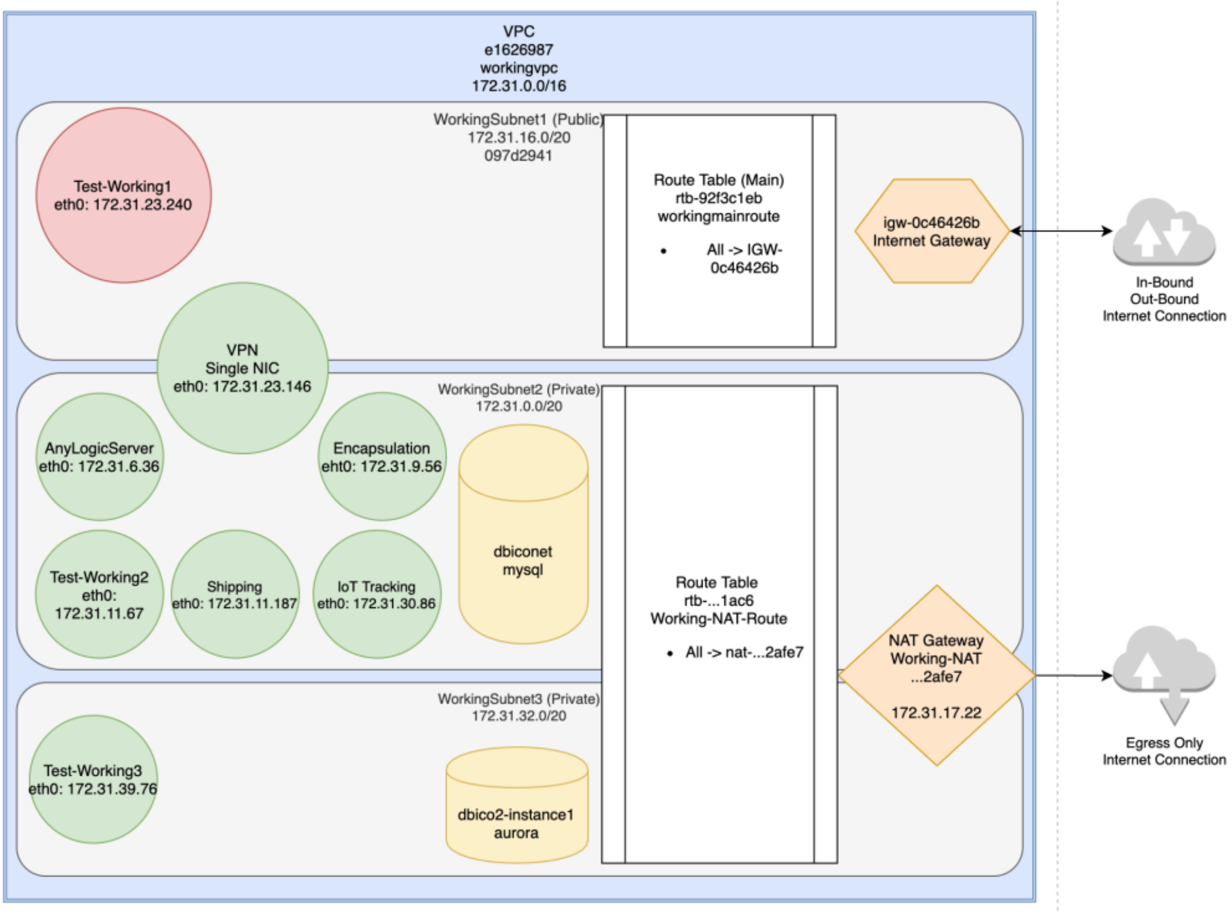


Figure 60 The ICONET PoC - AWS VPC Network Map

6.2 The Cloud IoT Platform – Final Release

Figure 61 depicts the final shape of the Cloud IoT platform, breaking it down into its main components and functionalities. Particularly, the architecture we have proposes will oversee implementing the following logical functionalities:

1. **The IoT manager**, in charge of managing the ingestion a scalable set of transaction coming from the IoT devices, thus understanding their consistency in terms of accepted data-models
2. **The Storage manager**, to implement the data persistence and storage functionalities for the data managed by the ingestion service.
3. **Analytics**, in charge of processing the data stored toward the generation of events, metadata and statistics, see also Sec.3.3.3.
4. **Interactions/APIs,** allowing the data exchange with third parties stakeholders involved in the shipment transaction.
5. **The Graphical User Interface,** allowing the data visualization and the real time reporting.

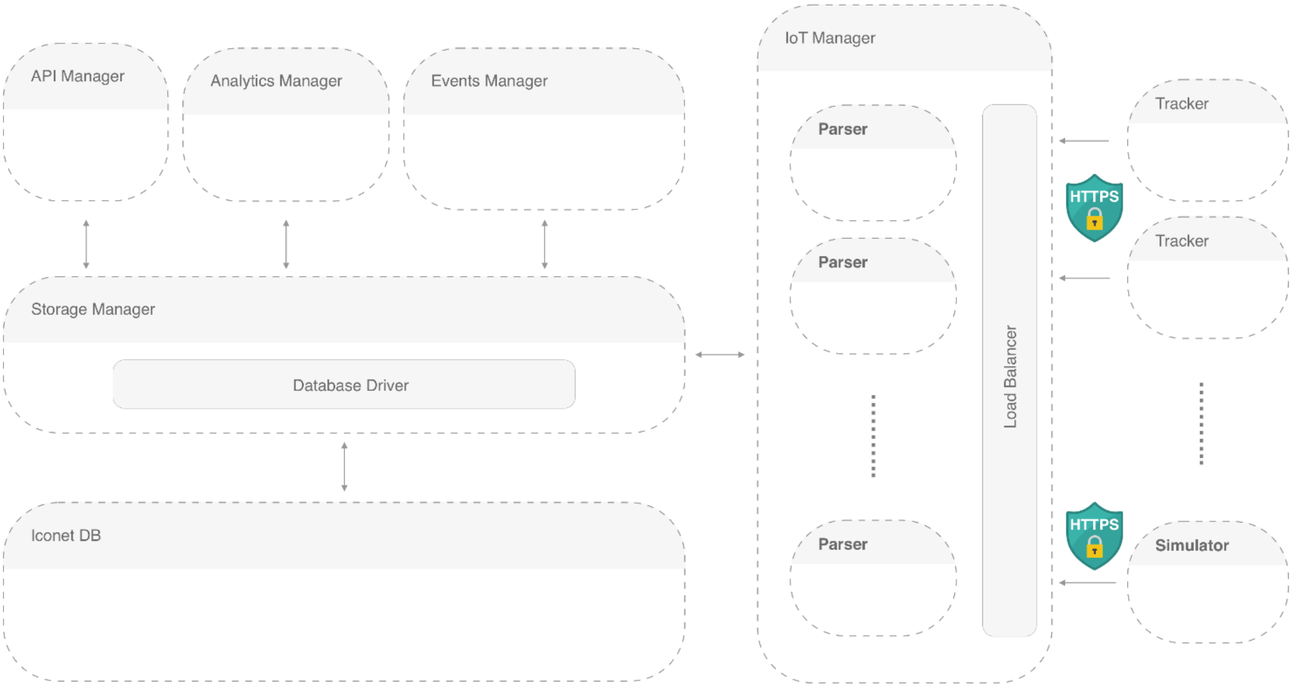


Figure 61 Cloud IoT Platform final architecture

6.3 The IoT manager

This component of the platform, highlighted in Figure 7, is responsible of the collection of the data from the IoT devices deployed within the field. The structure of this component is designed in a cloud friendly manner to be performant and scalable, thus allowing the management of an increasing number of IoT devices exploiting a load balancer module. Its main component is the parser, responsible of validating and standardizing the messages coming from the IoT devices. It is stateless and can be scaled depending on the number of messages it is receiving, cooperating with the load balancer functionalities. It is important to highlight that having separated the parsing functionalities from the Storage manager, processing the data gathered from the IoT devices in 9ms (on average), allowing to handle thousands of requests per second.

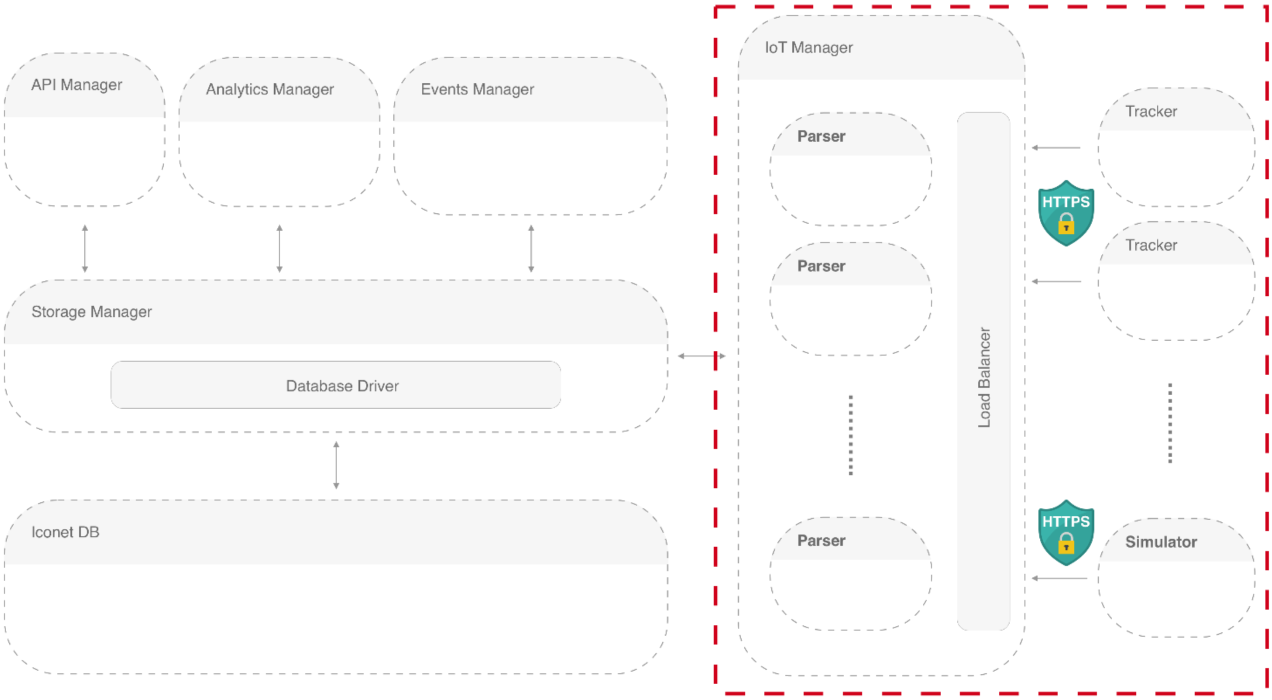


Figure 62 The IoT manager

6.4 The Storage manager

The Storage, depicted in Figure 63, oversees managing the database and it is responsible for the persistence of the data collected from the field, necessary for the analytics tasks as well as for the interaction with the other PI Node's components.

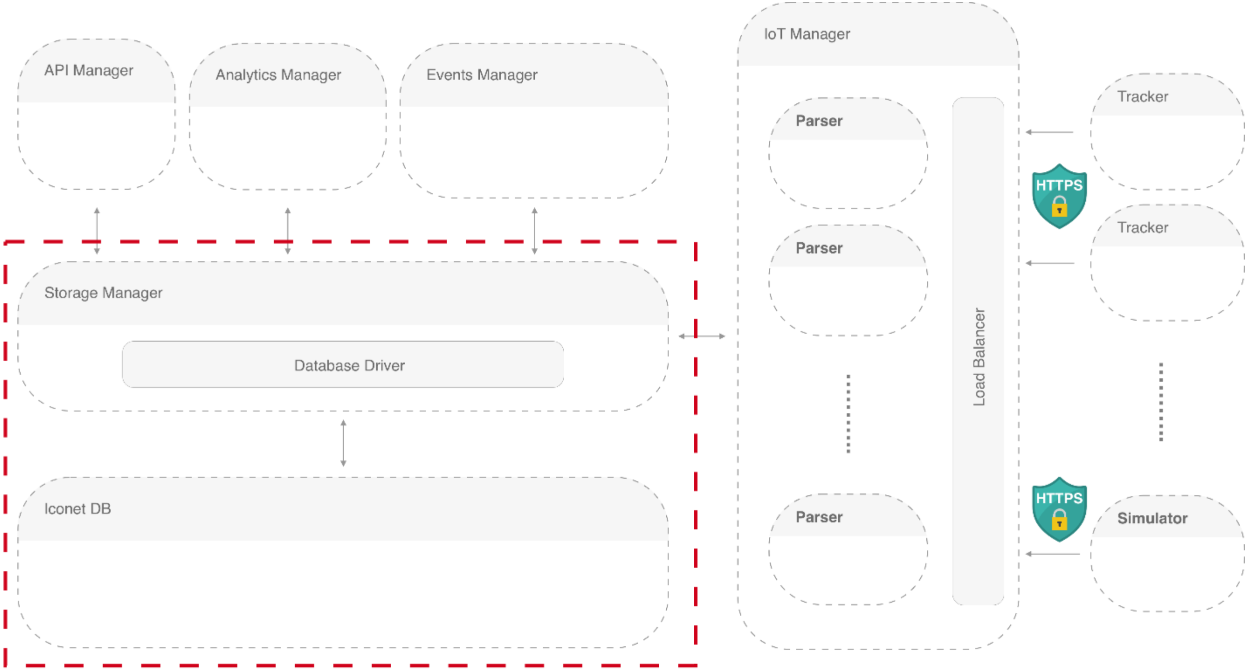


Figure 63 The storage manager

The Storage Manager is designed to interact with the other Cloud IoT Platform components in an abstract way, thus improving the flexibility and the security. Particularly, a set of different access permissions are assigned to interact with it, thus making the system more robust whether some Cloud IoT Platform components misbehave or are under attack.

