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

D1.12 PI Protocol Stack and enabling networking technologies -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	PI Protocol Stack and enabling networking technologies Final		
Work Package	WP1 ICONET PI Framework (PI Business Models and Key Enablers)		
Contractual due date	30/11/2020	**Actual submission date**	23/11/2020
Nature	Report	**Dissemination Level**	Public
Lead Beneficiary	EBOS		
Responsible Author	Philippos Philippou (eBOS)		
Contributions from	ILS		

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

Disclaimer

[1] According to ICONET's Quality Assurance Process:
3 months before Deliverable's Due Date: 50% should be complete. Review by the Quality Manager.
2 months before Deliverable's Due Date: 80% should be complete. Review by the Quality Manager.
1.5 months before Deliverable's Due Date:100% should be complete. Review by the Quality Manager and Peer Reviewer(s).
M: Sent to Inlecom for Submission to the EC after addressing all comments by Quality Manager (EBOS) and Peer Reviewers

Copyright message

Table of Contents

List of Figures

List of Tables

Glossary of terms and abbreviations used

Abbreviation / Term	Description
AS(n)	Autonomous System (Architecture)
API	Application Programming Interface
ATM	Asynchronous Transfer Mode
DI	Digital Internet
DoA	Description of Actions
E2E	Elastic Compute 2 {Amazon Service}
EDI	Electronic data interchange
ERP	Enterprise Resource Planning system
ERP	Enterprise Resource Planning
ETA	Estimated Time of Arrival
ETD	Estimated Time of Departure
FF	Freight Forwarder
GS1	Global Standards
ICT	Information and Telecommunication Technology
IoT	Internet of Things
LLs	Living Labs
LSP	Logistic Service Provider
NFV	Network Function Virtualisation
NOLI	New Open Logistics Interconnection
OLI	Open Logistics Interconnection
OSI	Open Systems Interconnection
OSPF	Open Shortest Path First
PaaS	Platform As a Service
PI	Physical Internet
PoC	Proof of Concept
PoD	Proof of Delivery
RIP	Routing Information Protocol
RFID	Remote Frequency Identification
RON	Resilient Overlay Network

SDN	Software defined Networking
SNA	Systems Network Architecture
TCP/IP	Transmission Control Protocol/Internet Protocol
T&L	Transport and Logistics
UDP	User Datagram Protocol
UN/CEFACT	United Nations Centre for Trade Facilitation and Electronic Business
VXLAN	Virtual Extensible Local Area Network
WMS	Warehouse Management System
WP	Work Package
XML	Extensible Mark-up Language

1 Executive Summary

This is the third and final version of the Project's D1.10 Deliverable addressing the Task 1.6 "PI protocol stack and enabling technologies". The requirements of the said task and specifically the analysis and assessment of the analogies and applicability of DI elements to the PI evolving concepts have well been detailed and discussed in the first two deliverable reports D1.10 and D1.11. The current and final third version refines the designed components following their implementation under WP2 and their actual application in the project's LLs particularly as regards to the formation and testing of the PI services.

The purpose of the first version report was to analyse layered service-oriented PI models proposed in the literature (most notably the OLI model) it then proposed fundamental capabilities of PI which would be used to inform subsequent design activities in Work package 2, that will prototype fundamental PI services such as networking and routing. In turn, these PI services will be used to simulate the operations in PI models that are derived from the Project's Living Labs. The previous version therefore did not aim to propose the 'best' PI architectural model, but to analyse, dissect, evaluate and critique existing models, and then synthesize a 'best of breed' set of PI characteristics that would form the input to WP2 PI service design activities. Since the existing PI models are largely theoretical the report used analogous developments in the more mature digital Internet, in order to appraise the significance of existing PI proposals. With the first version providing the framework of the analysis and analogies needed to satisfy requirements of task T1.6, the second version of the deliverable briefly revisited the findings of the initial work and then reviewed the outcome of these aspirations in the service tasks of WP2 and provided the next steps ahead. A more solid approach was already put in place by WP2 technical team and also more elements have been incorporated in the design processes, documented in sections below. Moreover, an analysis of the services under each layer showcased the path that the ICT platform and architectural endeavors need to consider as well as the simulation work, crucial to the testing of the PI performance and functionality. However as mentioned before in the reports produced, due to the conceptual nature of the reference models as well as a number of unknown factors in the PI concept realization, there is clearly more work to be done before concluding on subjects at hand.

Furthermore, the second version analysed the services offered by each layer as requested by the relevant subtask in relation to the upper and lower layer of each, and also visited and discussed the services within each layer for a better insight to the role that the OLI model can really play in the realization of the PI. The aim was a complete, efficient and sustainable logistics service offered across markets and industries. Finally, the ongoing testing in the project's Living Labs is also a decisive factor on the extent and level of adopting the current DI successful elements and components to the PI infrastructure.

The third and final report is structured in a similar way to the first 2 reports previously submitted i.e. the areas that inspired the protocol stack however in this last version the document concludes on the elements that the PI has adopted, how previous work has finally shaped crucial work under WP2 and the adjustments made to enable the adoption of the various components. Finally, the important outlook of the work beyond project end is also discussed. The main conclusion of this third version and indeed of the whole series of deliverable reports addressing the PI Protocol Stack and Networking technologies is that the Physical Internet is inspired and can truly be architected along the same principles as the digital Internet with many shared (in functionality) elements. The latest pandemic and its effect on the Supply Chain industries of the world has illustrated vividly the need for a new model for the industry's operational framework. One that puts emphasis on efficiency, energy impact, interconnecting and cost solutions considerations more than anything else without affecting (or necessarily improving) the customer service. Latest news from the T&L segments all refer to the need for data digitization, synergies, collaborations and shared models/ language via the use of technologies. ICONET as well as other future projects will, does exactly that. One can also refer to similar methodologies in a number of other business domains like Production and Operations management where again standardization and interoperability in a controlled and measured environment can offer advantages in capacity management and quality products and services.

2 Introduction

The majority of the work related to the requirements of Task T1.6 has been performed in the first two versions of this deliverable, D1.10 and D1.11, submitted on April 2019 and February 2020 respectively. The primary aim of this third version of the deliverable, and having gone through a detailed analysis and assessment of the various PI architectural stacks (OLI, NOLI, and others) and networking technologies, is to analyse the findings and final adoptions based on the progress, research and work performed in the last period. As well as list a set of relevant Internet standards and technologies which assisted the design work in WP2 i.e. the PI Control and Management Platform and Reference Architecture as well as the design testing domain of the LLs in WP3. Moreover, the interoperability and very definition of the PI Services is affected by the current work and again the related efforts in WP2 is addressed to reflect current developments

Beyond project end, this work will hopefully inspire more work to reinforce the concepts discussed and pave the way for a full implementation of the PI network.

2.1 Deliverable Overview and Report Structure

In this section we provide an overview of the deliverable's structure. We explain how it was informed by other Project tasks/deliverables, and how in turn it will inform subsequent ones. Finally, being he last version of this deliverable series we shortly explain how the deliverable can inspire future work beyond project end.

This document is the third and final release of the deliverable aiming to satisfy the objectives under the task T1.6 'PI Protocol Stack and enabling Networking Technologies', of which initial work was carried out in D1.10. It shortly reconsiders the findings of D1.10 and D1.11 and then reviews the adaptation of the recommendations to the design tasks of WP2 i.e. the PoC Platform, Reference Architecture and design of key PI services by referencing deliverables: (a) D2.2 'PI Reference Architecture Final', (b) D2.21 'ICONET PI Control and Management Platform – Final version' and (c) D2.5 'PI networking, routing, shipping and encapsulation layer algorithms and services Final'

In addition to how the series of reports addressing task T1.6 has further influenced work in the above final versions. More specifically the identification of the services under each layer and interrelationship between layers that will guide the design of the architectural effort and the PoC platform for the required connectivity at digital level. It finally examines the influence of previous work on the deployment of LLs and work under D2.3 which shapes the PI (Supply Chain) Activities into layered architecture of services and which are these functions that each layer supports.

Chapter 1 provides the Executive Summary.

Chapter 2 details the ICONET's DoA commitments and Task description and the mapping to the deliverables' output, with details on how these are addressed in the report's sections. Furthermore, it gives the Report structure.

Chapter 3 provides an overview of the findings of the first version deliverable D1.10 on PI Protocol Stack and a review of the work carried out in design tasks under WP2 namely PoC Platform, the Reference Architecture and the design of PI activities as services. Deliverables D2.21, D2.5 and D2.2 are referenced. It also reviews the effect of the OLI and other models on the design and development of work in the project's Living Labs as well as the effect of the specified services on the simulation work. Deliverable D3.1 'Planning & Monitoring of Living Labs' Activities v1' is referenced. Finally, it provides a list of the services offered by the OLI layers in the Logistics arena of today's world.

Chapter 4 discusses the Networking Technologies as PI enablers with a short revisit of findings and recommendations in version 1 and continues to evaluate the effect on work done in WP2. Deliverables D2.19 and D2.1 are referenced. Finally, it offers the next steps.

Chapter 5 lastly provides an overview of the report's findings and some final conclusions

3 The OLI Protocol Stack

3.1 Introduction

This section firstly revisits the work done in the previous two versions and briefly lays out the summary of the findings. It then continues to examine how the said work has finally shaped the layered stack of services and the design tasks of WP2 (as intended to do) if any barriers or problems were identified while implementing the ideas and concepts under study.

3.2 Background Work of Version 1 (D1.10) and 2 (D1.11)

3.2.1 Context

Deliverable D1.10 is the first version of this deliverable, and it has discussed in depth the concepts of OLI in an effort to investigate the possibility of adopting the layered structure of services to the PI.

To avoid re-stating the findings of the first and second version of this deliverable, the current document shows these findings in Annex I. However, if and where needed, and for the purposes of the present version, it revisits the main findings to draw conclusions. More specifically, Annex 1 discusses:

1. Background and Fundamental concepts of OLI
2. The Importance of π -units from the perspective of the shipper, the freight forwarder and the carrier
3. Mapping Shipments To π -Units
4. Transporting Cargo in the π-Network
5. Open Logistics Interconnection (OLI)
6. New Open Logistics Interconnection (NOLI)
7. The Layered Protocol Analogy of the Internet and PI
8. Analysis of information entities and flows in the OLI model

Deliverable D1.11 is the second version of this deliverable and has provided a more detailed picture on how the above elements affected the work under WP2 and namely the PI Control and Management Platform and the Reference Architecture as well as the evolution of the WP3 Living Labs (LLs). It went on to list the specific services under each layer as required by Task 1.6 and interaction amongst the services in each and every LL in an effort to satisfy task requirements. The findings of the second version are summarized in the sections below.

3.2.2 Applicability of OLI Layers to ICONET

3.2.2.1 Introduction

Apart from the above, we revisit the expanded model of the OLI model and the more detailed definitions of its layers, including the identification of their core concepts, functionalities and services exposed by them, to examine again how they apply to the ICONET Project. Deliverable D1.10 has found the following:

3.2.2.2 Logistics Web Layer

The Logistics Web Layer monitors and validates the capabilities, capacities, prices and performances of π -nodes and π -means, in general of π -service providers, as well as the status of signed contracts and of deployed π - containers [1]. In ICONET capabilities, capacities, prices and performances of π -nodes and π -means are mainly captured in D1.1 (PI-aligned digital and physical interconnectivity models and standards). On the physical side, this deliverable covers the existing and emerging digital (i.e. data, transactions, events, etc.) and physical (i.e. packaging, operational facilities, handling systems, vehicles) interconnectivity models and standards, and associated adoption barriers and drivers [5].

This implies that there is standardization of load units, which allows for the automation of cargo handling at transshipment points. For example, small-sized load units that can be accommodated within intermodal transport containers like modular pallets or the M-box of the EU project MODULUSHCA [6], targeted for fast moving consumer goods.

3.2.2.3 *Encapsulation Layer*

This layer as provides the means for efficiently encapsulates products of a user in uniquely identified π-containers before accessing the π-networks [1]. The essential element of this layer is the visibility of a π-container. A key technology for the Encapsulation Layer is Blockchain [7]. Within the PI network, blockchain can be used for establishing trusted, auditable and secure distributed ledgers of transactions as containers flow within the PI network. Blockchain utilises smart contracts to form and maintain a trail of transactions between the shipper and the dynamically allocated LSPs as the PI containers get handled and forwarded from one PI node to the next [1]. The investigation of the Blockchain technology within the aforementioned concept is being conducted in T2.4 Blockchain mechanisms for secure and privacy-preserving distributed transactional ledgers.

3.2.2.4 *Physical Layer*

This layer monitors the physical objects of PI involved in handling and transporting cargo such as means of transport, vehicles, carriers, conveyors, stores and sorters. ICONET project investigates these concepts captured in D1.1 (PI-aligned digital and physical interconnectivity models and standards) where the solutions for generalising and functionally standardising unloading, orientation, storage and loading operations is being investigated.