6.5 Analytics and event manager

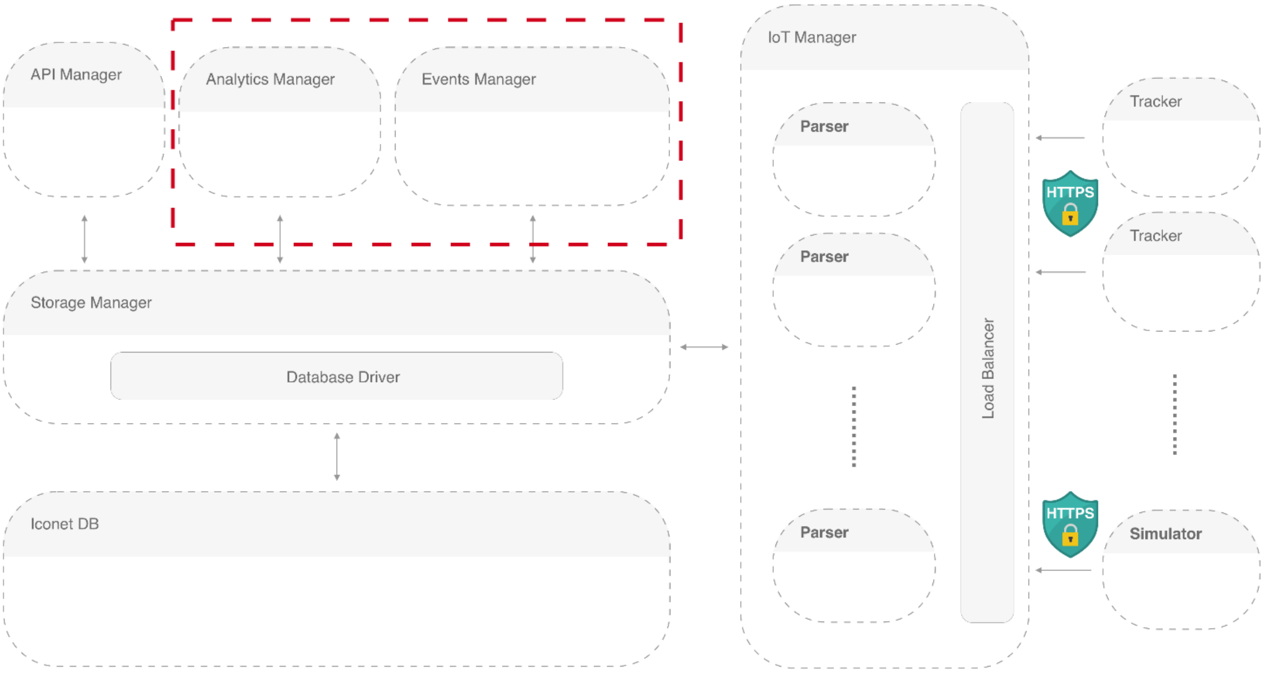


Figure 64 Analytics and Event Manager

The analytics component (depicted in Figure 64) is responsible of implementing several processing on the data gathered from the field or shared during Cloud IoT Service handshakes (i.e. the PI order, see Figure 30). Particularly, it will be in charge of managing the events computed at the edge, as well as computing, at the Cloud level, the events, metadata and statistics listed and described in Sec. 3.4.6 - Generation of event, statistics, and meta-data. In this respect, it oversees implementing two types of analysis:

1. **On-line analysis**. It checks on-line the data gathered during the shipments, processing each sample following the requirements specified in the PI order (as explained in Sec. 5.1.1 - Cloud IoT Service Initialisation service and related data models). The output of this processing is called **events** (for example when a threshold is exceeded). Event can be generated both at the edge side as well as the Cloud side.
2. **Mining analysis**. It aims at extracting useful insight that can be used for enabling data-enabled decision making. These operations are performed off-line and will extract **metadata** (information capable to characterise the single shipment) and **statistics** (aggregated information capable to characterise the supply chain).

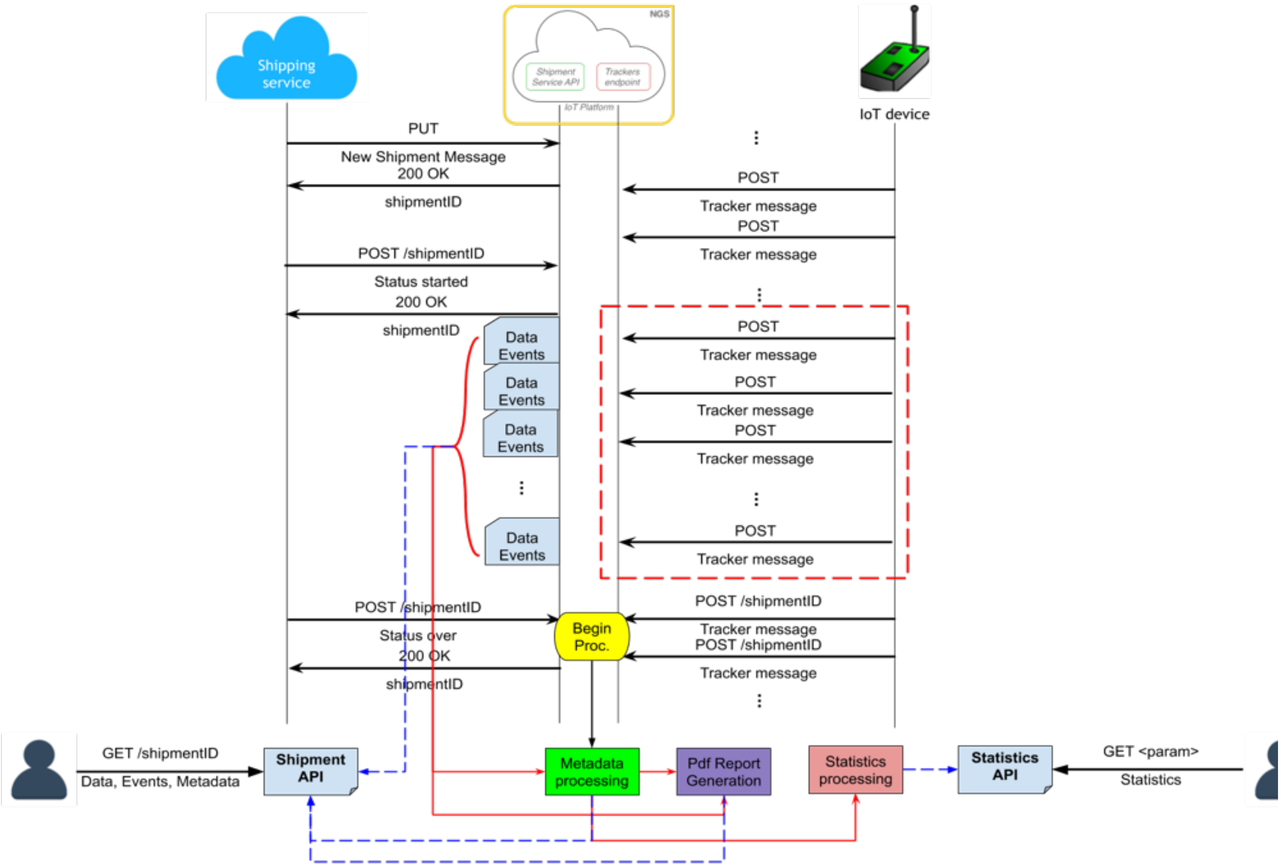


Figure 65 Data Collection - Event, metadata and statistics computation

Figure 65 depicts the complete evolution of the track&trace and monitoring operations and the service made available:

- When the shipment is started, real-time data (gathered by the IoT devices) and events are shared exploiting the "Shipment service API".
- When the shipment operation is terminated, metadata and report are computed and shared exploiting the "Shipment service API". Statistics are updated and shared exploiting the "Statistics API".

Further detail regarding the APIs are provided in the following section.

6.6 APIs manager

The APIs allow the interaction with the platform enabling the sharing of data and information gathered and processed by the platform. Particularly, these will enable several services as the visualisation and reporting, as well as the communication with the shipment services or with authorised third parties' stakeholders. The API manager (see Figure 66) is in charge of managing all the available interfaces to share the data and information with the PI world.

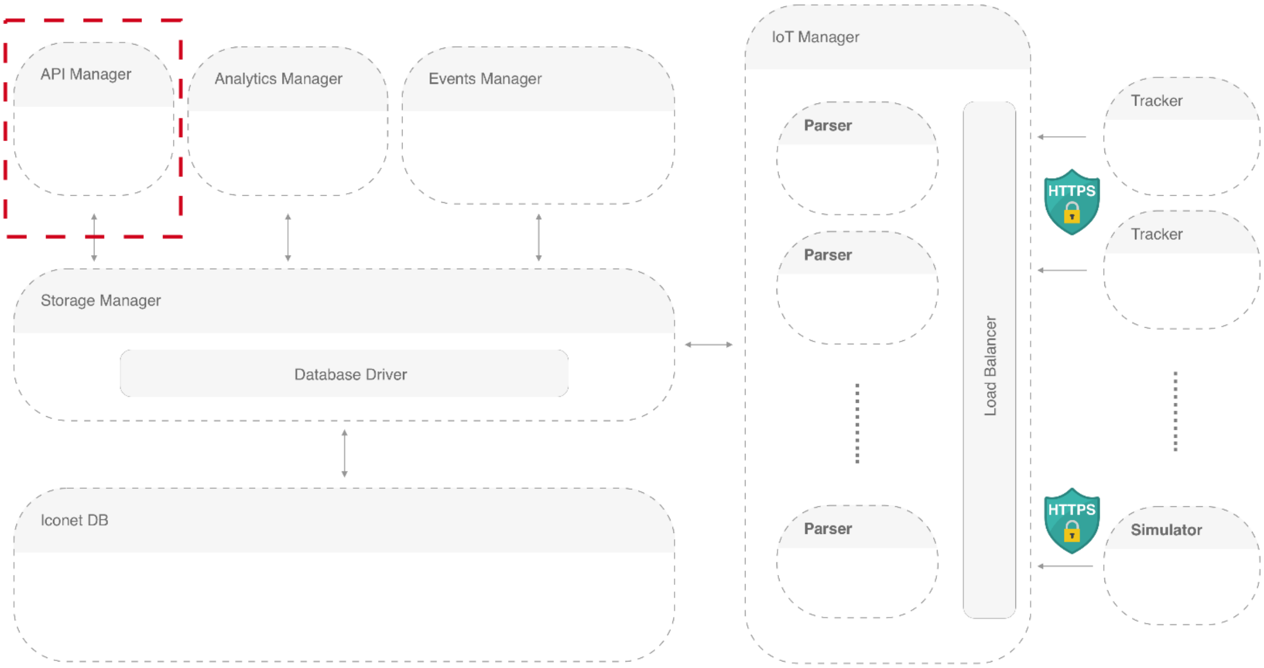


Figure 66 API Manager

It is important to highlight that all the implemented APIs are based on the REST paradigm, considering the protocols HTTPS. All the APIs (and the associated data model) are described using the OpenAPI standard[7]. Particularly the APIs manager has in charge the management of the APIs described hereto:

1. The **Shipment Service APIs**, in charge of sharing related to a certain shipment. For this reason, a certain stakeholder (authenticated with username and password in the Cloud IoT platform) can monitor the considered shipment only if this stakeholder belongs to the group of users related to the shipment itself and identified by the "shipmentID". In this scenario, this stakeholder can retrieve real time data and events from the Shipment Service API, as well as metadata (after the completion of the logistics transaction) passing the correct "shipmentID".
2. The **Statistics API** is in charge of making available certain statistics regarding the performance of the supply chain. Since this data are anonymous, it can be accessed by all the authorised users.

6.6.1 Shipment Service APIs

The **Shipment Service APIs** (New Generation Sensors, s.d.) is the main interface toward the PI world. It is a set of REST APIs that users and the Shipment Services module (see Sec. 6.1) use to manage the shipments (setting up orders, see Sec. 5.1.1 - Cloud IoT Service Initialisation service and related data models) and retrieving the associated data, events and metadata. It can be decomposed in three different subsets of APIs:

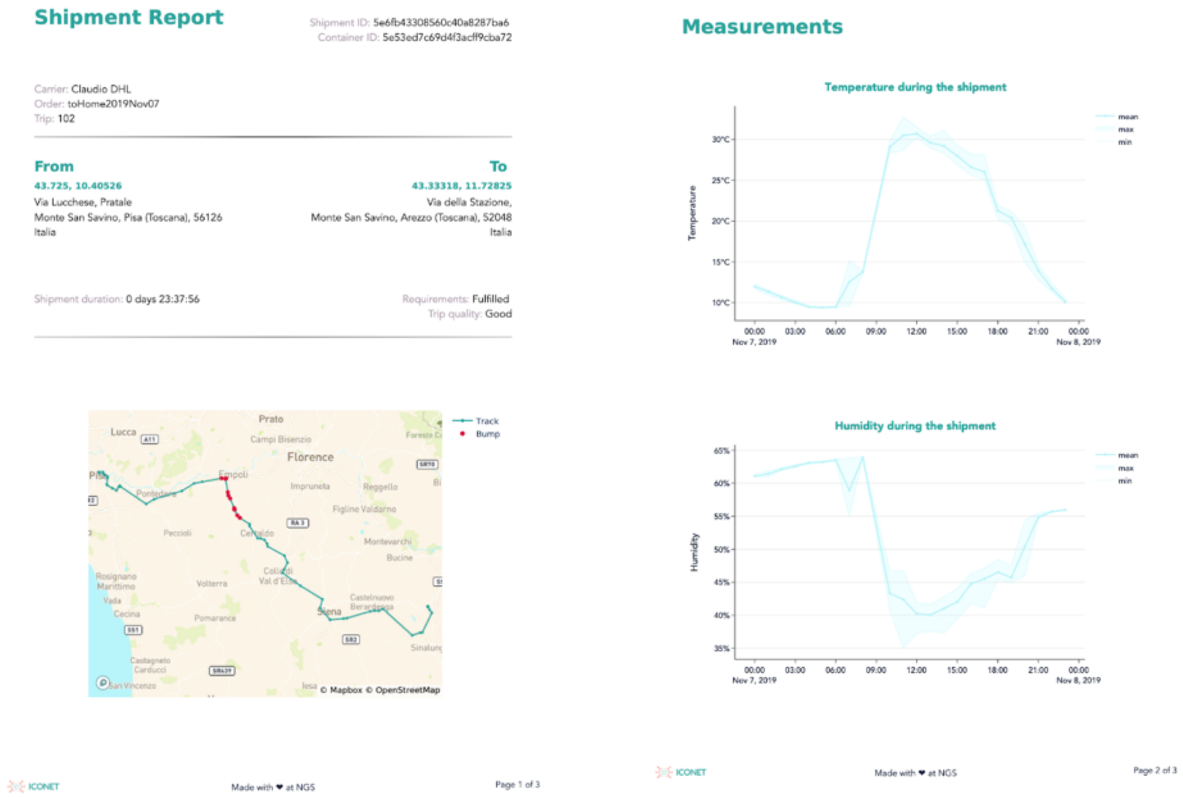
1. **Creation and listing APIs**. A set of APIs in charge of creating a logistics transaction and list the transactions connected with to a certain user (authenticated with username and password).
2. **Show and Modify APIs**, A set of APIs to show the details regarding a shipment and modify its properties (shipment status, i.e., "started" and "over").
3. **Retrieve data APIs**. A set of APIs that allow the Retrieval of the data collected by the IoT device during the shipment.

Further information regarding the discussed APIs are detailed in Table 17, while the complete documentation compliant with the OpenAPI standard is in (New Generation Sensors, s.d.).

[7] The usage of the OpenAPI standard enables the use of all the tools available in its ecosystem from the interactive visualization to the code generation, easing the integration and increasing the interoperability.

Table 17 The shipment service APIs

	Name	HTTPS Meth.	Description
Creation and listing APIs	Add a new Shipment	**PUT**	It allows the creation of the shipment retrieving as parameter the PI order and providing an unique shipmentID.
	Shipments List	**GET**	It provides all the shipment (i.e., shipmentID) related with a user authenticated with username and password.
Show and Modify APIs	Shipment info	**GET**	It retrieves the information regarding a shipment summarising all the rules suggested by the PI order and detailing its status (i.e., “started” and “over”).
	Update shipment status	**POST**	It allows the update of the shipment status (i.e., “started” and “over”).
Retrieve data APIs	Retrieve shipment measurements	**GET**	It allows the collection of the raw data and events collected directly from the IoT devices on the field. It can be collected real-time during the shipment.
	Retrieve shipment events	**GET**	It allows the collection of the list of events happened during the shipment. It can be collected real-time during the shipment.
	Retrieve shipment metadata	**GET**	It allows the collection of the list of metadata automatically computed after the end of the shipment (as depicted in in Figure 65).
	Retrieve shipment report	**GET**	A PDF generator is implemented to generate automatically a report capable to aggregate and visualize the data and information (see Figure 67). As depicted in Figure 65, the pdf is automatically generated at the reception of the shipment end notification (“status over”) and it will be then stored within the Cloud IoT Platform. This API allows the retrieval of this report.



Shipment Report

Shipment ID: 5e6fb43308560c40a8287ba6
Container ID: 5e53ed7c69d4f3acff9cba72

Carrier: Claudio DHL
Order: toHome2019Nov07
Trip: 102

From
43.725, 10.40526
Via Lucchese, Pratale
Monte San Savino, Pisa (Toscana), 56126
Italia

To
43.33318, 11.72825
Via della Stazione,
Monte San Savino, Arezzo (Toscana), 52048
Italia

Shipment duration: 0 days 23:37:56

Requirements: Fulfilled
Trip quality: Good

ICONET   Made with ❤ at NGS   Page 1 of 3

Measurements

Temperature during the shipment

Humidity during the shipment

ICONET   Made with ❤ at NGS   Page 2 of 3

Figure 67 Pdf report sample

6.6.2 Shipment Statistics APIs

The **Shipment Statistics APIs** (New Generation Sensors, s.d.) a set of REST APIs that users and the Shipment Services module (see Sec. 6.1) use to manage the shipments retrieving the statistics regarding the corridors and the containers to extrapolate their performances. The informations made available by these interfaces are the statistics (see definition in Sec. 3.4.6), that represent anonymised data capable to represent the performance of the PI system. For this reason, these are not directly linked with a shipment, but with the PI components: the parameters required by these APIs identify a specific corridor (e.g., passing the origin and the destination) or a container (e.g., passing its ID). As matter of example, it provides features capable to represents:

1. The performance of a selected PI container, evaluating how long it stays unused, thus supporting asset management and optimisation.
2. The performance of a selected PI corridor, providing information regarding the shipment duration and its bottlenecks (e.g., waiting time for intermodal shifts), as well as its quality (e.g., average number of bumps).

In the ICONET project, we have computed four different statistics (see list in Sec. 3.4.6) and for each of them an API is instantiated. Further information regarding the discussed APIs are detailed in Table 18, while the complete documentation compliant with the OpenAPI standard is in (New Generation Sensors, s.d.).

Table 18 The statistics API

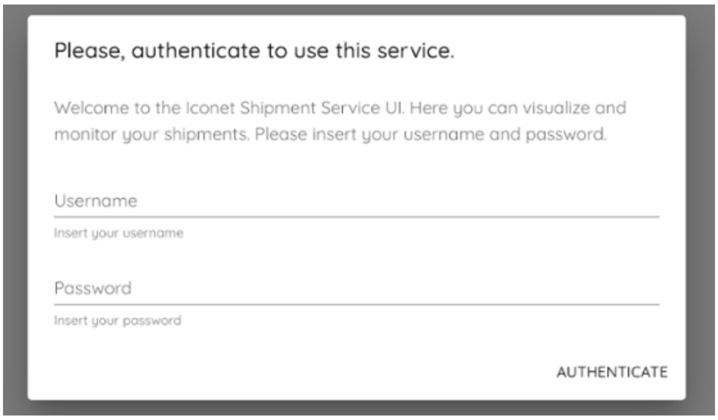
Name	HTTPS Meth.	Description
Average duration	**GET**	It retrieves the average duration of a journey in a certain corridor (defined by the parameters origin and destination).

Average distance	**GET**	It retrieves the average distance of a journey in a certain corridor (defined by the parameters origin and destination).
Time of stop	**GET**	It retrieves how long a container (identified by the Container ID) stay inactive in a certain position
Container efficiency	**GET**	It retrieves the time that a container has spent on travelling in a certain period

6.6.3 The Graphical User Interface (GUI)

The described GUI is a proprietary visualization tool that interacts with the APIs described in Sec. 6.6, allowing the visualisation of the data in a human understandable manner. Below some screenshots that highlight its features regarding:

1. **Authentication** (see Figure 68), where each user can insert its own authorisation credentials (i.e., username and password. Pressing the "AUTHENTICATE" button, the system extracts the list of connected shipment and allows the visualisation of the graphical report.



Please, authenticate to use this service.

Welcome to the Iconet Shipment Service UI. Here you can visualize and monitor your shipments. Please insert your username and password.

Username

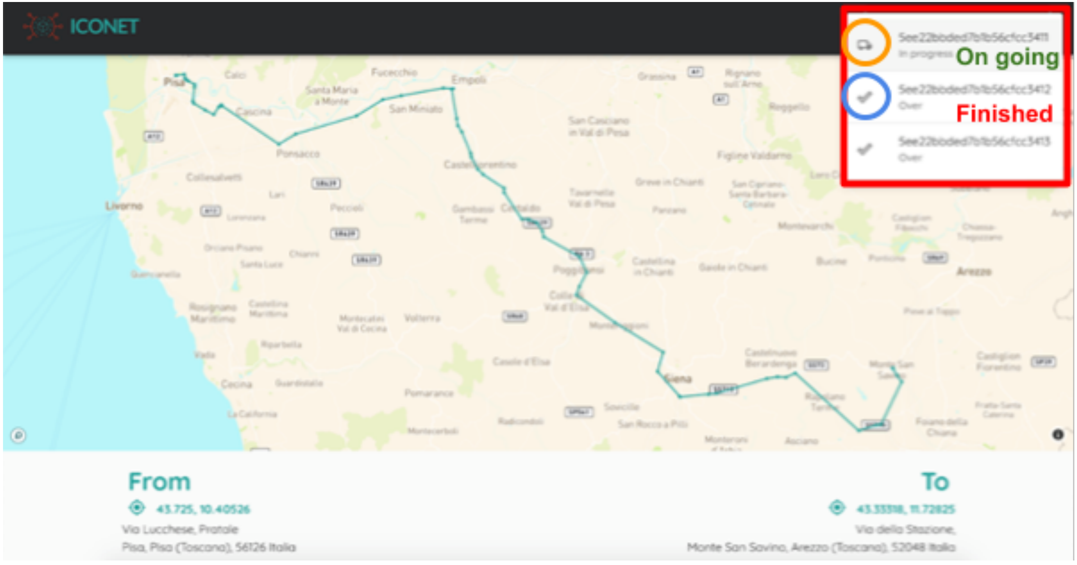
Insert your username

Password

Insert your password

Figure 68 Authentication

2. **Shipment route map and measurement summary** (see Figure 70, Figure 71 and Figure 71). In this page the user can select and visualise all the connected shipment, both on-going and finished (as depicted in Figure 70). After the selection, it can visualise the shipment route map and have an aggregate view of what is happened during the shipment. This report can be used also to monitor the shipment in real-time.