The work carried in analysis and collaboration between technical partners and LLs leaders alike revealed that the Physical Layer was not in the scope of the project. More explanation on this in section 3.3 below.

3.2.2.5 *Link Layer*

In ICONET this layer provides mechanisms for efficient and reliable shipping of (sets of) PI containers from shippers to final recipients. The management of the procedures and protocols for configuring the quality of service, monitoring, verifying (acknowledgement), adjourning, terminating and diversion of shipments in an end-to-end manner is being conducted in ST2.2.3 Shipping algorithms and services will be specified in deliverable D2.4 ('PI networking, routing, shipping and encapsulation layer algorithms and services v1').

Latest progress on analysis of activities and developments has indicated that the Link Layer functionality was handled by the Shipping Service. More explanation on this in section 3.3 below.

3.2.2.6 *Network Layer*

The network layer focuses on the interconnectivity, integrity and interoperability of networks within the Physical Internet [1]. For ICONET this layer provides the networking and shipping algorithms and services. Smart assignment of PI containers to PI means on PI links ensure the flow of PI containers across the PI network and reliable shipments in an end-to-end manner. Deliverable D2.4 ('PI networking, routing, shipping and encapsulation layer algorithms and services v1'), provides a reference design and implementation for core networking, routing, shipping and encapsulation layer protocols and services. The use of Smart-Routers and Smart-Gateways, Smart Interfaces and Smart Sensor-based IoT Services enable continuous tracking and reporting and contribute in the information flow to achieve a reliable end to end routing. Deliverable D2.6 ('Smart PI Containers – Tracking & Reporting as a Service v1'), provides the IoT mechanisms for transforming PI Containers into Smart PI Containers that are utilized by the Networking Layer.

3.2.2.7 *Routing Layer*

Based on the networking services of the Network Layer, the Routing Layer handles the efficiency in the transportation of π-containers from its source to its destination by selecting the optimal routes. To achieve routing optimisation, advanced techniques are used in Deliverable D2.12 ('Intelligent Optimization of PI Containers and PI Means in PI Nodes v1'), to incorporate cognitive capability into the components of the PI Node Control associated to the LL use cases. Moreover, machine learning and/or graph analytics techniques that support PI Node operations in smart decision making are researched and developed to enable optimised orchestration of PI logistic network objects, thus enabling smart decision-making within a PI Node. Analytical algorithms to support best route decisions, where best route might be on the basis of costs, throughput or emissions are part of deliverable D2.4 ('PI networking, routing, shipping and encapsulation layer algorithms and services v1'), including optimisation that considers hub topology, network state and cargo type.

3.2.2.8 Shipping Layer

The Shipping Layer provides the functional and procedural means for enabling the efficient and reliable shipping of sets (corresponding to orders for instance) of π -containers from shippers to final recipients [1]. Implementation Theory [10] could also be implemented within the scope of the Shipping Layer potentially in combination with the Routing Layer. It is a theory that encapsulates the engineering side of an economic theory [10] which, "given a social goal, characterizes when we can design a mechanism whose predicted outcomes (i.e., the set of equilibrium outcomes) coincide with the desirable outcomes, according to that goal" [10, p.1]. Another potentially useful mechanism that will enable the procedural means of the Shipping Layer is Blockchain smart contracts that form and maintain a trail of transactions between the shipper and the dynamically allocated LSPs. The investigation of the Blockchain technology within the aforementioned concept is being conducted in task T2.4 ('Blockchain mechanisms for secure and privacy-preserving distributed transactional ledgers').

3.2.3 Summary

In summary of Annex 1 and for the purposes of the present version, the main characteristics of the protocol stack and its attributes do share relevance to the PI vision and the ICONET project. The interconnectivity and network communication which lie at the core of the PI concept not only can be positively influenced by concepts of the matured digital internet like TCP/ IP protocol principles but also certain elements can be adapted and transferred. There are however differences between the DI and the PI and specific areas (again identified in D1.10) needing more research and indeed this has been accomplished throughout the work developed in the recent time frame.

3.3 Analogies and Adopted protocol stack model to ICONET project

Compared to the OLI, the NOLI model leans closer to logistics networks rather than digital ones. The TCP/IP and the OLI models define their physical components in the lower layer [2]. For the TCP/IP model, this design was fit since the sole physical components are the devices that transmit data and the physical transmission medium. Nevertheless, in the Physical Internet and in extent, logistics, the carried objects are physical components whereas in a digital layer are data bits. In the PI, containers and cargo are objects.

Considering the aforementioned, Colin, Mathieu and Nakechbandi [2] suggested that the Physical layer is impossible to include definitions of all physical objects, therefore their definitions must be defined in the distinct layers when they first appear. Hence, the Product layer of the NOLI model attempts to define the possible cargoes and their specificities, including their "exact identification of the type of cargo, and its characteristics such as the fact that it is perishable or that it is fragile" [2, p.5]. In a similar manner, the Container layer defines the characteristics of the PI containers (e.g. size). The PI means correspond to the physical electronic components.

The Layered Protocol Analogy of the Internet and PI closely follows the five layers of the Internet Protocols and defines the analogies at a high level, similar to the other models as well. However, work in LLs and WP2 activities has led to the conclusion that a hybrid version of the OLI layer (with elements of the NOLI one) would most appropriately suit the LLs efforts. The table below explains this:

Table 3-1 Comparison and adoption of Protocol Stack Layers

OSI Layer	OLI Layer	NOLI Layer	ICONET Layer	Resulting Service
Application	Logistics Web	Product	Logistics Web	Logistics Web
Presentation	Encapsulation	Container	Encapsulation	Encapsulation
Session	Shipping	Order	Order	Shipping
Transport		Transport	Transport	
Network	Routing	Network	Routing	Routing
	Network		Network	Network
Data Link	Link	Link	Link	-
Physical	Physical	Physical Handling	Physical	-

The study carried forward in the design sections of WP2, the relevance in the considerations of the Reference Architecture as this is described in details in deliverable D2.2 'PI Reference Architecture (Final)' submitted on 31/7/2020 and analysis through the LLs business spectrum has led to the conclusion that a mixture of the layers and resulting services shown above under column 'resulting service' would serve better the needs and aims of the project. The notion above was also discussed in final version of the deliverables D2.5 'PI networking, routing, shipping and encapsulation layer algorithms and services'. To explain, below a brief explanation of the best solution identified with the Layers:

1. The Link layer manages and ensures the smooth flow of goods between PI Nodes. To achieve that, this layer must enable the pre-evaluation of potential options, identification of potential issues across the supply chain and the suggestion of appropriate corrective actions to mitigate. In ICONET this layer provides mechanisms for efficient and reliable shipping of (sets of) PI containers from shippers to final recipients. The management of the procedures and protocols for configuring the quality of service, monitoring, verifying (acknowledgement), adjourning, terminating and diversion of shipments in an end-to-end manner is being conducted by the Shipping algorithms and services as specified in deliverable D2.5 'PI networking, routing, shipping and encapsulation layer algorithms and services final, and as such, the functionality of the conceptual Link Layer will thus be handled by the Shipping service.
2. The Physical Layer monitors the physical objects of PI involved in handling and transporting cargo such as means of transport, vehicles, carriers, conveyors, stores and sorters. ICONET project investigates these concepts captured in D1.1 (PI-aligned digital and physical interconnectivity models and standards) where the solutions for generalizing and functionally standardizing unloading, orientation, storage and loading operations is being investigated. The Physical layer is responsible for the physical actions that need to happen for a shipment to begin and conclude its' trip throughout the π-Network based on the decisions made of the other services.
   As these actions already occur with a variety of different ways in the current logistics domain, further technical work on Physical layer was deemed out of scope for the ICONET project, and instead, focus was in providing already established outputs that can be used for already occurring Physical actions (such as picking lists for product loading etc.).
3. In the NOLI model, the Shipping layer of the model is further divided into Transport and Order services. This division is helpful in order to better conceptualize the separation of concerns when designing the Shipping protocol from a technical standpoint, while also being more closely related to the current state

of logistics. As mentioned in D1.10, it is important to note that while generally the layers of the NOLI model follow a top to bottom flow of information, in order to be able to better communicate and propagate needed data, inputs & outputs can travel both ways, not necessarily following the assumed order of the OLI/NOLI layers. Layers can also be unified or changed in order to serve more efficient flow of data and interconnection. In short, it is more useful to think these layers as a conceptual guideline while acknowledging that a full technical implementation could potentially generate a new conceptual paradigm. As such, the Transport layer will handle all the communications & data exchanges needed for a set of PI containers to be transported through the PI network, while the Order layer will monitor & update the PI order state from initialization to termination, under the umbrella of the Shipping services.

The remaining services as described in section above and analysed in a number of deliverable reports, remain at the core of the design work and guide amongst other components the structure of the resulted architecture.

In conclusion, for the reasons mentioned above the project team has followed this hybrid version of the protocol layers to better suit the interoperability and functionality of the designed and tested services.

3.4 The ICONET Protocol Stack and the Service Design Tasks

3.4.1 Introduction

This section discusses how the Service design tasks and work in WP2 have ultimately taken recommendations on board of findings of the task T1.6 ‘PI Protocol Stack’ performed in D1.10 and D1.11. It lays out considerations and final touches concluding on the Integration platform and the Reference Architecture final schema as this was presented in submitted D2.2 ‘Reference Architecture’ deliverable.

3.4.2 The ICONET Layers model and the PoC Integration Platform (PI Control and Management platform)

3.4.2.1 Earlier work

The first two versions of this deliverable, D1.10 & D1.11, laid out the aspects of the TCP/ IP, OSI, OLI and NOLI reference models and how these elements could benefit, amongst other elements, the design efforts of WP2. Based on that analysis the D2.20 ‘ICONET PI Control and Management Platform – Intermediate version’ submitted in February 2020 has analysed findings and discussed the PI connectivity aspects on how the OLI and NOLI reference models did in fact shape (amongst other factors) the connectivity scenarios that effectively were finalized and implemented in the project. The D2.20 covered the majority of work related to the PoC integration with clear indication of the virtual networks’ configuration, the PI Services deployment and service/ simulation integration framework.

The work shown through the above WP2 report, supported not only the simulation services as testing agent but enabled the design, development, deployment and integration of al the PI services developed which in turn were based on the PI Protocol stack considerations. From the overall data flow/ sequence diagrams established to the data models, topologies and roadmap to the defined API definitions, inputs/ outputs and interrelationships for services deployment and integration, the principles of the layered stack of services inspired by the OLI/ NOLI models was kept in mind in this intermediate version with potential relationships and discovered interactions, mirroring at the end of the day the Supply Chains’ vital daily operations and of course the relevance of the conceptual PI network.

The chosen approach was a decentralized integration as a self-managed peer to peer network with no central management of the PI network based on the project requirements as well as the discussions between the WP2 partners and the LLs use cases. Node to node communication as a key element to the PI concept remains a priority and legacy systems will access the offered PI services and exchange information. Integration between each service to the simulation was structured to further advance the LLs cases and GPICs. Finally, the external connectivity with the IoT tracking service was also discussed and showcased. The figure below shows the peer to peer decentralized approach agreed by the WP2 team ensuring a simple development framework.