5ee22bbded7b1b56cfcc3411
In progress

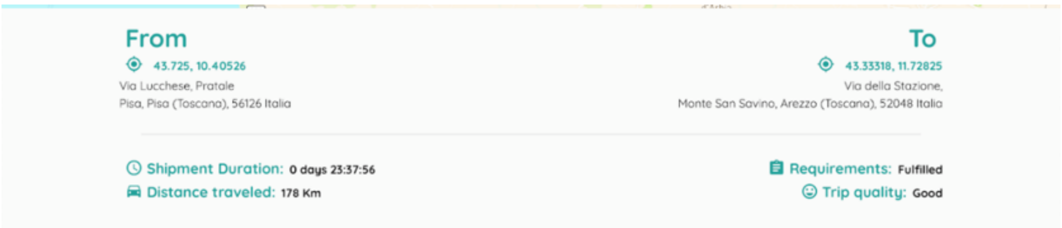
5ee22bbded7b1b56cfcc3412
Over

5ee22bbded7b1b56cfcc3413
Over

From
43.725, 10.40526
Via Lucchese, Pratale
Pisa, Pisa (Toscana), 56126 Italia

To
43.33318, 11.72825
Via della Stazione,
Monte San Savino, Arezzo (Toscana), 52048 Italia

Figure 69 Shipment route map and Shipment selection



From

43.725, 10.40526

Via Lucchese, Pratale
Pisa, Pisa (Toscana), 56126 Italia

To

43.33318, 11.72825

Via della Stazione,
Monte San Savino, Arezzo (Toscana), 52048 Italia

Shipment Duration: 0 days 23:37:56

Distance traveled: 178 Km

Requirements: Fulfilled

Trip quality: Good

Figure 70 Measurement summary

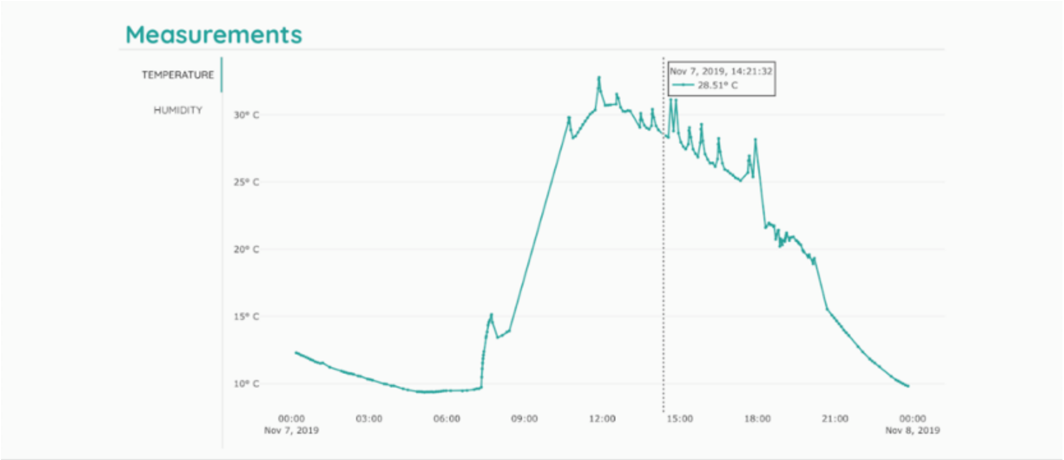


Figure 71 Measurement summary

6.7 Security and ad-hoc access

All the transaction capable to interoperate with the Cloud IoT Platform are implemented in a secure and ad-hoc manner, thus guaranteeing the integrity and the privacy of data. To implement such characteristics two internet standard approaches are considered: “HTTPS” and “Basic Authentication”. For further details see the following sections, Sec. 5.8.1 and 6.7.2 respectively.

Both the approaches are considered to implement a secure environment capable to realise access control at every side of the collection chain. Figure 72 and Figure 73 show the different operative mode foreseen in the ICONET architecture:

1. **Stand-alone configuration** (see Figure 72). In the stand alone approach, IoT devices communicate directly toward the Cloud exploiting the mobile connectivity. In this scenario, the IoT transactions’ security at the premise is managed by the considered cellular protocol, while the transactions between the mobile operator platform is managed using the HTTPS protocol. In the meanwhile, the HTTPS transaction implements the Basic Authentication protocol. In this scenario, each IoT node is equipped with certain immutable username and password. To improve system robustness, we envision as future job the automatic and periodic update of these access keys.
2. **Cooperative configuration** (see Figure 73). In the cooperative approach IoT devices communicate with an IoT gateway using a wireless communication that can implement an encryption protocol (e.g., AES128 in LoRaWAN), that protects the data confidentiality and integrity. At the gateway side, if the connection belongs to the internet world (e.g., Ethernet of Wi-Fi), an HTTPS transaction (enforced with Basic Authentication functionalities) is done, otherwise, if a mobile connectivity is available, the same paradigm described in the Stand-alone configuration is implemented.

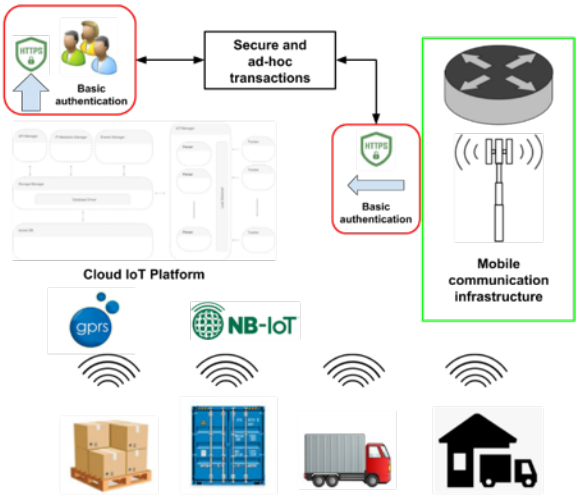


Figure 72 Security and access control in a stand-alone configuration

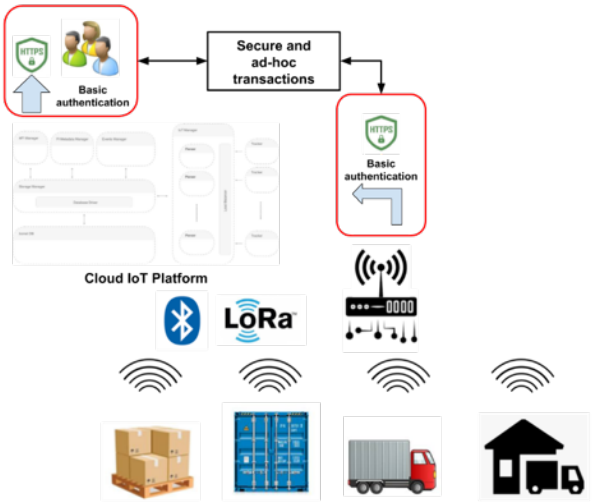


Figure 73 Security and access control in a cooperative configuration

The PI users accesses in the platform, deployed in the Cloud (or in a proprietary server), are completely managed with the REST APIs of Sec. 6.6, implemented as HTTPS interfaced enforced with Basic Authentication functionalities.

6.7.1 HTTPS

HTTPS (or HTTP over TLS) is the secure implementation of HTTP, standardised by IETF (IETF, 2000). In HTTPS, the communication protocol is encrypted using Transport Layer Security (TLS). Servers and clients communicate using HTTP to each other, but over a secure TSL connection that encrypts and decrypts their requests and responses. The TSL layer has 2 main purposes:

1. Verifying the interaction of the client with the correct server.
2. Ensuring that only the server can read what the client send it and only the latter can read what it sends back.

A TSL connection between a client and server is set up by a handshake: once the client-server connection is established, the handshake allows both the parties to agree on the algorithm and keys used to securely send messages to each other.

In this scenario, HTTPS main features are:

1. **Access authentication**. In fact, during the handshake the encryption algorithm, but, mostly, the used keys define a bidirectional and unique relation between client and server.
2. **Protection of the confidentiality and integrity of the exchanged data**. The TSL encrypts the data so these cannot be sniffed and understood by third parties' users. Moreover, data cannot be modified or corrupted during transfer, intentionally or otherwise, without being detected.
3. Consequently, **HTTPS protects the transactions against man-in-the-middle**[8], **eavesdropping** [9]**and tampering**[10] **attacks.**

In conclusion, HTTPS guarantees data **confidentiality** and **clients' authentication**, thus respecting the privacy and security issues of the users (i.e., the computer network clients), as well as **data integrity**, having available effective measurements and parameters for implementing fact-based and data-oriented decision making policies.

[8] Man-in-the-middle attack is an attack where the attacker secretly relays and possibly alters the communications between two parties who believe that they are directly communicating with each other

[9] Eavesdropping is the act of secretly or stealthily listening to the private conversation or communications of others without their consent.

[10] Tampering means a deliberate altering or adulteration of the information.

6.7.2 Basic Authentication

“Basic access authentication” or “Basic authentication” is a method for an HTTP/HTTPS to provide a username and password when making a request, standardized by IETF (IETF, 2015). In basic HTTP authentication, a request contains a header field in the form of Authorization (i.e., Basic <credentials>) where credentials is the Base64[11] encoding of ID and password joined by a single colon, for example “<username>:<password>”. Because base64 is easily decoded, Basic authentication should only be used together with other security mechanisms, as, in our case, in combination with HTTPS.

In the ICONET Cloud IoT platform, The Basic authentication is used to implement the following functionalities:

1. **Authentication**. A user can authenticate with a platform if is registered on it and knows the related password. This step authorises the user to visualise anonymised and general data and information. To access to private data information, the user must be authorised.
2. **Authorisation**. An authenticated user can only access to the information related to its profile (*access control/ad-hoc access*), thus maintaining the data confidentiality and the users’ privacy. In this scenario, a hierarchy of users’ profiles that can access to different set of data (e.g., admin, super-user, user, …). Now, only one type of user is implemented, and this hierarchy will be implemented in the future investigations (e.g., admin, sender/receiver, container owner, logistic operators, etc.).

6.8 The running platform and the deployment on AWS

Because of the confidentiality issues arisen by the considered ITC, the Cloud IoT Platform is deployed within a private server (physical machine) within NGS, uniquely identified by a public IP address. The platform is designed to be natively scalable, thus capable to manage several transactions in parallel exploiting the multi-threading approach. In the contingent scenario, the IoT manager and the APIs manager can manage up to 1000 transaction per seconds on both the interfaces[12].

Moreover, it is implemented in a cloud friendly/cloud native manner considering the Docker Container technologies, simplifying the portability and the scalability boosting.

In this scenario, the Cloud POC deployed within the AWS space (briefly described in Sec. 6.1) interacts with the Cloud IoT platform exploiting the interfaces exposed by the APIs manager (see Sec. 6.6), as depicted in Figure 74. This approach represents a real scenario, since (usually) the PI node platform collect the data from third parties Cloud IoT platform in charge to manage the data gathered from the field.

[11] Base64 is a group of binary-to-text encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation

[12] Experiments done exploiting the open source software called Postman - https://www.postman.com/.

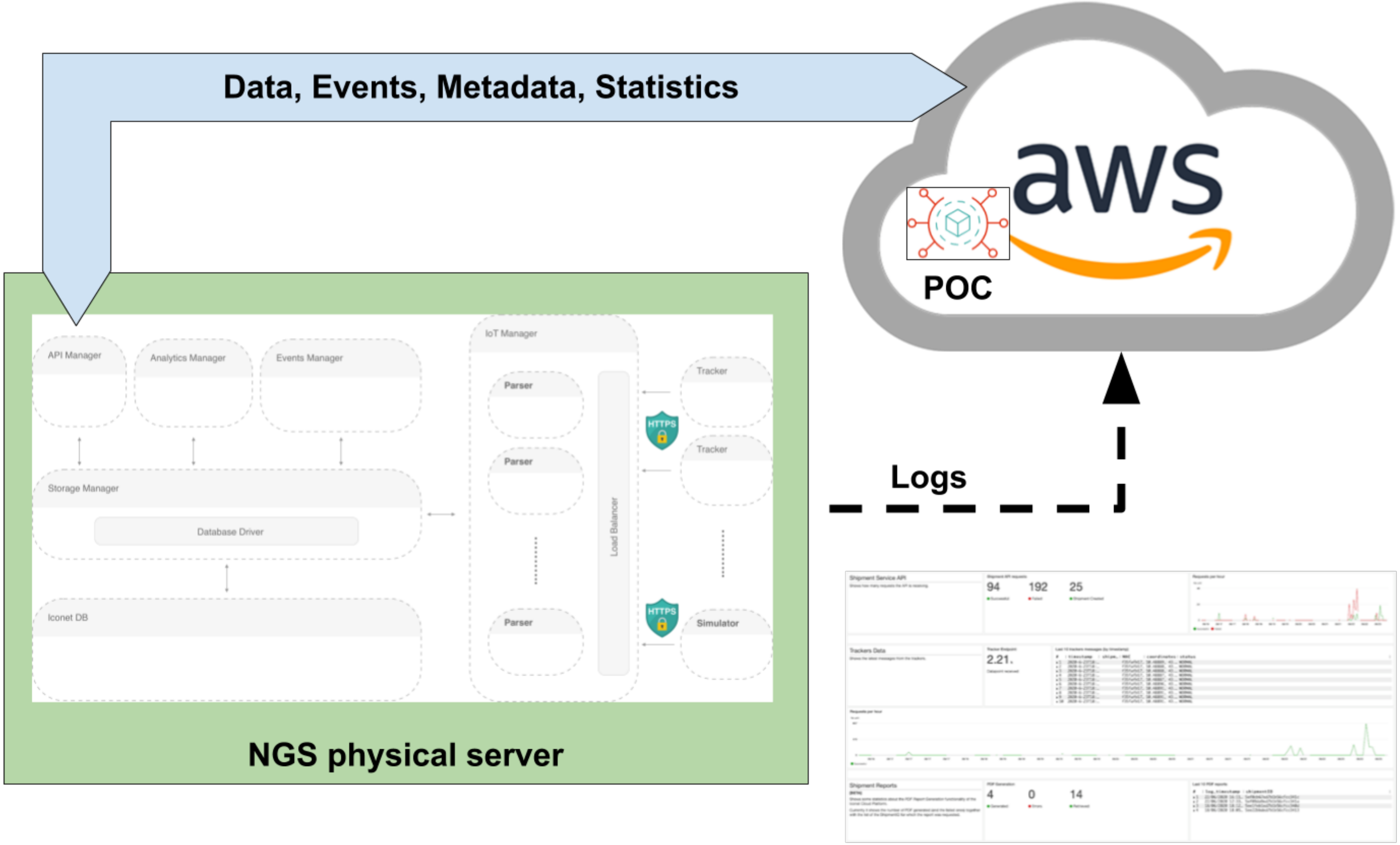
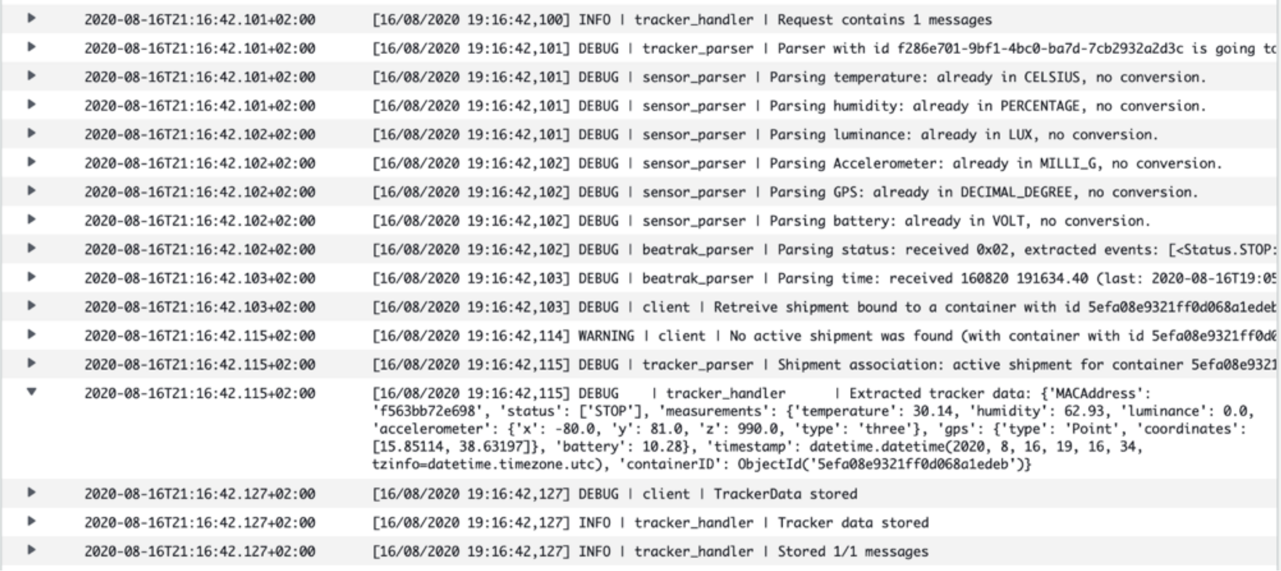


Figure 74 The Cloud IoT Platform deployment

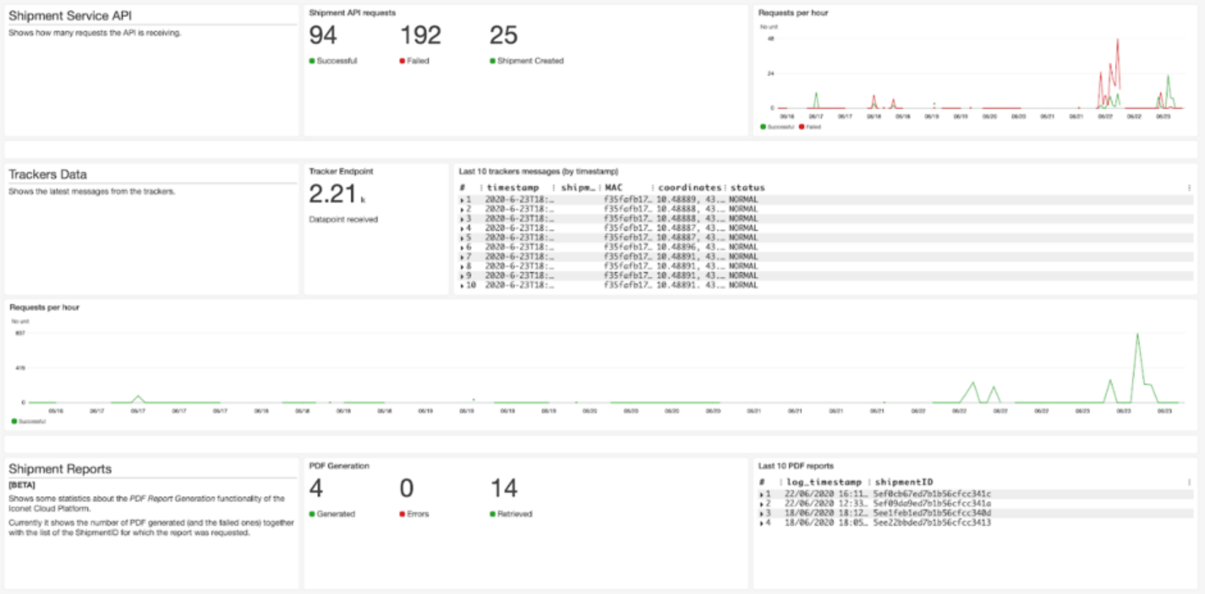
On the other side, the AWS Cloud management functionalities are used to manage and analyse the logs coming from the platform (see Figure 75), allowing the realisation of a platform management system, to

- Monitor the platform transactions (maintaining the confidentiality of the data related to the considered ITC).
- Generate and notify alerts if a system component fails.
- Visualise in a dashboard (depicted in Figure 76) the usage of the platform and its performance.



```
▶  2020-08-16T21:16:42.101+02:00   [16/08/2020 19:16:42,100] INFO | tracker_handler | Request contains 1 messages
▶  2020-08-16T21:16:42.101+02:00   [16/08/2020 19:16:42,101] DEBUG | tracker_parser | Parser with id f286e701-9bf1-4bc0-ba7d-7cb2932a2d3c is going tc
▶  2020-08-16T21:16:42.101+02:00   [16/08/2020 19:16:42,101] DEBUG | sensor_parser | Parsing temperature: already in CELSIUS, no conversion.
▶  2020-08-16T21:16:42.101+02:00   [16/08/2020 19:16:42,101] DEBUG | sensor_parser | Parsing humidity: already in PERCENTAGE, no conversion.
▶  2020-08-16T21:16:42.102+02:00   [16/08/2020 19:16:42,101] DEBUG | sensor_parser | Parsing luminance: already in LUX, no conversion.
▶  2020-08-16T21:16:42.102+02:00   [16/08/2020 19:16:42,102] DEBUG | sensor_parser | Parsing Accelerometer: already in MILLI_G, no conversion.
▶  2020-08-16T21:16:42.102+02:00   [16/08/2020 19:16:42,102] DEBUG | sensor_parser | Parsing GPS: already in DECIMAL_DEGREE, no conversion.
▶  2020-08-16T21:16:42.102+02:00   [16/08/2020 19:16:42,102] DEBUG | sensor_parser | Parsing battery: already in VOLT, no conversion.
▶  2020-08-16T21:16:42.102+02:00   [16/08/2020 19:16:42,102] DEBUG | beatrak_parser | Parsing status: received 0x02, extracted events: [<Status.STOP:
▶  2020-08-16T21:16:42.103+02:00   [16/08/2020 19:16:42,103] DEBUG | beatrak_parser | Parsing time: received 160820 191634.40 (last: 2020-08-16T19:05
▶  2020-08-16T21:16:42.103+02:00   [16/08/2020 19:16:42,103] DEBUG | client | Retreive shipment bound to a container with id 5efa08e9321ff0d068a1edet
▶  2020-08-16T21:16:42.115+02:00   [16/08/2020 19:16:42,114] WARNING | client | No active shipment was found (with container with id 5efa08e9321ff0d0
▶  2020-08-16T21:16:42.115+02:00   [16/08/2020 19:16:42,115] DEBUG | tracker_parser | Shipment association: active shipment for container 5efa08e9321
▼  2020-08-16T21:16:42.115+02:00   [16/08/2020 19:16:42,115] DEBUG    | tracker_handler      | Extracted tracker data: {'MACAddress':
                                   'f563bb72e698', 'status': ['STOP'], 'measurements': {'temperature': 30.14, 'humidity': 62.93, 'luminance': 0.0,
                                   'accelerometer': {'x': -80.0, 'y': 81.0, 'z': 990.0, 'type': 'three'}, 'gps': {'type': 'Point', 'coordinates':
                                   [15.85114, 38.63197]}, 'battery': 10.28}, 'timestamp': datetime.datetime(2020, 8, 16, 19, 16, 34,
                                   tzinfo=datetime.timezone.utc), 'containerID': ObjectId('5efa08e9321ff0d068a1edeb')}
▶  2020-08-16T21:16:42.127+02:00   [16/08/2020 19:16:42,127] DEBUG | client | TrackerData stored
▶  2020-08-16T21:16:42.127+02:00   [16/08/2020 19:16:42,127] INFO | tracker_handler | Tracker data stored
▶  2020-08-16T21:16:42.127+02:00   [16/08/2020 19:16:42,127] INFO | tracker_handler | Stored 1/1 messages
```

Figure 75 Logs regarding data acquisition



Shipment Service API

Shows how many requests the API is receiving.

Shipment API requests

94 ● Successful

192 ● Failed

25 ● Shipment Created

Trackers Data

Shows the latest messages from the trackers.

Tracker Endpoint

2.21 k

Datapoint received

Last 10 trackers messages (by timestamp)

#	timestamp	shipm…	MAC	coordinates	status
▸1	2020-6-23T18:…		f35fafb17…	10.48889, 43.…	NORMAL
▸2	2020-6-23T18:…		f35fafb17…	10.48888, 43.…	NORMAL
▸3	2020-6-23T18:…		f35fafb17…	10.48888, 43.…	NORMAL
▸4	2020-6-23T18:…		f35fafb17…	10.48887, 43.…	NORMAL
▸5	2020-6-23T18:…		f35fafb17…	10.48887, 43.…	NORMAL
▸6	2020-6-23T18:…		f35fafb17…	10.48896, 43.…	NORMAL
▸7	2020-6-23T18:…		f35fafb17…	10.48891, 43.…	NORMAL
▸8	2020-6-23T18:…		f35fafb17…	10.48891, 43.…	NORMAL
▸9	2020-6-23T18:…		f35fafb17…	10.48891, 43.…	NORMAL
▸10	2020-6-23T18:…		f35fafb17…	10.48891. 43.…	NORMAL

Shipment Reports

[BETA]

Shows some statistics about the *PDF Report Generation* functionality of the Iconet Cloud Platform.

Currently it shows the number of PDF generated (and the failed ones) together with the list of the ShipmentID for which the report was requested.

PDF Generation

4 ● Generated

0 ● Errors

14 ● Retrieved

Last 10 PDF reports

#	log_timestamp	shipmentID
▸1	22/06/2020 16:11…	5ef0cb67ed7b1b56cfcc341c
▸2	22/06/2020 12:33…	5ef09da9ed7b1b56cfcc341a
▸3	18/06/2020 18:12…	5ee1feb1ed7b1b56cfcc340d
▸4	18/06/2020 18:05…	5ee22bbded7b1b56cfcc3413

Figure 76 Logs analysis dashboard

7. The Tracker and the Smart Router

Trackers and Smart Routers are two essential devices to ensure continuous tracking and monitoring service of the Smart PI Containers (see Sec. 3.5.1). They are built on a common hardware platform, that can be configured from the remote platform to satisfy the needs of the shipment (described in the PI order, see Sec. 5.3). The functionalities of the Trackers and the Smart Routers are summarised in Table 19, while the interaction model of the Tracker/Smart Router is depicted in Figure 36.