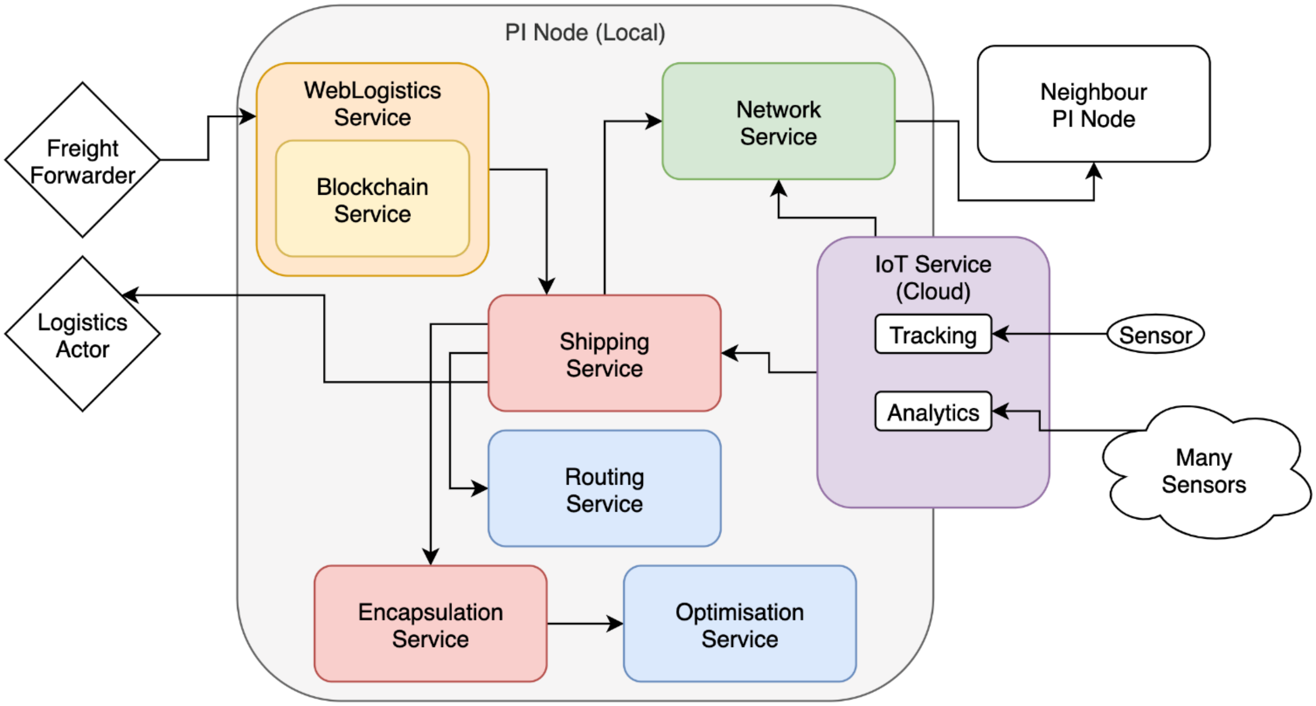

Figure 3-1 Decentralized Architecture

In ICONET Amazon Web Services (AWS) was finally chosen as the cloud vendor as AWS has a particularly rich catalogue of PaaS options (see 3.4.2.2), some specifically identified as useful for ICONET such as Lambda and the API gateway. ICONET also engaged the use of Virtual Private Clouds, Software Defined Network Infrastructure, Elastic IP Addresses, Elastic Load balancers, Elastic Container Registries, Elastic Container Services, Identity and Access Management, Deployment Pipelines and more services beyond. All of these services have commercial or open source alternatives, but a significant amount of time and effort was saved by choosing PaaS alternatives, and at a very low financial cost. More details can be found in D2.19 and D2.20 deliverables.

3.4.2.2 Latest developments

The last deliverable report D2.21 'ICONET PI Control and Management Platform – Final Version' submitted in October 2020 enhanced the flexibility, security and robustness of the PI Service networks, node-to-node communication and PI Service deployments by ensuring they are cost effective and valid in real world scenarios. The considerations in earlier work still remain valid. Furthermore, it introduced the concept of PaaS as a supporting and deployment cost-effective framework for engaging cloud services in conjunction with containerisation and microservices matching the PI Service stack with an outlook to the future PI research domain.

The document also addresses the service discovery under which services have been deployed to allow the integration and testing across various use cases of the LLs and this services interoperability is a key factor of the Protocol stack and discussions earlier. Services are no longer in isolation but can call upon other services for input. This discovery is possible by DNS (Domain Name Server) as D2.21 describes. Lastly, IoT connectivity as a tracking was also addressed and designed for in the AWS cloud.

Finally, the report addresses the important point of adoption of the technical assets produced by ICONET in real world conditions following a structured approach for deployment. Indirectly this includes all aspects and elements with analogies transferred through from the DI (and thus work in the series of reports from D1.10 to D1.12) and effectively moving forward the concepts of PI in terms of the Layers and networking technology inspirations discussed also later on. The resulting ICONET services are explained in section 3.3 above and the main conclusion is that all studies and cross collaboration of the ICONET WPs resulted in a final shape and form of the reference model and related services that best suited the LLs requirements as well as the design endeavours of the technical teams.

3.4.2.3 Characteristics

Task T1.6 aims to investigate and ultimately enable a smooth universal interconnectivity in the PI network by borrowing analogies and elements from computer network standards describing the interoperability between devices and software as well as transmission of information through the internet. The purpose at the beginning was not to solidly identify an exact architectural solution but to suggest a best 'recipe' of components which could under further analysis and testing provide a well-suited design framework for the architectural scope.

This interconnectivity of PI services and secured, reliable exchange of data between PI nodes and PI means, remains a vital element to the PI vision and ensures, amongst others, the smooth flow of containers in the PI network. Analysis has shown that adaptation of components/ layers or even unification of them could provide solutions towards achieving the above goal. The way the information is shared between the layers and the how this information is made available to the PI 'user' is crucial and an area which needs to be further investigated.

Work in WP2 is inspired on the reference protocol architectural models discussed in detail in D1.10 and D1.11. The design efforts in D2.20 and latest D2.21 clearly identify the potential of these connectivity scenarios and relevant considerations and as shown above, scenarios have been already formulated. Furthermore, the results of the LLs which will span through to the end of the year 2020 as the testing area of the GPICS will perhaps unfold new issues and alongside other work in WP1 and WP2 workstreams, will eventually determine the best configuration pattern along the above design patterns with some fine tuning of the ingredients.

A strong integration pattern between the PI Services and between each of the service and the simulation service has already been described above and in detail in deliverables of WP2 and the Protocol stack has played a pivotal role towards that.

3.4.3 The ICONET Layers model and the Reference Architecture

3.4.3.1 Earlier work

Deliverables D1.10 & D1.11 have informed also the first iteration of the ICONET reference architecture together with other WP1 outputs like IoT developments and Blockchain technology. More specifically the layers of OLI model were used as a basis for defining the main architectural components providing a conceptual reference architecture for the design and development of PI network functions and services. In its first version Deliverable D2.1 'PI Reference Architecture v1' has analysed all WP1 findings and Figure below shows the initial iteration of the reference architecture with a clear identification of the services and the systems in place.

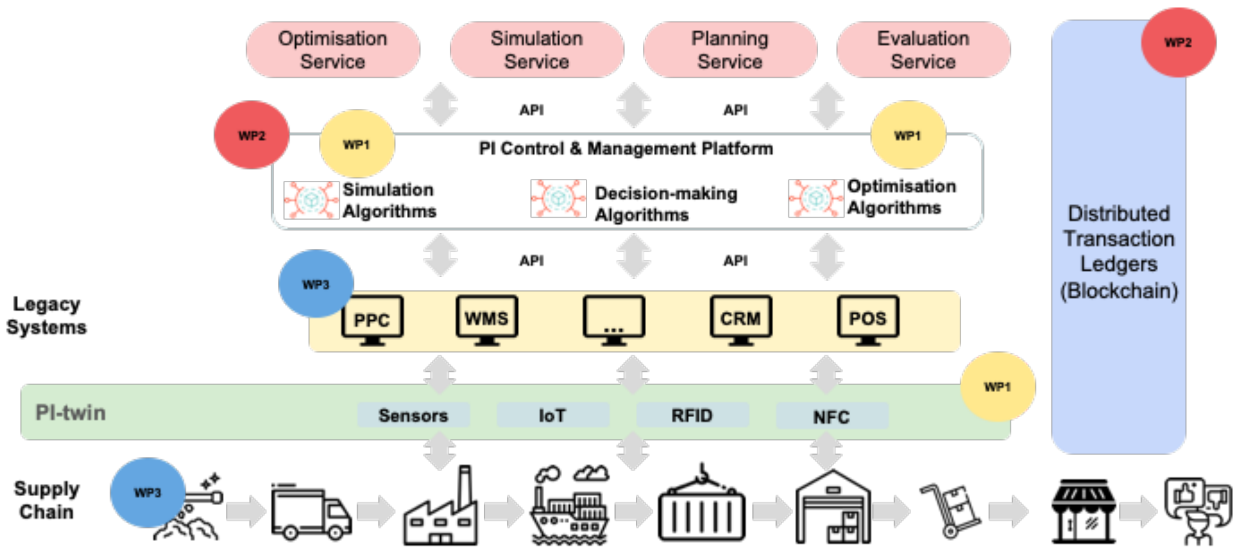

Figure 3-2 Initial Conceptual Architecture

Further research work and output from WP1 led to a more focused view of the different modules / services. These identified modules initially enable the key PI functionalities. Table below shows the key modules.

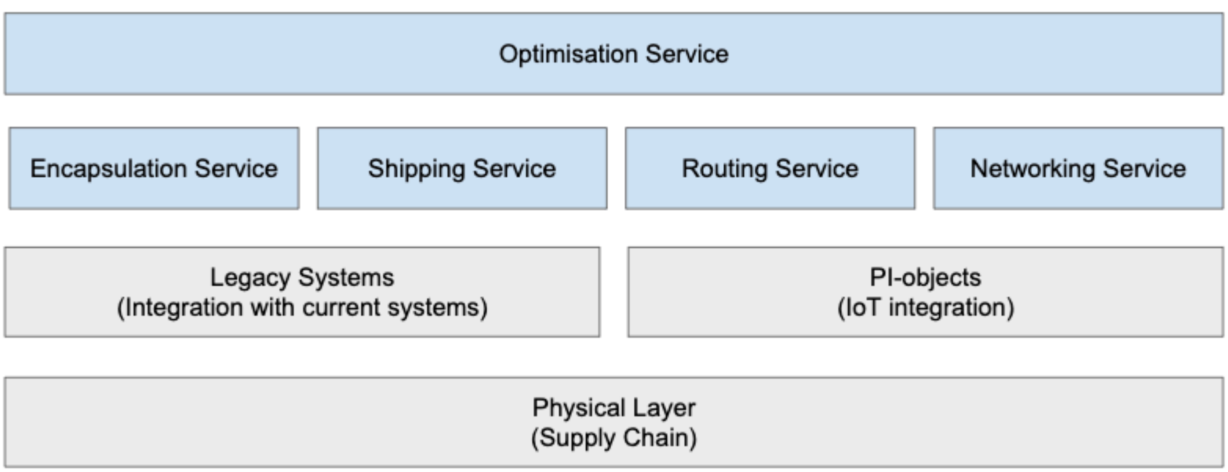

Figure 3-3 Initial Key Modules for the PI functionalities

An outline of these services with impact on architectural elements, and in relation to the OLI layers, was discussed previously. These requirements originating from the generic case study with supply chain scenarios, include technical aspects, decisions, events, and information/data flows that are needed from the architectural framework in order to realize the vision of PI. The full analysis can be found in Deliverable D2.1.

3.4.3.2 Latest developments

The analysis provided by the 2nd version of this deliverable, D1.11 and following the recent progress of work and developments in 2020, fed the design criteria of the architecture in terms of the potential technical implementations, establishing protocols, models and methodologies to achieve pre-specified objectives for the PI Control and Management platform. This is effectively done not only by surfacing the specific functions under each layer but also highlighting the interconnectivity and interoperability of each layer with the others as well as the integration with external legacy systems while maintaining security.

Close collaboration between the WPs of the project and design efforts of the architecture team have finally produced a blueprint of the reference architecture model to support the PI Network operations which D2.2, submitted in July of 2020, lays out. Figure below shows the Reference Architectural model with the functionalities split in three distinct planes. The Control, Management and Forwarding planes. A detailed analysis can be found in D2.2.