Table 19 Tracker vs Smart Router

Functionalities	Smart Router	Tracker
Collect information regarding position and time of a container	Yes	Yes
Collect added value measurement exploiting on board sensors	Yes	Yes
Monitor the presence of connected goods encapsulated within it (e.g., Smart PI Pallets)	Yes	No
Collect added value environmental data inside/outside the container, exploiting short range IoT protocols	Yes	No
Communicate remotely exploiting different protocols the data collected	Yes	Yes
Provide the possibility to receive configuration/actuation messages from remote	Yes	Yes
Implementing data logging functionalities in case of missing collection	Yes	Yes
Low power consumption, battery powered, power saving policies	Yes	Yes

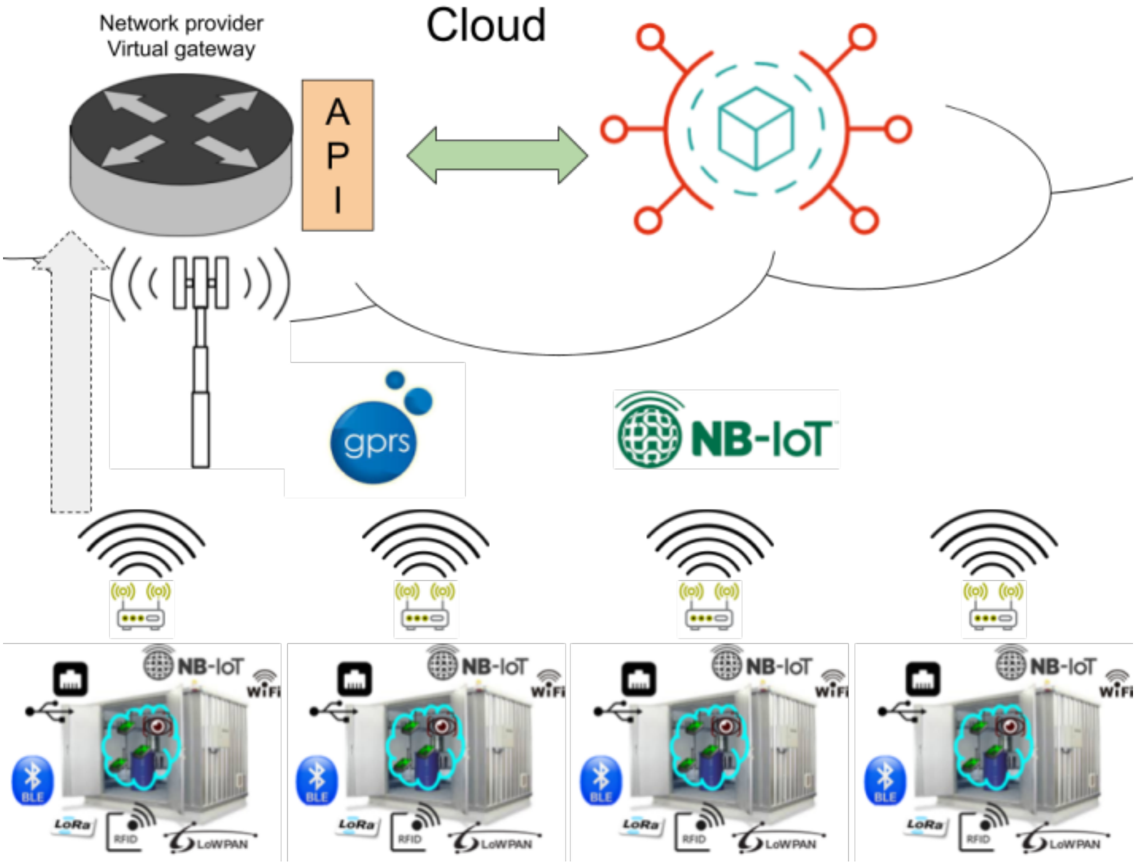


Figure 36 Tracker/Smart Router interaction model

7.1 The ICONET Smart router characteristics

The ICONET Smart Router represents the key components in charge of:

1. Integrating added value on-board sensors (i.e., temperature-humidity, acceleration, and light) capable to generate asynchronous events such as bump detection, movement detection, temperature thresholds exceeding and door open/close detection.
2. Implementing an internal and external container connectivity, thus allowing the management of the encapsulation of the connected goods as well as their (distributed) monitoring, allowing the integration of a scalable set of IoT devices capable of providing added value measurements, and the monitoring of the container itself (e.g., connected seals status).
3. Tagging the data and information gathered from both the IoT devices and the on-board sensors with their geo&time-reference, allowing to understand the place and the time where certain events have happened, as well as the implementation of the track&trace service.
4. Implementing an improved interoperable connectivity, integrating five different protocols in the same board (i.e., GPRS, NB-IoT, LTE Cat-M1, BLE, IEEE802.15.4). Particularly, the solution is 5G ready, and compliant with the guidelines of DSCA standard release 1 and with the CALM ITS station. For further details, see Sec. 7.4.
5. Designed to realise the features of the ICONET IoT architecture.
6. Energy saving design, though to be battery powered and enforced by an (solar) energy harvesting functionality, thus improving its integrability and reducing its maintenance.

7.2 The ICONET Smart Router Reference Implementation

Both the Tracker and the Smart Router are based on the full-fledged release of the Micro-FLEXX board (see Figure 77). On the other hand, Figure 78 shows the final FSM to implement the Smart Router and the Tracker (without considering the state highlighted in green).

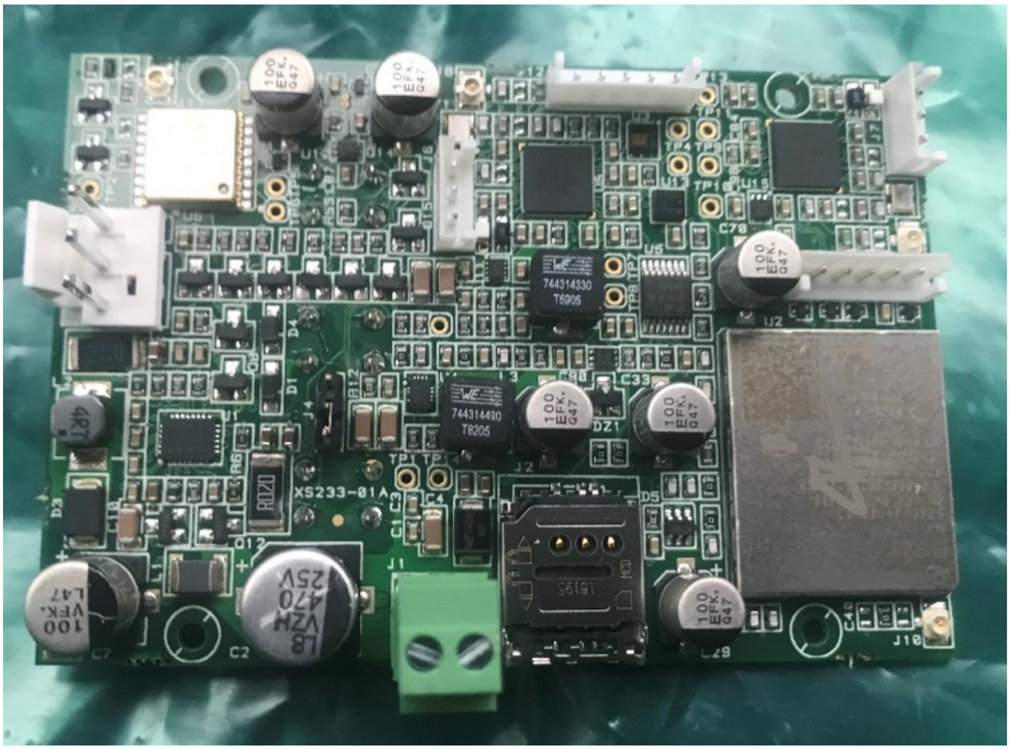


Figure 77 The Micro-Flexx board

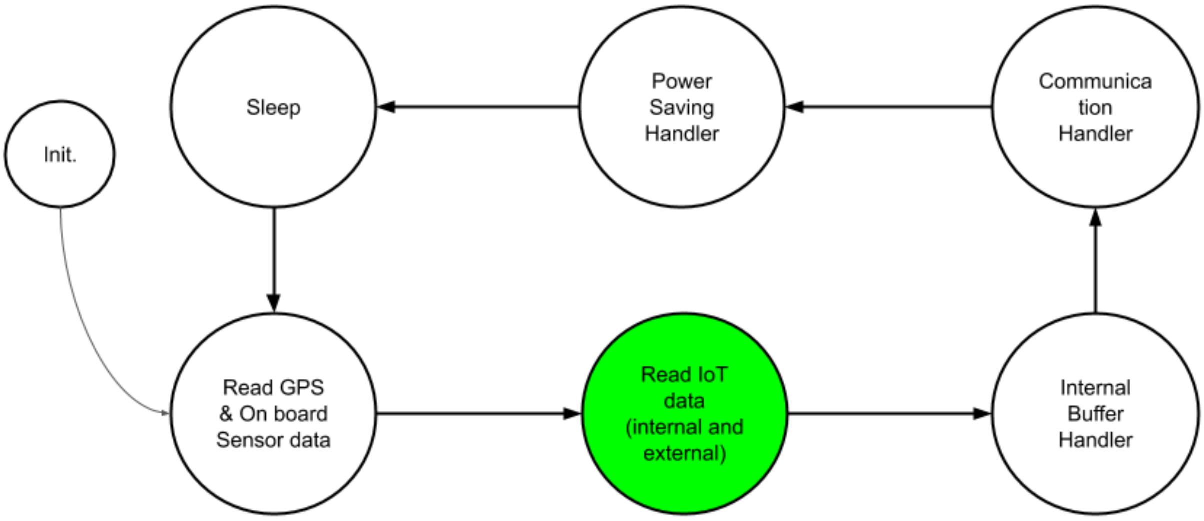


Figure 78 Tracker/Smart Router FSM

7.3 Micro-FLEXX details and performance

The reasons of the development of the Micro-FLEXX board in Figure 77 derives from the impossibility to exploit already existing programmable platforms (IoT tracker companies do not provide the possibility to customise their hardware) since not optimised for the purposes of the ICONET project. The one we have considered at the beginning of the project (RAK, s.d.) was characterised in D2.6 and has the following limitations:

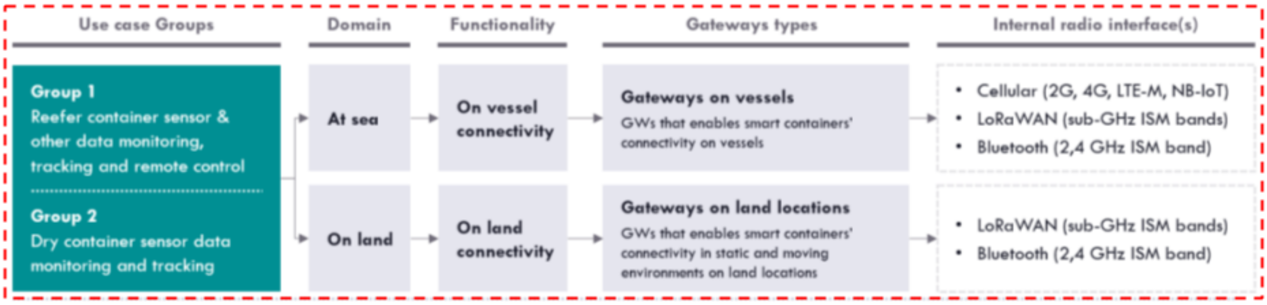
1. The main limitation regards the not optimised power supply sub-component. Particularly, the GPS supply circuit drowns current continuously, also during the sleep period. In the following table the comparison of the power consumption, coming from laboratory experiments, is shown, demonstrating a consumption optimisation of almost the 20%.

Table 20 Power consumption comparison

Board	**Battery capacity (Wh)**	**Duration (Days)**	**Consumption per day (Wh/days)**
RAK	18.5	16	1
Micro FLEXX	24	30	0,8

2. Lack of network interfaces. The proposed solution enables only one BLE network interface, as well as NB-IoT/GPRS.

7.4 Standardisation considerations


Use case Groups
Domain
Functionality
Gateways types
Internal radio interface(s)
Group 1
Reefer container sensor & other data monitoring, tracking and remote control
Group 2
Dry container sensor data monitoring and tracking
At sea
On land
On vessel connectivity
On land connectivity
Gateways on vessels
GWs that enables smart containers' connectivity on vessels
Gateways on land locations
GWs that enables smart containers' connectivity in static and moving environments on land locations
• Cellular (2G, 4G, LTE-M, NB-IoT)
• LoRaWAN (sub-GHz ISM bands)
• Bluetooth (2,4 GHz ISM band)
• LoRaWAN (sub-GHz ISM bands)
• Bluetooth (2,4 GHz ISM band)

Referencing to the specification of DCSA standard release 1 (Digital Container Shipping Association (DCSA), 2020), the gateway capable to interact with the Smart PI containers - in the contingent case, with the Smart Routers – can implement a sub-set of protocols between cellular (2G, 4G, LTE-M, NB-IoT), LoRaWAN, Bluetooth. In this scenario, the Smart Router is compliant with DCSA standard, installing the following four protocols: 2G, LTE-M, NB-IoT and Bluetooth.

Moreover, it is compliant with ISO ITS stack (ISO/TC204, s.d.) integrating also IEEE.802.15.4, thus allowing the integration of the 6LoWPAN (ISO, 2016) and CoAP (ISO, 2016) facilities. Finally, it is 5G ready since it integrated both the forerunner protocols NB-IoT and LTE Cat-M.

8. The Smart Gateway

The Smart Gateway is the edge computer capable of managing different IoT protocols and collecting the data coming from distributed sensors or from the tracker/smart routers associated with containers, as depicted in Figure 38.

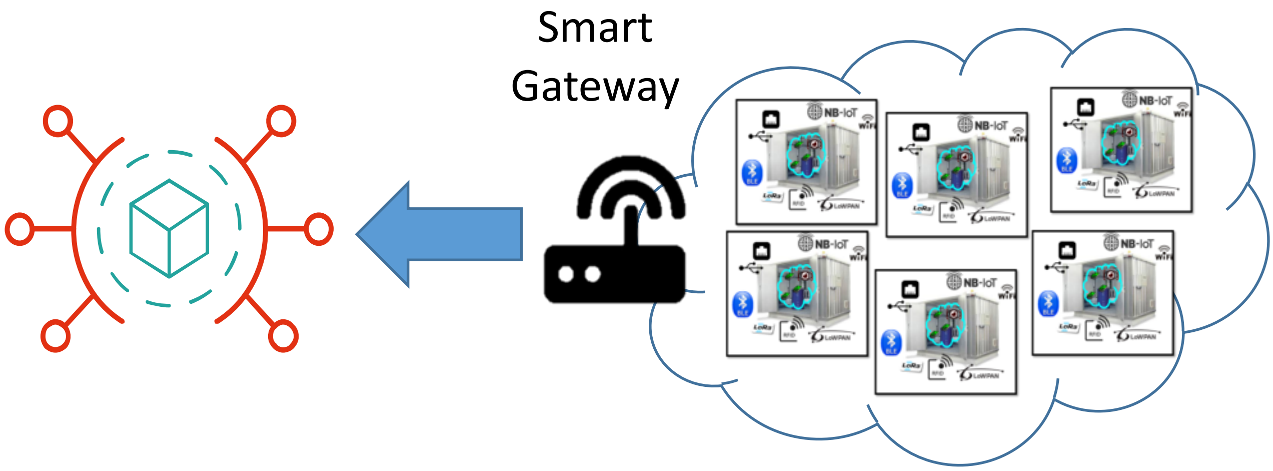
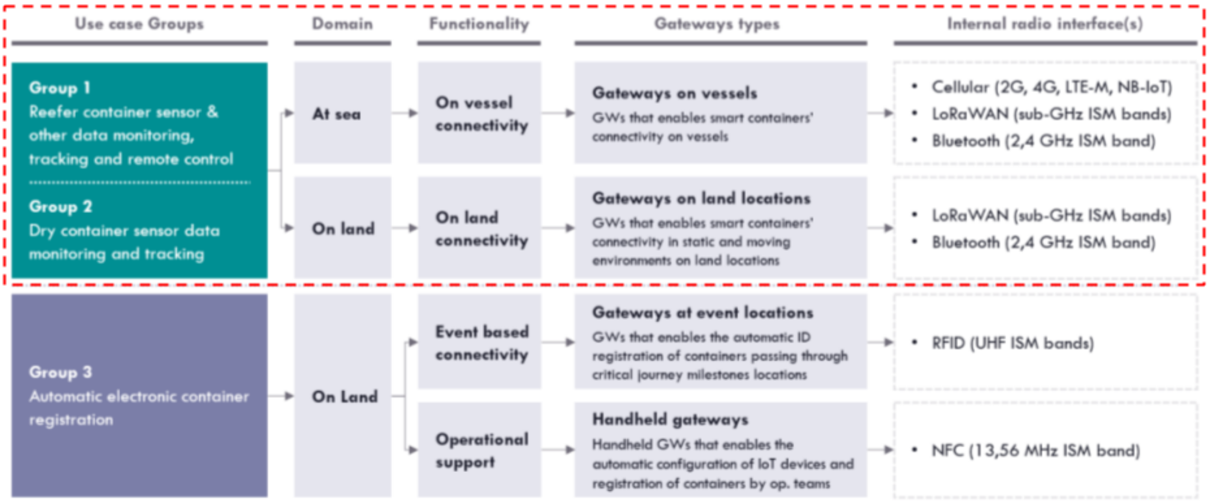


Figure 79 FLEXX gateway interaction model

Referencing at the DSCA reference architecture (see Sec. 3.3.1 - DCSA standard for digital containers) this device behaves logically as the gateway located on the vessels, on the trucks, on the train and on the barge: it enables the on-board IoT connectivity, thus, in the PI vision, providing information regarding the encapsulation of the container and of the goods.

According to DCSA standard guidelines (Digital Container Shipping Association (DCSA), 2020), the smart gateway characteristics are in-line with the requirements for realising the first two proposes, related on tracking and monitoring both reefer and dry container (see Figure 80): the smart gateway could be used for providing IoT connectivity to smart containers on vessels and on land, supporting two over three protocols (i.e., LoRaWAN and Bluetooth) listed in the aforementioned standards, as shown in Figure 80. Thus, the proposed smart gateway ensures the technical interoperability for logistics application for two group use cases by satisfying this standard.



Use case Groups	Domain	Functionality	Gateways types	Internal radio interface(s)
Group 1 Reefer container sensor & other data monitoring, tracking and remote control Group 2 Dry container sensor data monitoring and tracking	At sea	On vessel connectivity	Gateways on vessels GWs that enables smart containers' connectivity on vessels	• Cellular (2G, 4G, LTE-M, NB-IoT) • LoRaWAN (sub-GHz ISM bands) • Bluetooth (2,4 GHz ISM band)
	On land	On land connectivity	Gateways on land locations GWs that enables smart containers' connectivity in static and moving environments on land locations	• LoRaWAN (sub-GHz ISM bands) • Bluetooth (2,4 GHz ISM band)
Group 3 Automatic electronic container registration	On Land	Event based connectivity	Gateways at event locations GWs that enables the automatic ID registration of containers passing through critical journey milestones locations	• RFID (UHF ISM bands)
		Operational support	Handheld gateways Handheld GWs that enables the automatic configuration of IoT devices and registration of containers by op. teams	• NFC (13,56 MHz ISM band)

Figure 80 DSCA IoT protocols

As discussed in both D2.6 and D2.7, its software architecture is inspired by the outcomes of the Horizon 2020 project called AGILE-IoT (3), referencing on the open source library thought for the realisation of modular and configurable gateways. In the following sections, the implementation details for the realisation of the smart gateway are detailed.

8.1 The smart gateway implementation

In this section we describe the architecture of the generic framework that NGS is developing to manage the configuration of the Smart Gateway service. In Figure 81, the architecture of the framework is described, following the guidelines of the AGILE-IOT project. Particularly, the system contains three main services:

- **IoT Protocol Service:** capable to manage both: (i) the data acquisition from the IoT sensor nodes efficiently acquire sensor data, and (ii) the IoT sensor nodes configuration. This service includes subservices for different IoT protocols such as BLE, LoRa, etc.
- **Data Storage Service:** capable to manage a local database to store the data. As matter of example, this service can be configured to save in case the remote connectivity is not available.
- **Communication Service:** this service can enable the data exchange with remote platforms. The services' API will be implemented following the interoperability guidelines coming from D1.6 and D2.7.
- **Application** that configures the gateway to implement certain behaviours. The mentioned application is realised exploiting the web-oriented scripting approach and it is detailed in Sec. 8.1.1.

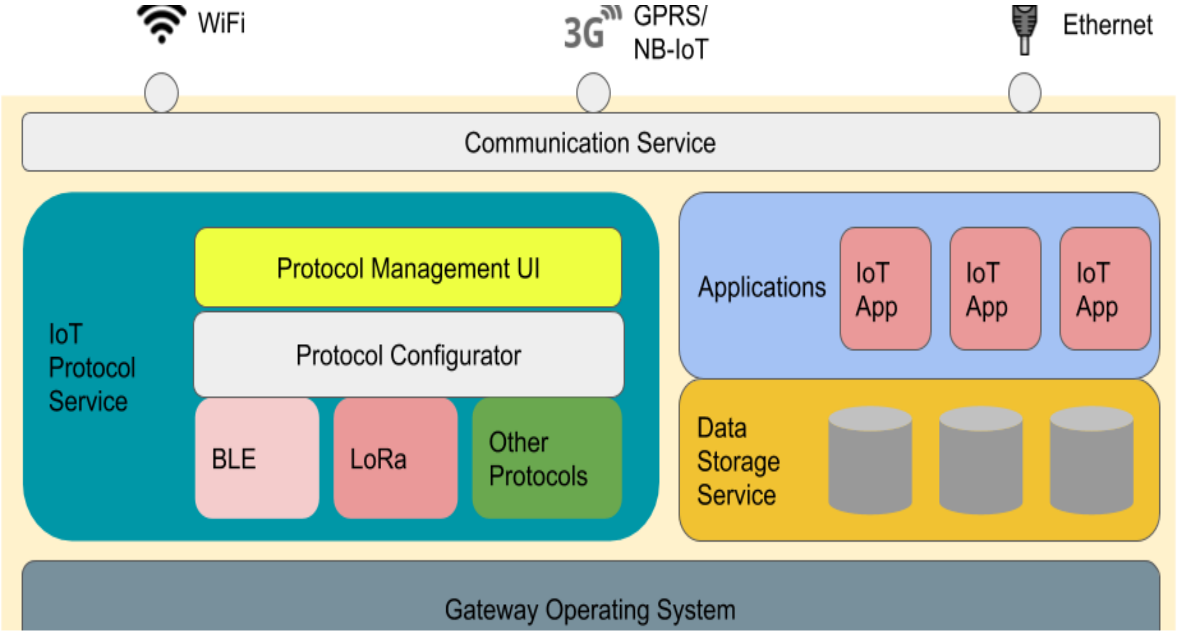


Figure 81 Smart Gateway Reference Architecture

8.1.1 The Smart Gateway Application

Starting from the framework architecture of Figure 81, we have configured the Smart Gateway application capable of managing different protocol services (e.g., BLE, LoRa, CoAP). It acquires meaningful data from the IoT environment and from the on-board GPS module, then construct the messages to satisfy the defined in D2.7 data-models. In case there is no connectivity to the Cloud IoT Platform, these messages are transferred to the Storage Service. Otherwise, the data can be transmitted directly to the Cloud IoT Platform, managing a Multiprotocol remote communication.

8.1.2 The Smart Gateway Board

FLEXX gateway v.2.0 board is the base of the Smart Gateway. Its implementation was completed in Jul. 2019, see Figure 82, and embeds the following functionalities:

- ARM Cortex-A8 @ 1 GHz, 4 GB eMMC memory, 512 MB of RAM
- SIM card
- GPS/GLONASS sensor
- IoT wireless interfaces v.1.0: LoRa Concentrator, BLE
- Remote connection with Wi-Fi, Ethernet, NB-IoT (5G)/GPRS transceiver
- Up to 24VDC power supply

.

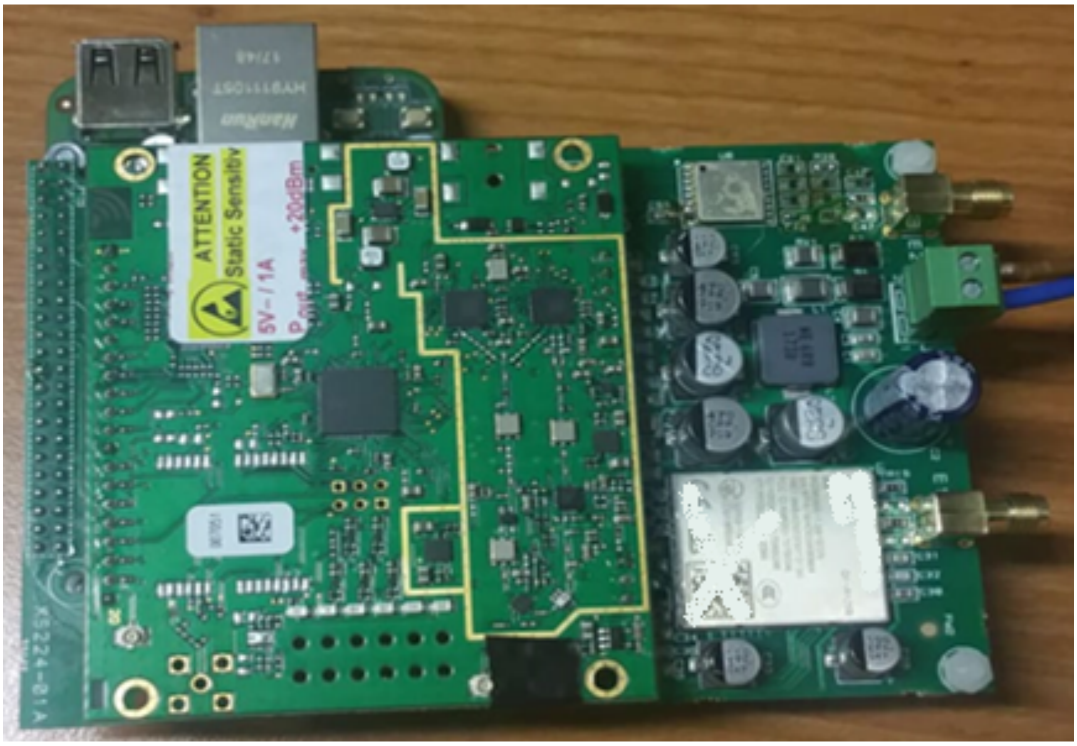


Figure 82 Smart Gateway hardware

9. Conclusions

The ICONET project aims to implement the first prototype of PI to realise efficient and interoperable logistics transactions, thus reducing their cost and their impact on the environment. This report is the direct consequence of D1.6, where the specifications and the general IoT architecture are defined and it concludes the series of three deliverables to report the technical advances regarding the realisation of an IoT environment for the PI: D2.6 and D2.7.