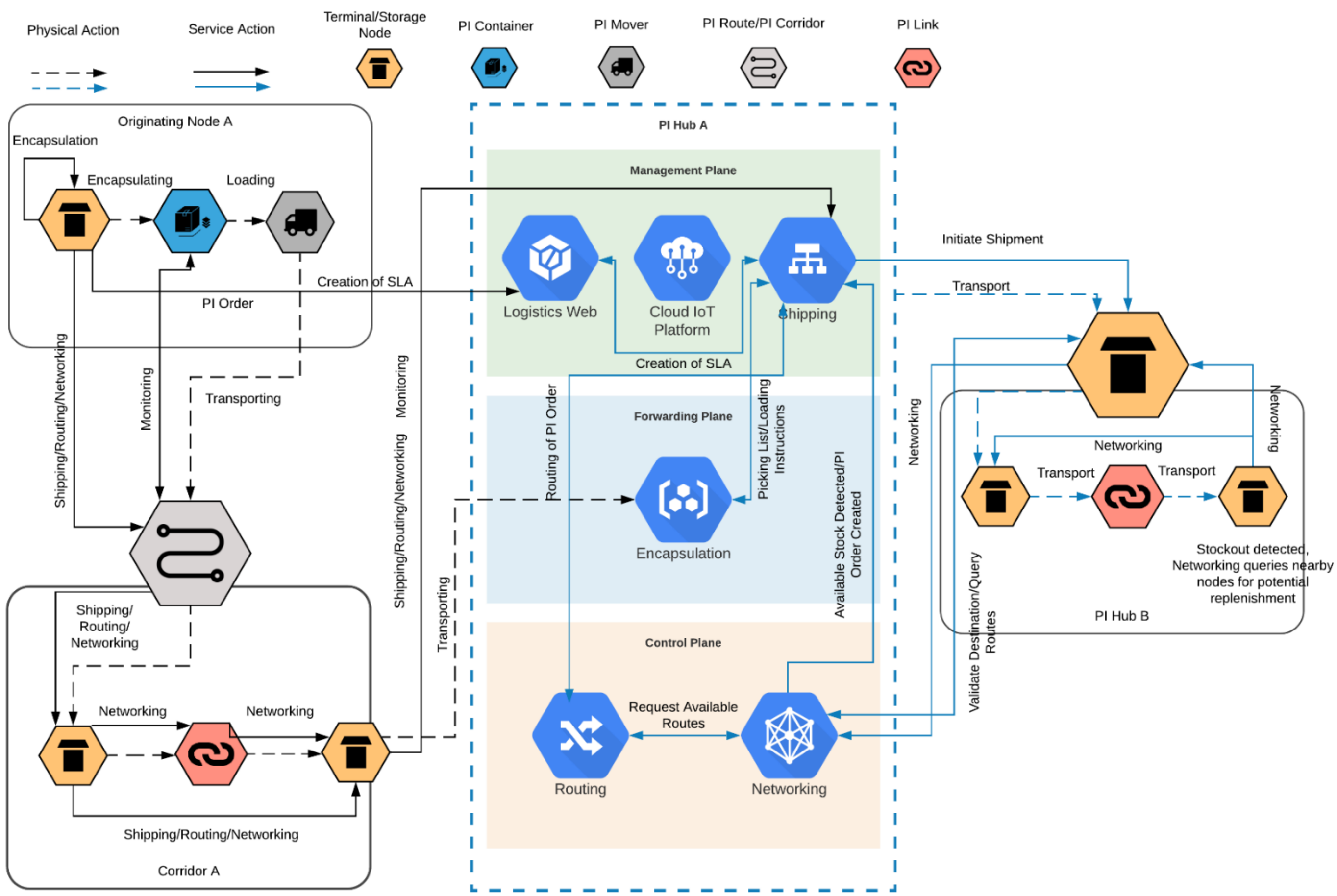

Figure 3-4 PI Reference Architecture

The main focus of the architectural work was on transforming key requirements, events and data required based on a generic scenario, as well as use cases driven by the project Living Labs in a reference architecture that addresses all required capabilities. Data specifications stem from the findings of WP1 deliverables and research conducted in the development of the multiple components of the ICONET project and their interactions.

3.4.3.3 Conclusion

The reference architecture must cover all supply chain stages (E2E). It must also support communication/data exchange between all supply chain actors. This means that all supply chain 'elements' must be PI-enabled and therefore the architecture needs to account for pi-containers, pi-hubs and all other elements. In addition, it is vital to identify any supply chain components that are not 'pi-upgradable' to ensure other sources of information are available (e.g. legacy enterprise systems like ERP and WMS).

The PI protocol stack has inspired the reference architecture in the following way. Since for example originally the Physical Layer has been defined as the network of all physical means of a supply chain network with the PI container at its core then this defines the data requirements. The date definitions and requirements define, in turn, the structures, data flows, common language (vocabulary) and system interoperability. Therefore, a PI container which Is to be uniquely identified and tracked while flowing in the PI network needs an identifier, linked to an IoT device (example a tracker) and in turn communicate with systems (perhaps external as well like ERP and WMS) or platforms transferring this data (e.g. location, temperature...). This data will serve as input to simulation, optimization, routing, shipping networking and encapsulation services. This data will then have to be exchanged to other PI services and finally made available to the 'consumers' of PI.

Similar approach has been adopted for PI nodes, PI means and so on. Difference being that the data will have to include other related information like frequency of services, routes etc.

At the end of the day these data specifications and services have been analysed to trace dependencies and define a common data model or ontology to organize the data and enable interoperability.

The D2.2 documents service requirements, definition of required inputs and expected outputs as well as dependencies between services, while also providing a more technically oriented approach to the potential architecture of a PI system. Major events, data and decisions that need to be considered throughout the journey of a PI-container in a PI-network, as well as a blueprint for designing and implementing a PI enabled architecture in a decentralized manner, were validated by the service development along with the simulated living lab scenarios.

3.4.4 The Services offered by each Protocol Layer

Subtask ST1.6.1 states the need to identify which are those services offered ('specifications') by each protocol layer to its upper and lower layer in an effort to refine and shape the inspirational work to the architectural components of the PI. However, suggesting that the services may be structured only between the layers would not be accurate and certainly would not lead to a sustainable and complete service. Effectively, supply services must be organized also between Logistics actors and service providers inside each and every layer in an effort to complement a sufficient supply chain cycle.

Previous versions of the deliverable had detailed the functionality of each and every protocol layer based on discussions. Moreover, deliverable report D2.5 in its latest edition submitted on 30/9/2020 reviews and details the nature and task for each of the four primary services i.e. Networking, Routing, Shipping and Encapsulation. For the purpose of this final version we do revisit the findings, having mind latest work in LLs, D2.5 and integration efforts in WP2. The table below is updated with current developments and findings to the day of the submission of the current report.

Table 3-2 Functions offered by each Protocol Layer in resulted ICONET schema

	Layer	Description	Functions
1	The Physical Layer	Operations related to the Physical Internet	*A Physical service was deemed out of scope for further technical work, as it would require significant effort to synchronize physical actions with the corresponding operations described in the PI concept and the Physical Layer.*

2	The Link Layer	Node to node transfer	*The functionalities and offerings under this Layer are unified in the Shipping service.*
3	The Network Layer	Interconnectivity, integrity and interoperability of networks	- The collection and integration of PI network information - The provision of PI Shipment specific information In detail: Consider key requirements and target KPIs to discover the best network (output will be set of hubs and links) appropriate for the requested order producing the routing table. Identify from the routing layer, the main nodes of the network and their functionalities, warehouse capacities, ports, train stations… Provide alternative to above networks/ paths to achieve service levels Assign PI means to PI containers Identify available transport services among two nodes. The output will be a list of transport means able to satisfy order requirements Consume input from external services (e.g. congestion levels on routes) to calculate as output the best alternative for routing decisions Acquire input from Link layer on the node condition or if delays exist and provide alternatives. The output will be alternative network(s)
4	The Routing Layer	Routing of the PI containers from starting point to their destination	Calculate best route /path for PI order Identify best route (optimal ones) based on input of criteria /filters including amongst others cost, times…) via a transportation plan (output will be a sequence of segments/ nodes including timing specifications) Provide re-routing to accommodate changes to original plan based on information from other layers. The output will be a set of routing decisions and a sequence of nodes to accommodate modifications to original schedule Provide transit times, transport means, ETAs and ETDs of routing scenario(s)

5	The Shipping Layer	Takes the role of the overall orchestrator Reliable shipping of PI containers via capabilities/ input of other services	• Study the management of the procedures and protocols for configuring the quality of service • Monitor, verify (acknowledge), adjourn, terminate and divert shipments in an end-to-end manner • Leverage the IoT means in accordance to the Blockchain principles whenever and wherever possible. In Detail: - Receipt of initial non PI order/ Map the data into a PI order (constraints defined if any) - Group the orders in transaction(s) - Compose necessary shipping documents - Pass on constraints to Encapsulation layer - Trigger Blockchain ledger - Receive IoT notices to generate events/ notifications - Manage shipping state of orders Assign Container id and API key linked to the PI order Provide Shipping instructions Acquire IoT data (API key) Provide status of shipment and any deviations from original transport plan (in terms of time, cost etc) Authenticate request Transform received IoT data (API key) to Transport Events Expose PI-Shipment Delays/Incident through communication with IoT Devices or other external systems. Output will be an events report Recalculate and expose as output ETA to next PI Node Provide PoD and payment notification Provide shipment notifications to PI operators, customers and brokers. Output will be details of orders, proof of delivery, date and time and any relevant to the consignment detail. Request services from Logistics web layer (instructions how to proceed), and from the routing layer (updates on the status of the shipment) Instruct the routing layer to re-route or cancel a shipment based on relevant outcomes. Output will be a revised routing plan.

6	The Encapsulation Layer	Stuffing /Unstuffing products to PI containers	The encapsulation service is responsible for the optimal loading of cargo into PI containers - Item encapsulation into handling Containers - Handling Container encapsulation into Transport containers - Shared Transport container encapsulation - Initial picking list generation
7	The Logistics Web Layer	Interface between the Physical Internet and the users of the logistics services	Obtain product characteristics and time specification of order. Input from the ordering party (client) and output a list with characteristics such as weight, dimensions, storage requirements together with the desired time for delivery Provide quotes for shipping goods, times in an optimal scenario. Output will be an offer with best available service to satisfy requirements of the order Obtain shipping rates, capacities, times etc. from PI actors through the output of Supply chain software modules like ERP, WMS etc Devise a transport execution plan and possible additional subcontracts between other involved PI service providers Dynamic (node to node) cost calculation and revenue as well as distribution amongst nodes and, if applicable, penalties based on SLAs and relative agreements. Create a PI Transport contract Create a smart contract (if applicable) Receive and store Transport events in Blockchain Evaluate the overall PI Cost from a start-to-end perspective of the container travelling in the network

3.4.5 Shipping Service Implementation

To clearly indicate the interoperability of the services based on the interaction of the Layers stemming from the ICONET Protocol stack the table below takes a close look to the Shipping Service with the Shipping Layer being the orchestrator of the supply chain operations from start to finish and in close collaboration with all the Layers. Analysis is based on the D2.5 deliverable findings:

Table 3-3 Shipping Service Functionality Inputs/ outputs

Service/Function	**From Service**	**To Service**	**Input**	**Output**

CreateOrder	Logistics Web Layer	Shipping Layer	Order arriving into PI ecosystem	PI enabled Order (uses PI data model)
GroupOrders	Logistics Web Layer	Shipping Layer	Collected PI Orders	Transactions of meaningful and cost-optimal PI Orders
GetDisposalDestination	Logistics Web Layer	Shipping Layer	Compromised PI Order	New Destination to which the shipment is to be diverted
InitializeIoT	Shipping Layer	IoT Cloud Platform	Transaction with container Id's generated by item encapsulation	API Key for IoT platform connectivity
ConfigureIoTTrackers	Shipping Layer	IoT Cloud Platform	API Key for IoT platform connectivity and product/shipment constraints, defined as a PI Order	PI Order updated with the successful or unsuccessful installations of tracking devices
ValidateDestination	Shipping Layer	Network	Destination, as sent by the Web Logistics Layer	PI Node
GetContainers	Shipping Layer	Network	PI Node and constraints (i.e. refrigerated containers required)	List of available containers that fulfil product constraints
RouteOrder	Shipping Layer	Routing	Order with valid origin & destination	Order with optimized routing information using established links/corridors
ExecuteItemEncapsulation	Shipping Layer	Encapsulation	PI transactions & products	Updated PI Order & PI container ids associated with products
GenerateShippingInstructions	Logistics Web Layer	Shipping Layer	Orders that have finished initial composition/decomposition	Orders with corresponding shipping documents

ApproveOrder	Logistics Web Layer	Shipping Layer	PI Order that has been approved by the stakeholders	None
GetIoTData	Shipping Layer	IoT Cloud Platform	Container ID and API Key	Tracking Data (i.e. GPS coordinates of each container and the condition of the alarms configured)

Any or all of the actions described above can be repeated as many times as necessary, pending receival of events such as arrival at PI hub (recalculation of the routing might be needed) or impeachment of order (part of order might need to be rerouted for safe disposal) etc.

3.4.6 Conclusion

D2.5 offers a thorough analysis and description of the services designed in a more technical manner to provide the needed functionality within the scope of the project. The ICONET protocol stack had a pivotal role in manifesting the relationships and boundaries of these components which at the end of the day lie at the heart of the PI network and conception.

A close look on these 'functions' that modern-day Logistics processes require, form an important part of the PI offering to the Supply Chain users. As mentioned earlier a combination of the resulting services for ICONET has been adopted as a better match to the project's needs. In addition to above and as shown in the Functions offered by each Protocol Layerthe exact services or functions per each ICONET layer have been analyzed and identified and have also assisted the architectural design and services implementation.

3.5 The ICONET Protocol Stack and the Living Labs

3.5.1 Introduction

The LLs provide the testing ground of the innovations and capabilities to be offered by the PI concept while at the same time identifying areas needing further study and validation.

With the Use Cases and set up activities completed for the four LLs, deliverable D3.1 has laid out the specific processes and elements required for each LL to compliment the architectural layers and logistics services to ensure interoperability and data flow. The report specifically addressed the protocol stack and layered architecture and its relevance to the deployment of the LLs. More specifically a closer look on the services required by each layer but also the services provided by each layer to layers above or below in the architectural stack. The report will be concluded with D3.2 final version due in February 2021.

One of the core objectives of the LLs has been identified as the need for continuous communication and feedback to/ from WP1 and WP2 to further enhance and improve the design work related to PI framework and platform/ reference architecture.

The preliminary results and tests of the LLs have also helped shape the outcome of a number of deliverable reports. Concepts, elements and principles stemming from the work of the WPs were always seen 'through the LLs lens' for relevance and alignment of efforts with the target to satisfy the LLs objectives.

It is important in the final version of the report, aiming to satisfy requirements laid out by DoA through task T1.6, to have indeed a closer look on the proposed-to-inspire OLI model, the architectural layered stack and its effect on the LLs framework.

The important aspect is the effect and added value of the Protocol Stack to the setup of the use cases in all LLs and specifically how the services would communicate with one another, the standardization of the communication, the role of each service and the overall seamless flow of goods and information throughout the PI Hubs and ultimately the PI network.

The previous version D1.11 detailed the layers for each and every LL and how the reference was drawn to address the LLs' special needs and requirements. For the purpose of the current report the LLs are addressed to illustrate the final approach as far as the functionality is concerned and the resulting services for each layer. A reminder of each LL use case scenarios accommodated by the protocol stack is stated in Annex II for ease of reference.

3.5.2 Latest Developments

The exact findings will of course be listed in the final version of the LLs deliverable reports to be submitted in December 2020 as per GA under the technical description part of each report. On a similar note the deliverable D2.5 submitted in September 2020 has detailed the design considerations and elements of the services in each GPICs modelling component being at the core methodology of the project. To revisit briefly the findings for the purpose of this document below section details the design guidelines and an end-to-end process for the services aspired by the Protocol Stack.

3.5.2.1 PI Hub (LL1)

Shipping service

Shipping Service is mainly responsible for the continuous monitoring of the mode specific traffic systems such as the Rail Traffic System for the Port of Antwerp. It configures also the IoT Devices within PI Containers (and video gates) and exposes all relevant information, via REST APIs, to the rest of the querying services. It continuously updates the position and status of the shipments, exposing this information to the Web Logistics Layer, that in turn can use it to recalculate the ETAs, divert shipments in case of heavy traffic, prioritize wagons based on their expected arrival or departure times. Finally, in a change of transport mode or during loadings/ unloading the service will call the encapsulation layer towards a picking list calculation. State of the order is managed; the next node is obtained in collaboration with routing service and notifies the Web Logistics layer for further actions.

Encapsulation service

The service is concerned with the loading and unloading of train wagons based on the bin packing algorithm, a subject discussed in detail in report D2.14 Intelligent Optimization of PI containers and PI means in PI Nodes submitted in September 2020. The encapsulation layer waits for a call from the Shipping Service to load or unload a list of PI-containers. The call will contain all the necessary parameters and restrictions necessary for the encapsulation. These parameters include: The Identifier of the wagon, its dimensions and information concerning the items to pack. Such information is: the location of origin and destination, the sender, receiver, the transport mode, the type of goods, the identifier of the items to pack their dimensions and their weight. The encapsulation returns to the shipping service the container's identifier, its total weight, all the parameters of the encapsulated items, with their position inside the wagon.

Networking service

The service considers a network that spans from the arrival and departure terminals offered by each mode (sea international and intercontinental trade, road, rail river connections to hinterland). In terms of network representation aggregation, the PI networking service captures port and regional intermodal hubs, but also port services that cargo needs to transverse, such as customs, or port stack capacity. The service allows PI enhanced modal shifts where possible in alignment with the Port's infrastructure improving the overall port management.

3.5.2.2 PI Corridor (LL2)

Shipping service

The Shipping Service design for PI Corridors is focusing on materializing a Tracking Service, with the implementation and installation of IoT Devices (Smart Trackers or Smart Routers) on real-life containers transporting PGBS goods. The Shipping Service, in this Living Lab, translates the Smart Contracts obtained by the Blockchain Ledger into IoT meta-data, configures the sensors in each PI Container and, using the SLAs imposed by the Ledger, monitors and validates their data throughout a shipment's lifecycle.

If an SLA is violated due to delays a re-route is requested from the routing service via an event. Additionally, the Shipping service may request a reallocation of the goods into PI containers and PI movers if need arises. Again, the shipping service as orchestrator collects data and notifies any other service associated with the specific movement.

Encapsulation service

In the case of a multimodal corridor the encapsulation layer will select the optimal number of transport containers to pack the items, those which maximize the utilized space and minimize the number of containers. The algorithm with the parameters and restrictions are the same as in the case of PI Hubs. At the locations where the mode of transport is changed, the Shipping Service contacts the Encapsulation Layer, to unload the items from one mode and load them in transport containers of the other mode. This operation may mean removing all items from a container and reloading them on a train's wagons as in a PI Hub and vice versa. In all the cases the encapsulation service returns to the shipping service all the details of the encapsulation.

Networking Service

The service undertakes the network discovery and shares the information with other services primarily Routing. The links and modes of transport are considered as well as constraints relevant to the desired service for example as set by the LL stakeholder P&G. For intermodal nodes, throughput capacity of intramodality is also considered. To enable synchro-modality and cargo consolidation at nodes, data covering infrastructure properties, status, as well as services schedule and loading status are maintained. Finally, utilization and fill rate metrics are considered impacting to lower emissions and costs.

3.5.2.3 eCommerce (LL3)

Shipping service

The main function of the Service in the case of eCommerce is to track and manage orders that traverse its PI Network. In further detail, the Shipping Service updates the state of orders of goods, providing a real-time overview of the movement of these goods. By utilizing IoT Data from the Cloud Platform it exposes the measurements to the rest of the services (such as the Routing or the Web Logistics) enabling them to re-act or act pro-actively to conditions such as congestions, allowing for a re-route of trucks to take place, thus minimizing the lead times.

The quality of service is validated with the service exposing all necessary data including IoT devices sent data in transportation means to the Web Logistics layer. Moreover, the shipping service exposes needs from a node to another which indirectly leads to reduction of stockouts for the retailer in question.

Encapsulation service

The service deals with the stuffing of trucks and similar to the PI Hub it considers various constraints to produce the best possible scenario. Like all instances the encapsulation service will return the information and details to the Shipping service.

Networking service

In this specific LL the urban deliveries of SONAE dictate a different approach in the sense that the nodes included the product stock level and capacities like picking capacity are also defined in the generic PI nodes. Also the capacity of each regional store to accommodate contingency stock and the capacity of each vehicle to consolidate deliveries, in the sense that it can pick up a contingency stock along the way to satisfy demand at an outlet approaching stock-out. The networking service optimal fulfilment store identification module is also deployed, if only less sophisticated approaches are in place.

Routing service

As a primary objective the service optimizes the best route by combining pickups and deliveries while stock fulfillment and order consolidation is being processed aiming to minimize cost. Thus consolidation of orders minimize costs, journeys are optimized to max fill rate and stores with most items can be considered s depots.

3.5.2.4 Warehouse as a Service (WaaS) (LL4)

Networking service

With rail and road links considered and clients scattered around regions the network details all links considering distances and delivery locations ultimately focusing on the interconnectivity and interoperability of networks within the PI.

The data structure looks at static level (nodes/ hubs' functions, capacity, Links' distances and costs, Movers' frequency, travel time) and dynamic data (queues, weather conditions, congestions, consolidation opportunities). The network density remains a main factor for optimization in relation to customer locations.

Routing service

The service operates between central and regional warehouses, customers and storage hubs with the objective to minimize cost via dynamic deliveries, resource allocation, best route with vehicles visiting warehouses and minimizing empty trips. In that respect vehicle capacity is checked, set of pending orders being identified in that route and best path calculated to cover minimal distance

3.5.3 The ICONET Protocol Stack and the Simulation work

Deliverable D2.17 'Mixed Digital/Physical Simulation Models for PI Networks- Final', has discussed extensively the philosophy and nature of the simulation work to be carried out in all LLs. The purpose of this part of the document is to briefly revisit why Simulation work is vital to the realization of the PI concept and how the analysis of the present report affects the design of the simulation effort.

The simulation models are effectively dynamic software modelling tools which in the LL case, recreate supply chain scenarios to evaluate performance and behaviour of the use cases. The simulation effort itself is an iterative, explorative process which could include calibration/ optimization work for its different parameters based on the outcomes and needed output. In all LLs, models will measure the output performance and ultimately evaluate the PI behaviour, efficiency and impact. These dynamic models will include information from both the digital aspects of the LLs as well as the physical ones i.e. real-life data. This mixed digital/ physical simulation is important because it enhances the representation of the behaviour and interrelationship of various elements and factors necessary to test and validate the PI concept from a more realistic, day-to-day, point of view.

The current deliverable and its findings offer insights to the simulation work in the following areas:

1. A digital simulation model allows creating a representation of the physical world and its behavior in a software model of a computer. As the simulation model is dynamic it evolves over time. The rules of behavior therefore, included in the model, must refer to changes in the states of the processes and the participating elements in each function and in each layer. The complexity and scope of the models must relate closely to the functions identified in each architectural layer.
2. Data collection plays a key role within the simulation, since the data must really emulate the realities of the system at required levels of precision and detail. Therefore, the input and output structure of the information must come from the IT architecture and must reflect the recognized functions in the layered architecture for results' relevance and accuracy.
3. The evolution of the simulation models in new ones in the future as explained in the above mentioned deliverable D2.16, must come from the services and architecture defined in, amongst others, current report and pave the way ahead in an effort to construct and test a realistic behavior of the PI model.

3.5.4 Conclusion

Early on the setting up of the LLs the baseline conditions and the creation of the use cases, the PI Protocol Stack have added significant value to the UCs. In an effort to satisfy business needs and interrelations as well as specific processes between operators, the protocol stack has allowed seamless integration among each layer and the respective services, message standardization between parties, predefined communication channels among the services and clearly defined responsibilities per service.

Thus, the smooth flow of information and goods was based upon a proper structured schema mirroring the T&L operations and specificities with clearly defined inputs and outputs through which the use cases enabled the deployment of container movement throughout the network while at the same time structured the model testbed of the simulation work.

The closer look at each of the LLs reference the inspiring architectural layered stack of Logistics services, offers the opportunity in this final version of the deliverable to analyse two things:

i. To relate the exact use cases set up in each LL to the specific architectural layer and what exactly each layer entails. Testing phases will examine closely the desired interoperability and of course relevance of the architectural framework and the roles of each layer. Also, to closely examine how the different data sources will feed the various interconnected layers and services guarantying uninterrupted flow of physical packets.
ii. To identify areas of further work through simulation and optimization modelling and therefore inform /get informed by work in WP1 and WP2 on how the various elements need to be designed to ensure smooth flow of goods and information in the PI network.

As testing phase 3 is under way and results continue to emerge, the first findings could be listed below:

1. Specific services in the LLs will most likely require more significance to some layers than others. In some cases, they may require unification of the under-study OLI layers to ensure seamless sharing of data and products. Examples of such cases are routing and warehousing services and already discussion earlier on a hybrid version of the ICONET Layers has been explained.
2. The sharing of data between layers (with accompanying network properties enhancement) will lead to specific rules /protocols to apply to make sure information is shared effectively.

The simulation testing and fine tuning of the use cases in all LLs will go beyond the submission date of the current report. The findings will be listed not only in the deliverable reports of the LLs but also in other documents like D3.15 Learning Conclusions to be submitted in February 2021 at project end.

3.6 The ICONET Protocol Stack and the PI Activities as Services

3.6.1 Introduction

As required by the sub task ST 1.6.1 the first version of the current deliverable inspired yet another series of reports, the D2.3 'PI networking, routing, shipping and encapsulation layer algorithms and services v1' under Task T2.2 which concluded and was submitted in September 2020 under D2.5. This document discussed the organizing and configuration of PI services as layered structure of activities. This is done by analysing the purpose and role in PI of the activities and producing algorithms which are to be tested /verified through the LLs' work.

The key services under study in D2.5 were:

- Networking: Creating and evolving the PI network (of networks) through which shipments are routed.
- Shipping: Specifying (instructing) what needs to be shipped, and monitoring and managing the process.
- Encapsulation: Preparing shipments so that they can be shipped via the Physical Internet
- Routing: Routing shipments through the PI network

What is of interest though is the in-depth analysis of the activities themselves as fundamental (to PI) operations which resulted from the OLI model recommendations and the approach behind the proposal of mathematical models to examine and understand how exactly these will function and interconnect in the real world under PI. Simulation and optimization (where applicable) techniques revealed the suitability of different networking elements and policies again proposed based on work under current report and thus ultimately offering a benefit to the design tasks of WP2.

3.6.2 Differences between non-PI and PI Logistics Operations

Briefly revisiting the analysis carried out in the previous version Table 3-6 below shows the differences between non-PI and PI Logistics Operations:

Table 3-4 Differences between non-PI and PI Logistics Operations

	Non-PI operations	PI Operations
1	Single network is used for sending containers to destination, owned by a single carrier.	Multiple carriers will participate
2	Cargo is basically rarely merged/ consolidated (only done within the work of one operator/ freight forwarder). Result being the dead space in containers	PI hubs will consolidate cargos across operators with the aim to increase efficiency and reduce costs while preserving service levels.
3	The existing agreements between supply chain associates limits the services to those channels covered by those agreements	Multiple hub peering agreements will enhance flexibility and options offered while at the same time reduce associated expenses.