Particularly, the impact of the visionary and innovative IoT architecture, capable to interact with multi-domain networks (logistics, ITS, smart city – being compliant with well-recognised standards) supports the ICONET environment (so the envisioned PI environment), thus allowing seamless and cost-efficient implementation of end-to-end visibility toward the improvement of the efficiency and the proactiveness of the logistics network.

The integration of the mentioned visionary architecture with the Cloud IoT Platform has allowed the realisation of innovative approaches, capable to support the realisation of a smart and proactive PI environments. In fact, analysing the findings highlighted in this report, the innovations generated by the proposed approach can be classified in 3 categories, following the different level of abstraction of the IoT environment:

1. The "Premise layer", where the realisation of the IoT architecture depicted in Figure 2 enables improvement in pervasivity and the interoperability. In this scenario, data exchange and collection with/from other linked domains (i.e., ITS), and an improved granularity in the goods and assets tracking and monitoring are enabled. Such approach gives to PI the capability to monitor, manage and optimise the whole supply chain at different levels, proposing a comprehensive environment capable to monitor goods, encapsulations, assets, means and infrastructure, and transforming these in added value Physical Internet components.
2. The "Analytics layer" aims at implementing in depth analysis of the shipments in terms of efficiency and quality. The analytics layer provides anonymised information derived from the aggregation of the shipments data and features. The output of this analytics component represents the characterisation of the PI network and represents an important support for the PI in the decision making and optimisation paths. In fact, it suggests the main criticalities on the supply chain, regarding the shipments' duration and quality, supporting the PI actions toward their optimisation.
3. The "PI Integration layer" represent the innovative functionalities and services enabled by the smart integration with the higher level PI components. In this scenario, the real data collected and processed by the Cloud IoT Platform (i.e., events, meta-data and statistics) are the main support on the proactive and real time decision making. Proactive routing and assets' visibility and management improvements are enabled by this integration, giving to PI the possibility to have a continuous, complete, and detailed vision of the supply chain. In this manner, PI can improve the optimisation activities intervening directly and continuously on the supply chain transaction, toward their optimisation in term of cost and environmental impacts.

D2.8 aims at reporting the final release of the technical components to satisfy the business requirements (summarised in Sec. 3.1) and to implement the ICONET IoT architecture (see Figure 2), both defined in D1.6. Moreover, it reports how the open, distributed, and interoperable IoT environment supports the realisation of the PI network, seen as a set of independent and distributed PI nodes that interoperate to organise autonomously the logistics network toward its optimisation. In fact, it suggests a set of direction to follow to realise the supply chain complete visibility claim, seen as:

1. An **interoperable IoT environment** where different parties can cooperate implementing different business models. In this scenario, the standardisation framework is analysed, and the ICONET approaches, architecture and the prototypes are positioned within it.
2. An **improved monitoring environment** where goods and their encapsulation are tracked at different granularity. Smart PI Container, Smart PI Pallet and Smart PI packet are defined to monitor the goods and track their encapsulation, as well as monitoring the assets itself. For further detail see Sec. 3.5.1 and 3.5.2.

3. **A scalable and specialised set of monitoring functionalities**. The ICONET IoT architecture allows the realisation of an improved monitoring granularity, enabling the realisation of special and dedicated functionalities on each Smart PI Pallet. This feature allows the implementation of a scalable set of interoperable special devices that represent a scalable set of special functionalities. For further detail see Sec. 3.5.1 and 3.5.2.
4. **Accessible and comprehensible Cloud IoT platform**, in charge of making available the collected data and the processed information to all the user involved in the PI transaction. A secure and ad-hoc access paradigm is implemented to protect the data flows toward the right owners. Big-data analytics is implemented to support the fact-based and data-oriented decision making. For further detail see Sec. 5.4.

This deliverable concludes the set of three advance report regarding the development of IoT technical components capable to enable the aforementioned characteristics for the realisation of the supply chain complete visibility. Each of these components is described in a dedicated section, to provide its definitive features:

- The **Smart Router** oversees providing information about position and time of the Smart PI container , as well as to monitor it internally with a pervasive and ad-hoc sensor network. It is the enabler for the realisation of the improved granularity in the goods monitoring, supporting the definition of the Smart PI Pallet and the Smart PI packet. Moreover, it enables the distributed monitoring, allowing the introduction a scalable set of IoT devices (internal and external the container) capable to collect added value information. The smart router is positioned in the standardisation path as compliant with the requirements with DCSA, ISO and 5G guidelines.
- The **Smart Gateway** oversees managing the connectivity within PI-means and PI-infrastructure. The smart gateway is fully compliant with the specification suggested to realise such type of devices by DCSA standard.
- The **IoT Cloud platform** oversees storing the collected data by the IoT devices, as well as to implement secure and ad-hoc transactions sharing the correct information with the stakeholder involved in a certain supply chain transaction.

From the technical point of view, this report describes the completion of the Cloud IoT platform, the core components to implement interactions between IoT devices in the field and the PI world, toward the improvement and the optimisation of the logistics services. In the following, the final finding related with the realisation of the Cloud IoT platform are listed:

1. Investigate regarding **PI common language**, suggesting interaction models from sensors or toward the users. While in D2.7, we have proposed the exploitation of the generic SensML standard approach to implement a semantic interoperability in the IoT transactions, in D2.8 we consider the general view proposed by DCSA standardisation framework. DCSA is the association of the containers' shippers, and the standard released in May 2020 defines approaches thought by the business stakeholders for the business realisation. At this stage, the technical and syntactic interoperability with this standard is guaranteed, while the semantic is guaranteed agreeing on a set of data models with the other project partners. As future work, we expect the DCSA standard next release to improve the semantic interoperability considering the suggested and standardised the data models (for further details see Sec. 3.4.2 and 3.4.4).
2. **Realisation of a secure and ad-hoc set of APIs to collect and share the data, based on standardised approaches**, implementing high level interoperability techniques, capable to support the forthcoming standard release (e.g., DCSA standard release 2).
3. **Cyber-secure and ad-hoc access**. Data privacy and protection is one of the hot topics in the IoT and Internet domain. In this report we describe our approach considering the standardised approaches currently used in the Internet world to implement secure and ad-hoc transaction from the sensors or toward the PI world. Also in this scenario, we are waiting for the suggestions from the DCSA standard release 3, related to the security and privacy issues. For further details see Sec. 3.4.3 and 3.4.4.
4. **Generation of events derived from the measurement gathered by the IoT devices.** Events represent punctual information extracted from the measurements of the sensors installed on the

IoT devices within the smart container e.g., bump detection, movement detection, threshold exceeding, etc. This information allows to provide advanced service the storage status of the goods all along the supply chain. Introducing the Smart PI Pallet concepts, the local detection of important, dedicated, and special events, without the needs of reconfiguring the whole system. For further detail see Sec. 3.5.1 and 3.5.2.

5. Big data analysis on the data retrieved from the IoT devices generating **high level metadata and statistics** (see Sec. 5.1.3). In this manner, raw data will be transformed in more comprehensible and aggregated information, added value services are provided to the PI users (to increase their satisfaction and trust), a strong instrument to support fact-based data-oriented decision making is provided.
6. Realising an **ergonomics Graphical User Interface** (**GUI**), capable to visualise in a comprehensible manner the route and the status of the containers. A **PDF report generator** completes the visual reporting portfolio proposed by the ICONET Cloud IoT platform.

Finally, this report is released with two months of delay because the COVID-19 pandemic that has affected the development, as well as the deployment of the IoT devices on the field. In several points of the report the criticalities introduced by the pandemic are highlighted, as well as the actions to mitigate its impact. The main to highlight is the realisation in cooperation with ITA **of a tool to emulate the IoT devices behaviours**, producing coherent routes and capable to interact with the Cloud IoT platform as the real component.

References

GSMA . (2018, April). NB-IoT Deployment Guide to Basic Feature set Requirements.

3GPP. (2016, August 19). "Standardization of NB-IOT completed".

Containers Owners Association (COA). (2019). *Guide to Container Tracking and Telematics Technology: An Overview of Technology Issues and Choices for Container Operators.*

DCSA. (2020). *IoT Container Standards - IoT Standard for Gateway Connectivity Interfaces.*

Digital Container Shipping Association (DCSA). (2020). *IoT Container Standards - IoT Standard for Gateway Connectivity Interfaces.*

ETSI. (n.d.). *IOT White Paper*. Retrieved from https://www.etsi.org/images/files/ETSIWhitePapers/IOP%20whitepaper%20Edition%203%20final.pdf

GSMA. (2018). *Mobile ioT in the 5G future - NB-IoT and LTE-M in the context of 5G.*

IETF. (2000). *RFC2818 - HTTP Over TLS.* Retrieved from https://tools.ietf.org/html/rfc2818

IETF. (2015). *RFC 7617 - The 'Basic' HTTP Authentication Scheme.* Retrieved from https://tools.ietf.org/html/rfc7617

INFORM. (n.d.). *The INFORM company*. Retrieved from https://www.inform-software.com/

INFORM. (n.d.). *The syncrosupply platform*. Retrieved from https://www.syncrosupply.com/

IoT-EPI platform. (2018). *Advancing IoT Platforms Interoperability.* River Publishers Series in Information Science and Technology. Retrieved from https://iot-epi.eu/wp-content/uploads/2018/07/Advancing-IoT-Platform-Interoperability-2018-IoT-EPI.pdf

ISO. (2016). *ISO 19079:2016, Intelligent transport systems — Communications access for land mobiles (CALM) — 6LoWPAN networking.*

ISO. (2016). *ISO 19080:2016, Intelligent transport systems — Communications access for land mobiles (CALM) — CoAP facility.*

ISO/TC204. (n.d.). *Intelligent transport systems*. Retrieved from https://www.iso.org/committee/54706.html

ITS standards. (n.d.). *Cooperative intelligent transport systems (C-ITS)*. Retrieved from http://its-standards.info/Guidelines/References.html

New Generation Sensors. (n.d.). *Cloud IoT platform APIs - Shipment Service APIs and Shipment Statistics API*. Retrieved from https://docs.iconet.ngs-sensors.it:8006/

NGS. (n.d.). *The Smart PlantOne project*. Retrieved from https://cordis.europa.eu/project/rcn/213263/factsheet/en

RAK. (n.d.). *RAK 5010*. Retrieved from https://store.rakwireless.com/collections/nb-iot-boards/products/rak5010-nb-iot-tracker

RFC8428, I. (August 2018). *Sensor Measurement Lists (SenML).*

Ruuvi. (n.d.). *RuuviTAG*. Retrieved from https://ruuvi.com/ruuvitag-specs/

Semtech. (n.d.). *https://www.semtech.com/*.

Shneiderman, B. P. (2016). Designing the user interface: strategies for effective human-computer interaction.

Swagger. (n.d.). *OpenAPI Specification*. Retrieved from https://swagger.io/specification/

Swagger. (n.d.). *OpenAPI Specification*.

The AGILE project. (n.d.). Retrieved from http://agile-iot.eu/

The BEinCPPS project. (n.d.). *http://www.beincpps.eu/* .

The CHARIOT project. (n.d.). Retrieved from https://www.chariotproject.eu/

The IoT-EPI portale. (n.d.). Retrieved from https://iot-epi.eu/

TU Delft. (n.d.). *https://www.tudelft.nl/3me/afdelingen/maritime-and-transport-technology/research/transport-engineering-and-logistics/theme-3-real-time-coordination-for-operational-logistics/synchromodal-container-transport/*.

UN/CEFACT. (2019). *Smart Containers - Business Requirements Specifications.*