The above differences help the understanding of inefficiencies of existing processes and where the focus of the process design and objectives under PI's fundamental operations should be and how these have been accordingly inspired by the PI Protocol stack and networking technologies. Furthermore, focus is also given to the LLs work and testing scenarios.

3.6.3 Sequence diagrams

With the clear definition of the Services offered by each ICONET Protocol Layer the design team of WP2 in collaboration with the LLS under the supervision of the project management team concluded on sequence diagrams which effectively represented the interaction (between services) and sequence of operations to effectively complete the supply Chain activities through the network. The diagram, of which example is shown in figure below (PI Container arriving at a PI Node), was updated many times with new insights and ideas following testing in LLs and design efforts in parallel.

The diagrams which were used in all stages of the LLs technical offerings proved to be valuable means for exactly representing the interrelationships and inputs/ outputs between the services and thus assist in the architectural efforts of WP2 and finalization of the LLs scenarios.

The PI Protocol Stack has effectively driven these dataflow relationships and enabled the possibility for any business scenario to be mirrored simply with the addition of the desired service as per whatever needs could come through.

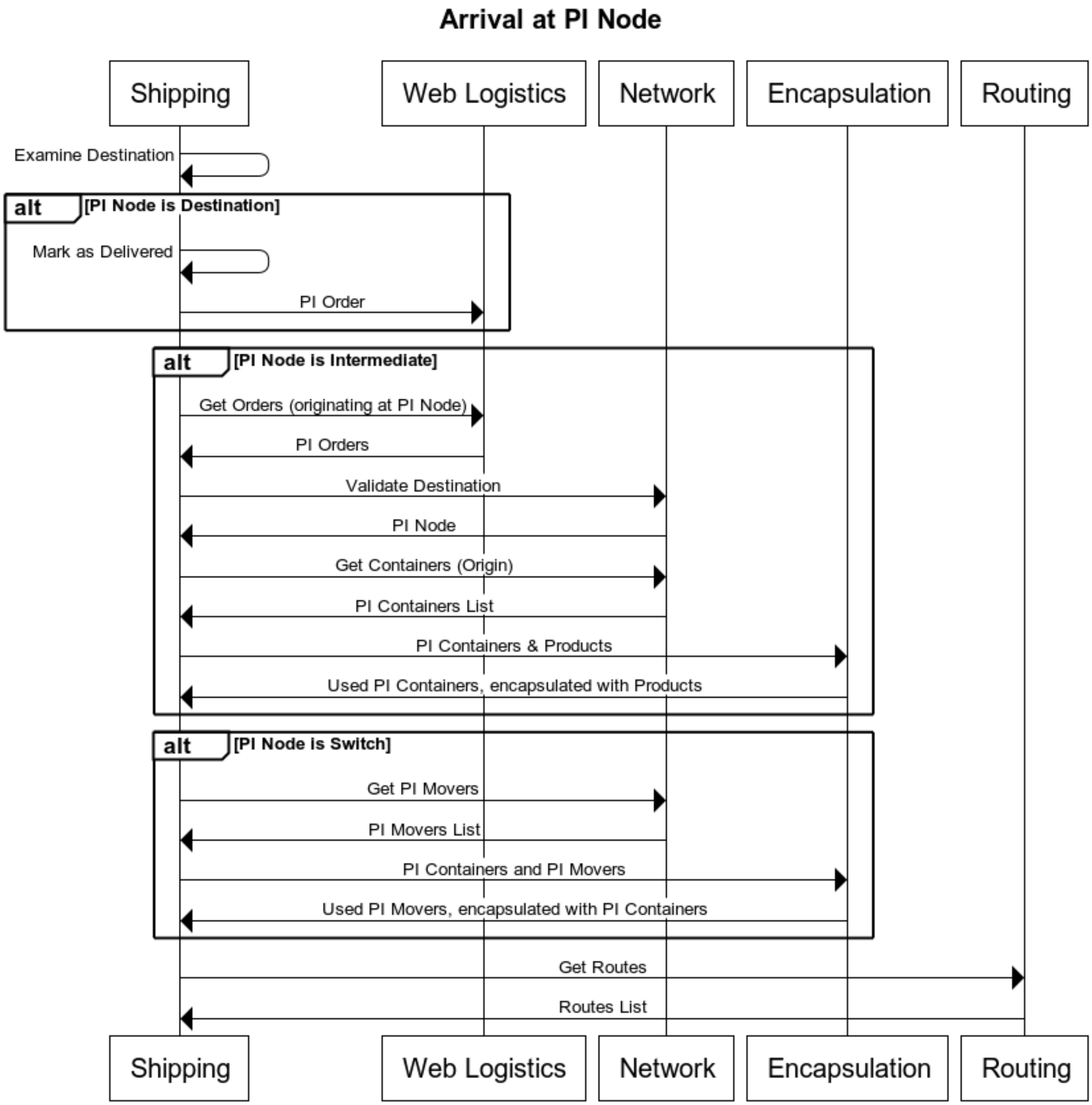

Figure 3-5 Iterative protocol upon PI container arrival at PI Node

3.6.4 Key findings of the analysis and Conclusion

Across the analysed PI activities and adopting the structured approach of the OLI model in its ICONET final form with all its layers requiring specific procedures, operations and technologies the following needs appear to prevail based on the refinements of D2.5 efforts:

i. The key information as input to the various operations should be readily available and accompany the shipment throughout the PI network
ii. There are currently manually performed logistics operations which in the future will need to feed the PI processes within the context of the connecting layers and therefore automation will have to be considered for efficiency. Example is container stuffing.

iii. The dynamics of the PI networks share resemblance to the digital world principles with the established services being enriched perhaps with new, versatile ones which again can rely on the architectural layers to interconnect.

iv. Algorithms like routing algorithms for PI should be implemented in a dynamic and online way similar to the digital internet to reflect the latest changes on topology, time factors and other variables evident in the PI network. The functioning of these protocols will have to be tested even more however early analysis shows relevance.

v. The decentralised nature of decision making at the PI nodes is again surfacing and its merits have already been discussed in a number of deliverables. The applicability of networking technologies suggested by the first 2 versions of this report was studied, to support the above notion of decentralized framework. More analysis in next chapter.

4 The Digital Networking Technologies as PI Enablers

4.1 Introduction

This section briefly summarizes findings in versions 1 and 2 of the report series and then addresses the effects of the analysis on the service tasks including a final discussion. The majority of work done in D2.20 as well as D2.21 for the PI Control and Management Platform represents the majority of the elements in question under the Platform Task requirements and analysis of those reports is quite detailed.

4.2 Background Work in Version 1 (D1.10) and version 2 (D1.11)

The first version of this deliverable has discussed in detail the applicability of existing Digital concepts and technologies to the Physical Internet, in order to gain further inspirations for the design of PI services.

To avoid repetition of the analysis the content is shown in Annex III at the end of the report. For the purpose of this report we draw conclusions from the findings summarised below, if and where required.

Annex III in summary discusses:

1. Connection Oriented and Connectionless Networks
2. High Level Architecture of Internet
3. Routing concept
4. Software Defined Networking (SDN)
5. Properties of Networks

As shown in Figure 4-1 below, the discussion of digital networks was broken down into a number of sub-areas:

- *Network architectures:* in particular the architecture of the (digital) Internet, and how the Physical Internet architecture can map to it.
- *Networking*: What are the building blocks that when connected they form a network. How the building blocks/components of the digital network and their types of connections correspond to the building blocks and connections in the Physical Internet.
- *Routing*: How information flows through the digital network? What are the rules/protocols for routing such information? How the routing concept applies to the flows of physical objects through the Physical Internet?

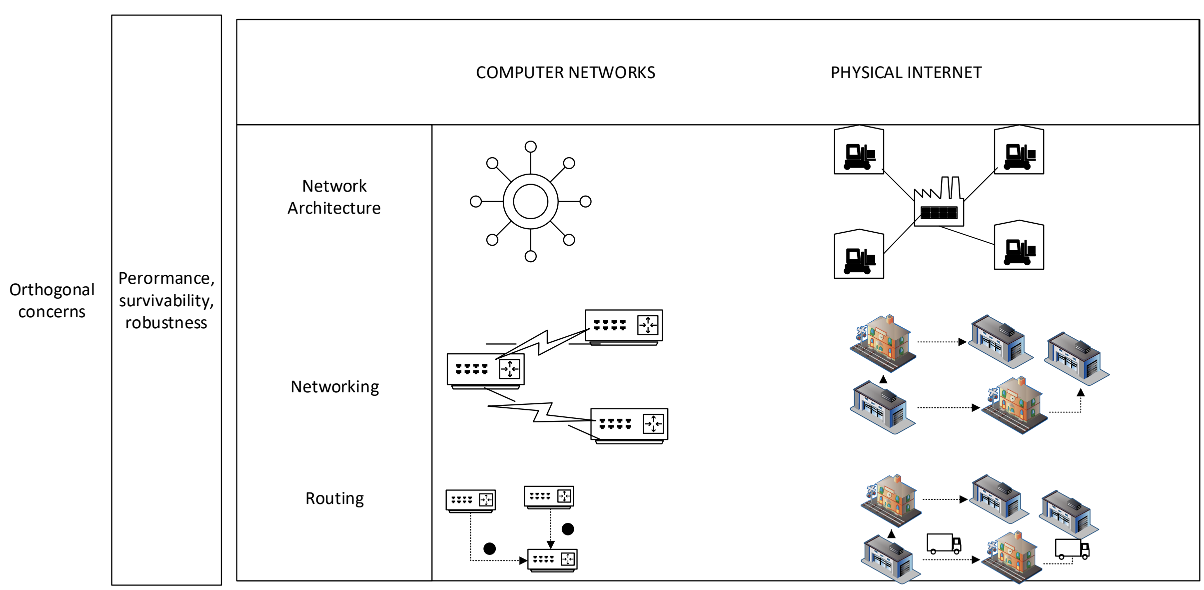

Figure 4-1 Networking concepts covered in version 1

4.2.1 Summary

The results of the analysis of previous versions are briefly shown, by category, below:

4.2.1.1 The IP, TCP and UDP family of protocols and their relevance to PI

The Internet Protocol (IP) [1] (specified in RFC 791), the TCP[2] (specified in RFC 793) and the User Datagram Protocol (UDP) are all protocols in Internet communications and they specifically make sure that data packets are transmitted over networks by performing a number of functions.

In the context of the Physical Internet, the IP and TCP are influencing two distinct components: encapsulation and coordination of shipments.

IP concepts, applied to PI, relate to the use of encapsulation techniques to ensure that shipments are packaged in the right physical format in order to be routable through PI. The ability to assemble and re-assemble shipment loads to match the transport capabilities available across the PI routes is essential. This however implies that the traceability between the original shipment and its different assemblies is maintained end-to-end. Therefore, in additional to the physical format of PI packets (in the IP sense), information must accompany each subassembly, so that at some point, all subassemblies can be assembled into the original shipment. Such information needs also to include origin and destination addresses, where its address can be hierarchical, similarly to IP's ability to address both network and sub-networks in the IP header.

In comparison, TCP protocols can be used to inform the design of the PI protocol regarding the handling of shipments between two PI nodes.

As in the digital Internet, errors and other problems with the physical transport link, means that the wrong (physical) packages may be shipped, or packages may fail to be delivered.

Establishment of control techniques between sender and receiver PI nodes, can result in identification and handling of transmission errors. A packet that was wrongly transmitted, can for example be returned to the sender node, or forwarded to the correct node. Additionally, as in the case of TCP/IP any irregularities can be notified to higher levels in the PI stack, so that the owner of the shipment is aware of any abnormal events and exceptions.

Finally, UDP appears to be less relevant in the context of PI, as lossy transmissions are simply not tolerated, and even if transport speed is important, service quality is essential and receives the highest priority.

4.2.1.2 Routing protocols

The key function of a router is to accept incoming packets and forward them appropriately (e.g. based on information contained in the packet's header). Routers are therefore responsible for discovering appropriate routes through the network. A number of different protocols like Routing Information Protocol (RIP) and Open Shortest Path First (OSPF) and they are analysed in detail in Annex III.

Variants, or specialised areas of the Physical Internet where some of the nodes are mobile can possibly utilise the above routing protocols of ad-hoc and wireless sensor networks, but this area needs to be studied and verified through specific case studies and applications in terms of stability, efficiency and ability of protocols to handle complex networks.

4.2.1.3 Relevance of SDN and NFV to PI

A single SDN controller may control multiple logical networks providing adaptability, error reduction, mobility and enhanced security. In SDN the network is programmable by applications running on top of the SDN controller. SDN introduces some important network abstractions such as the separation of data and control,

[1] https://tools.ietf.org/html/rfc791
[2] https://tools.ietf.org/html/rfc793

where data plane devices become simple packet forwarding devices. This approach could be implemented in the functionality of p-hubs that act in a packet forwarding role, where the routing decision has already been made at a different node or location of the Physical Internet.

Network Function Virtualisation (NFV) is the concept of virtualising network functions in software and running them in virtual machines (VMs) allowing network elements to become independent applications with the ability to increase or decrease their capacity.

Overall, the separation of data from control might contradict some of the principles of PI such as the autonomy of PI containers (i.e. where data and control decisions are decentralized and assigned to the PI container itself, rather than to some central controller) end yet some principles are indeed used through the use of AWS Cloud Platform.

However, work under D2.21 the last report to describe in detail the PI Control and Management Platform, discusses in detail the technology transfer of SDN to cooperative multi-cloud workloads enabling the communication between environments.

4.2.1.4 Relevance of Network Properties to PI

By Network Properties, analysis refers to Failure and survivability, Fault tolerance, Dependability and Robustness.

The above types of properties can apply to all types of networks both data/computer networks and physical networks. Being a packet-oriented network, PI should naturally exhibit some resilience/survivability characteristics. Other network specific resilience properties of PI need to be studied and codified in order for it to meet the expected challenges and industry requirements.

The sections below take a closer look on the design domain of the project and the WP2 efforts and the resulted framework as far as the networking properties are concerned.

4.3 Digital networking technologies and the Service Design Tasks

4.3.1 Introduction

This section discusses how the Service design tasks and work in WP2 have taken onboard recommendations of previous analysis by earlier versions on digital networking technologies. This refers to the final PI Control and Management platform and the final PI Reference Architecture of which deliverable reports were submitted in October and July 2020 respectively.

4.3.2 The Networking Technologies and the PoC Platform

Software Defined Networking SDN

Deliverable D2.19 has studied the elements and possibilities of the SDN supporting the design efforts of WP2 following work in D1.10. In order to support the PI simulation activities, the report discusses the option to reinforce the integration environment with additional open source SDN solutions (probably comprising hybrid elements such as Local Switch Virtualisation, Network Function Virtualisation and Network Overlay) to strengthen the PI concept as and where required. Features like network capacity and independence were key to support the usage of such solutions and below list illustrates the benefits anticipated by using specifically one of them the Network Overlay.

Network Overlay is an implementation of SDN that manages virtual links running over physical infrastructure such as routers and switches. It consists of RON (Resilient Overlay Network) nodes deployed to various locations on the Internet or potentially the ICONET PoC integration platform, which form an application layer overlay that participates in routing packets. Benefits are:

1. Decouples the virtual network from the physical network, which can eliminate certain scaling issues associated with physical infrastructure such as MAC table size restrictions on hardware switches.
2. Decouples virtual node IP addresses from those assigned to the physical network. The abstraction of the physical network provides geographic independence for virtual machines and resources. They can be easily relocated whilst retaining the same configuration settings, because they are mapped by their virtual network ID (VNID) and are not constrained by a physical network ID.
3. Massively increased virtual network capacity. Maximum possible capacity with a physical 12-bit VLAN tag is 4096 virtual networks. The 24-bit tag with a VXLAN based solution allows for a theoretical 16000000 virtual networks
4. Allows for routing based on IP addresses, distributed hashes, XMPP based on endpoint jabber (userid@domain/id), JXTA XML and PUCC P2P.
5. Does not require modification of the underlying physical network infrastructure.
6. Can integrate quickly with other products such as VMware, openstack and docker etc.
7. Can provide for quicker recovery and convergence times after network failures
8. Provides QoS for services and nodes running on the virtual network
9. Optionally route packets over the virtual network links or physical links based on QoS decision algorithms
10. Facilitates manual or automated API programmability for interacting with virtual network and nodes

Report D2.21 submitted end of October 2020, discusses the approach of SDN and how this affected the end design solution. SDN principles have been used in the creation of the PoC environment for the project. And although issues above are still valid (centralized format) still it is based largely on the concept of the Virtual Private Cloud (VPC) which is a private network space that administrators can create to contain whichever resources and assets they wish to deploy. The VPC can be further configured into different zones and virtual networking functions are used to create subnets, switches, routers, NAT gateways, public gateways, DNS servers and so on. VPCs are highly secure and both public and private access routes can be carefully configured to maintain this security

Network Properties

-Resiliency/ Robustness

D2.20 and D2.21 analysis shows that the hosting platform supporting the PoC integration environment needs to be consistently available so as not to impede development towards achieving ICONET deliverables and tasks. A hosting platform that provides redundancy and high availability for power, cooling, storage and network services along with 24x7 datacenter support staff would significantly reduce the risk of unexpected service outage that would negatively impact PI prototype development and project deadlines.

Given the nature of the ICONET project and what it is trying to achieve in terms of PI research and development, flexibility regarding network connectivity and topologies may be important to supporting the PI simulation efforts. Also, different data sources and methods of data ingestion between developed assets, simulation framework, IoT sensors and data analytics may drive complex network topology requirements that the cloud hosting platform potentially needs to accommodate as the project progresses.

-Security and Data Protection

To protect project confidentiality and the integrity of any intellectual property created during the course of the project, adequate levels of security and compliance are required in terms of firewalls, user access controls – both logical and physical, encryption, secure remote access and managed services to industry standard compliance levels. Rather than spend a lot of time setting up such infrastructure and security configurations, which would also divert focus from the core aims of ICONET, it is more efficient to avail of the inherent security aspects that comes standard with a commercial cloud provider. Additionally, these measures protect against inadvertent downtime that could be caused by malware or denial of service attacks etc

The latest D2.20 And D2.21 reports on the Control and Management platform discuss in depth the recommended strategies and of course the importance of the above elements for a reliable and business-attractive network. To discuss in detail the said subject would be out of the scope of this report and above reports provide a clear insight to the issues. The network properties have indeed inspired security issues like all networks must do and the deliverables of WP2 discuss the subnets employed and layout of virtual private cloud networks which are allowed in the employed AWS.

4.3.3 The Networking Technologies and the Reference Architecture

The deliverable D2.2 'PI Reference Architecture Final version' has also studied the analysis of the PI enabling Networking Technologies. The work focused on identifying the key requirements, events and data required through the use of a scenario. It also attempted to identify existing legacy systems and the data that are needed to enable PI operations. The data specifications stem from the findings of deliverables in WP1. This deliverable also documented the preliminary service requirements, defining required inputs and expected outputs as well as dependencies between services.

Last D2.2 offered the final reference architecture, taking into account the areas discussed in sections above in conjunction with work in WP2 also covering Living Lab's cases and expanding on the data specifications as they emerge in the context of ICONET.

More specifically, the Reference Architecture analysis has considered PI Data adapters at the integration points between external systems and the PI service stack to ensure data security while transferring of data takes place. These adapters can come in the form of secured communication protocols such as HTTPS or anonymization protocols in case of personal data need processing for any PI operations. As a result, interoperability and secure integration of legacy data within the PI network is ensured along the lines of DI principles outlined above.

4.3.4 Conclusion

The PI protocols preserve many characteristics of their digital internet counterparts like for example the routing protocols to be employed to route packets through the network. Having said that care was taken as far as special constraints existing in PI like availability of transport modes (needing approval by destination mode to ensure balanced loads and avoid congestions). Deliverable D2.5 has also discussed the routing protocols in particular and findings reveal the strong relevance of some protocols, namely RIP, to the PI routing. It concludes that routing in PI follows practices used by large Internet networks in the sense that to maintain quality of service it uses multiple routes, caching of data and similar techniques. Of course, this requires coordination between PI nodes and a multistep route planning.

Similar to the DI, the PI can employ a similar architecture with smaller networks in autonomous form (PI hubs) connecting to each other through gateways and forwarding physical goods from origin to destination. Work in WP2 points towards this direction justifying the need for decentralized control of each PI Hub/ Node and within the desired parameters of the project.

Physical properties are important to any kind of network for resilience, robustness and fault tolerance. Likewise, any such characteristics in the DI will be shared with the PI assuming (and eventually examining) that same principles apply. Example is the possibility of a link failure (for example a road connecting two links is closed for works) so underlying support network is important. It was concluded that again more work is needed to research specific needs. The ability of the PI network to maintain its structure and functionality in the face of external perturbations or failures must remain crucial to the architectural design tasks.

The very essence of the Protocol Stack and Networking technologies series of reports was to investigate and elaborate on how these DI elements can inspire the PI network. Indeed, documents D2.2, D2.20 and D2.21 under WP2 design work repeatedly discuss the analogies drawn and conclusions aiming at a robust and resilient environment which will support functional and non-functional requirements. The sections above have addressed this in detail.

The final design elements of the Management Platform as this is described in D2.21 submitted in October 2020.

5 Final Conclusions

D1.10 set the foundation for addressing T1.6 requirements, with primary goal to feed and support all PI Services Design activities under Work Package 2. The central focus was on the information and control content of services, as opposed to detailed implementation aspects. The current document is the final of the series and while briefly revisiting the work of D1.10 and D1.11 it lays out the characteristics of elements and components of the Digital Internet to have inspired WP2 Architectural and integration environment investigation.

The report continues to examine how these elements were actually reviewed by the design tasks of WP2 by referencing the relevant deliverables and critically discussing the framework which resulted. Furthermore, the work by the other WPs also affected the analysis in this document. More importantly however this report analyses and displays the exact services or functions offered by each layer in an effort to support the architectural endeavors for an efficient and sustainable supply Chain service as well as contributing to other important PI components like the simulation effort. It also emphasizes the value adding attribute of the Protocol Stack to the designing of the PI services and the testing work in WP3.

The report also addresses the principal considerations of the said technologies and reference models through the lens of the Living Labs in WP3. It considers the effect on the set up of the LLs, the related simulation work and of course the aspects that prevail in terms of services provided and data collection.

Finally, in each section there is a discussion with the findings of each section for a more organized approach.

The main outcome across the sections is the identification of the functions through and between the architectural layer of services and usefulness in future PI work but also the need for further analysis as work in progress will reveal perhaps new stipulations and requisites. The conceptual form of the reference models and early stages of the PI network design require the continuous study of the various principles in search of the acceptable format. It may be proven that new methods of networking and routing are more suitable in the various domains and they are the ones to be followed. Or as it was finally shown a hybrid version of existing components already discussed in previous and current version needs to be further dissected to offer new highways of information and transferability of knowledge from DI to PI.

The components finally adopted in the realization of the PI network operational framework offering evidence of the appropriateness and operational suitability of the various elements have been discussed in detail. The final report on PI Protocol Stack and Networking Technologies offers an informed inside to the research carried out as to how existing technologies and analogies can inspire the PI of tomorrow.

The Physical Internet envisions the Supply Chain of tomorrow in a sustainable, cost efficient and interoperable network of networks inspired by the Digital Internet elements and analogies. In a fragmented and complex industry of Transportation and Logistics, which remains a key contributor to macroeconomic developments, recent globalization, technological advancements and data digitization, the Physical Internet promises a novel, technologically advanced and transparent cooperative framework offering benefits to all operators and users. The work undertaken to investigate the analogies of the DI elements to the novel concept of the PI has proved that there is common ground and inspiration for the PI network.

6 References

[1] B. Montreuil, E. Ballot, F. Fontane. *An Open Logistics Interconnection model for the Physical Internet.* Proceedings of the 14th IFAC Symposium on Information Control Problems in Manufacturing Bucharest, Romania, May 23-25. 2012. Available: https://www.sciencedirect.com/science/article/pii/S1474667016331718 [Accessed 21 Nov 2018].

[2] J. Colin, H Mathieu, M Nakechbandi. *A proposal for an open logistics interconnection reference model for a Physical Internet.* 2016 3rd International Conference on Logistics Operations Management. 2016. Available: https://ieeexplore.ieee.org/document/7731719 [Accessed 21 Nov 2018].

[3] B. Groothedde. Collaborative logistics and transportation networks: a modeling approach to hub network design. Delft : TRAIL Research School. 2005. Available: http://www.worldcat.org/title/collaborative-logistics-and-transportation-networks-a-modeling-approach-to-hub-network-design/oclc/839641250 [Accessed 21 Nov 2018].

[4] S. Pan, E. Ballot, G. Huang and B. Montreuil, Physical Internet and interconnected logistics services: research and applications. Research and applications, International Journal of Production Research, 55:9, 2603-2609. 2017. Available: https://www.tandfonline.com/doi/pdf/10.1080/00207543.2017.1302620 [Accessed 21 Nov 2018].

[5] European Commission. *GRANT AGREEMENT NUMBER — 769119 — ICONET.* Research and Innovation action. 2017. Available: https://inlecom.teamwork.com/#files/5750386 [Accessed 21 Nov 2018].

[6] European Commission. Cordis. Final Report Summary – MODULUSHCA. Grant agreement ID: 314468. 2012. Available: https://cordis.europa.eu/project/rcn/105754/reporting/en [Accessed 5 Feb 2019].

[7] A. Badzar. Blockchain for securing sustainable transport contracts and supply chain transparency - An explorative study of blockchain technology in logistics. Department of Service Management and Service Studies. 2016. Available: https://lup.lub.lu.se/student-papers/search/publication/8880383 [Accessed 5 Feb 2019].

[8] X. Ya. et al. Heuristics for Container Relocation Problem. Journal of System Simulation 2008-14. Available: http://en.cnki.com.cn/Article_en/CJFDTOTAL-XTFZ200814013.htm [Accessed 5 Feb 2019].

[9] T. Roughgarden. An Algorithmic Game Theory Primer. An Algorithmic Game Theory Primer - Stanford CS Theory. 2008. Available: https://theory.stanford.edu/~tim/papers/tcs08.pdf [Accessed 5 Feb 2019].

[10] E. Maskin. Mechanism Design: How to Implement Social Goals. Les Prix Nobel 2007. 2008 :296-307. 2008. Available: https://scholar.harvard.edu/maskin/publications/mechanism-design-how-implement-social-goals [Accessed 6 Feb 2019].

[11] C. Dong and R. Franklin. From the Digital Internet to the Physical Internet: A conceptual framework with a simple network model. IPIC 2018 5th International Physical Internet Conference. Bringing Physical Internet to life. 2018. Available: https://www.pi.events/sites/default/files/IPIC%20proceedings%20-%20online.pdf [Accessed 6 Feb 2019].

[12] H. Zimmermann. OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection. IEEE Transactions on Communications (Volume: 28 , Issue: 4 , April 1980). 1980. Available: https://ieeexplore.ieee.org/abstract/document/1094702 [Accessed 11 Feb 2019].

[13] G. Bora et al. OSI reference model: An overview. International Journal of Computer Trends and Technology (IJCTT). 2014 Jan;7(4):214-8. 2014. Available: http://ijcttjournal.org/Volume7/number-4/IJCTT-V7P151.pdf [Accessed 11 Feb 2019].

[14] Giancarlo Tretola , Viviana Verdino and Dario Biggi. A Common Data Model for the Physical Internet. Mines Paris Tech, 6-8 July 2015.

[15] GS1 https://www.gs1.org/about

[16] Montreuil B. (2017). Omnichannel Business-to-Consumer Logistics and Supply Chains: Towards Hyperconnected Networks and Facilities, Progress in Material Handling Research Vol. 14, Ed. K. Ellis et al., MHI, Charlotte, NC, USA. Universal Business Language v2.0 Standard, OASIS. 12 December 2006.

[17] Franklin, R. Presentation on Using the Five Layer Internet Protocol Stack to organize thinking about research areas for the Physical Internet. The Kuhne Logistics University.

[18] Stallings, W. Foundations of Modern Networking SDN, NFV, QoE, IoT, and Cloud. Pearson 2016.

[19] Hedrick C. Routing Information Protocol Request for Comments: 1058. Rutgers University. June 1988

[20] IEEE 802.1D-2004 - IEEE Standard for Local and metropolitan area networks.

[21] Singh Heena, Monika Agrawal, Nidhi Gour, Naveen Hemrajani. A Review on Routing Protocols in Wireless Sensor Networks. International Journal of Engineering Research & Technology (IJERT) Vol. 3 Issue 1, January – 2014.

[22] Rekhter Y. et al. A Border Gateway Protocol 4 (BGP-4) IETF RFC 1654, 1994.

[23] Gautam Khetrapal & Saurabh Kumar Sharma. Demystifying Routing Services In Software-Defined Networking. Aricent, 2019

Annex I: The OLI Protocol Stack

Background and Fundamental concepts of OLI

The architecture of the Physical Internet (PI) can be defined as a layered stack of protocols that PI shipping/destination points and intermediate PI nodes, must implement, to make it possible for PI containers to flow within the PI network. The Open Logistics Interconnection (OLI) model proposed by Montreuil et al in 2012 [1], was inspired by the Open Systems Interconnection (OSI) reference model and comprises 7 layers. However, according to its authors, the OLI model at its current state is abstract, and requires further analysis and description of the functionality and services offered by each protocol layer.

To understand the OLI layers, their roles and interdependencies in PI, we need to understand the typical PI process. Let's assume a push/demand driven process where a shipper wants to ship goods via the Physical Internet (PI-network). We start with the type of the shipper's consignment and how it can be repackaged in a suitable format in order to be transported via PI. A consignment is defined as a separately identifiable collection of goods items (available to be) transported from one consignor (a shipper for example like a manufacturer) to one consignee (a customer such as a retailer), via one or more modes of transport. According to this view, the shipper's consignment will first arrive at some PI spoke. A PI spoke is the location with the handling and storage capacity where goods are prepared in order to be shipped via PI. The consignment will typically be arrived in conventional packaging such as (non-PI) containers, boxes, crates etc.

In the PI spoke, the original container units will be stripped, and items are repackaged in suitable p-containers with the goal to reduce empty space. The original consignment is transformed to a number of π -consignments. In logistics nomenclature, a *logistic unit*, is any combination of trade items packaged together for storage and/or transport purposes; for example, a case, pallet or parcel. In the context of the Physical Internet we define accordingly a *π- logistics unit* (*π-unit* for short) a combination of individual cargo items (pallets, boxes, etc.) into a single loading unit that can be handled and transported easily by the π-network infrastructure. An important property of π-*units* is that they can be packed tightly into π-containers. *π-units* come in several sizes, but they all have modularity as common allowing them to be combined tightly together to reduce dead space inside π-containers as illustrated in the figure below.

The original consignment number is mapped to the equivalent *π* consignment numbers. The shipper its agent or the logistics service provider are given the new consignment numbers that will represent the tracking numbers to trace the movements of the consignment items through the PI. Once the goods are packaged inside a *π-unit* object they remain there until the last node they will travel to in the Pi network. Thus, the tracking of the consignment object becomes the same as the unique number of the *π* - unit. Uniquely identifying *π* -units allows to trace them throughout their journey through the *π* -network. A unique reference to the such as GS1's UINN [15] can be assigned to *π*-units.

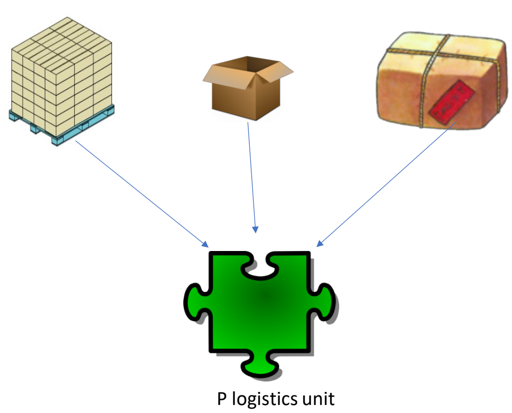

Figure 0-1 Typical logistics units and their mappings to π-units

The Importance of π -units from the perspective of the shipper, the freight forwarder and the carrier.

Shippers that use PI do not necessarily employ PI-compatible packaging for their products. It is the task of another party, such as a freight forwarder, 3PL, or another agent, to repackage the shipper's logistics units in a physical format that can be efficiently transported in the π -network.

The repackaging operation can take several different forms, depending on how the product was originally packaged by the manufacturer, the type and purpose of the packaging (i.e. reusable/resalable or disposable) and several other factors. So, a product may be removed from the original box and packaged in a suitable π-unit of the correct type and sizing, or the product in its original packaging may be inserted in a suitable π-unit. How this is accomplished depends on the physical properties of the logistics units such as volume, weight, type of content etc. The outcome of the repackaging is that the shipper or its agent (a freight forwarder, 3PL etc.) has now logistics units that are compatible with the π-network. The FF/consolidator may combine shipments of multiple customers into a single π-container (Figure 0-2). This is not different from the current Less than Truckload (LTL) practices as it is explained below.

Figure 0-2 π-units filling up a π-container

Mapping Shipments To π -Units

One important property of this approach is that the contents of the π-unit do not change from the moment the π-unit enters the π-network and until the moment it exits it. Although a π-unit will be potentially unbundled and re-bundled several times to form π -containers during its transit, their contents stay unchanged. This happens in order to maintain traceability of the original shipment throughout its movement.

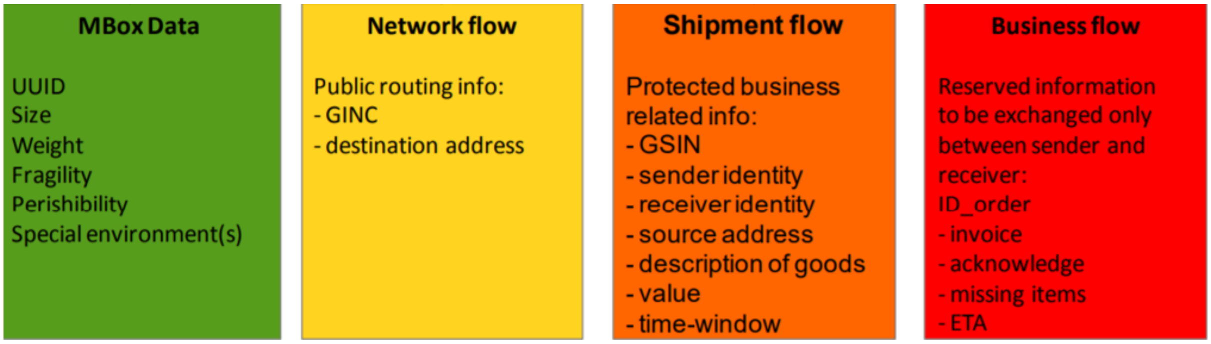

Figure 0-3 Data models required for the different OLI Layers (from Tretola et al, 2015 [14])

As Figure 0-3 illustrates, different types of data and information models are required to describe the shipment from the moment of the shipping instruction to the time it is delivered to the ultimate recipient/customer. This information viewpoints correspond to different stakeholders with different requirements and responsibilities for handling the shipment. Some information for example such as the senders' and receivers' identities may be restricted while other information such as the destination address within the PI network can be more openly shared.

Transporting Cargo in the π-Network

Moving π-units along the π-network is similar to what happens currently in the less than Truckload (LTL) transport business. A "hub and spoke" type network connects small local terminals (the 'spokes'), to larger more central terminals ('hubs' -also called Distribution Centers/DCs). Spoke terminals (operated by FFs/LSPs or similar business operations collect local freight from shippers and consolidate that freight for transporting to the delivering or hub terminal, where the freight will be further sorted and consolidated for additional transporting (known as *line hauling*).

Similarly, π-containers are transported from local terminals (spokes) to the nearest π-hub. The π-units making up a π-container may be heading for different destinations (although they are all packaged in a single π-container for efficiency purposes). Thus, upon arrival at the π-hub they need to be again re-consolidated: disassembled from the original π-container and re-assembled (together with other π-units) into new π-containers.

Here, the type of transport between (major) π-hubs is typically different from the type of transport between a spoke and its hub. Services must be able to carry a higher volume of cargo (π-containers) to multiple destinations, as typically, a π-hub will connect to several other π-hubs. Services may be less frequent but regular and offering higher capacity (the ability to carry many π-containers) compared for example with a single π-container carried by a single truck.

The lifecycle of a π-unit begins at the moment a shipper or other party bundles its shipment into one or several π-units and ends when the π-unit reaches the end of its journey through the π-network and its contents get unbundled. In between the π-unit becomes bundled and unbundled potentially several times in (one or more) π-containers and travels between at least two π-hubs by one or more transport services.

The procedures, mechanisms, rules and data that help to get this accomplished are defined in the Open Logistics Internet (OLI) model's layers, in line of the above are discussed in the following sections.

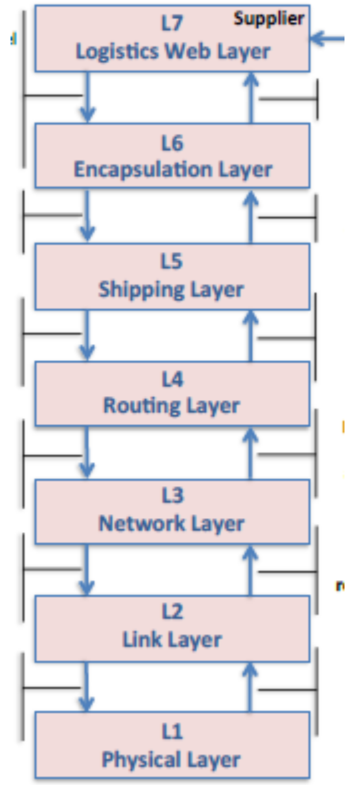

Figure 0-4 OLI layers from Montreuil et al 2012 [1]

Open Logistics Interconnection (OLI)

The fact that logistics networks are often dedicated to single companies or specific markets [1], causes fragmentation which raises logistics costs [3]. Nevertheless, fragmentation can be positively related with service quality, since fragmentation can be the result of smaller, customised shipments [3]. Therefore, there is a trade-off between quality and costs [3]. Some of the aims of the design and application of Physical Internet is to diminish both downsides of the aforementioned trade off by combining digital transportation networks.

The Open Logistics Interconnection Model [1] was conceptualised to enable seamless universal interconnectivity by borrowing analogies from a computer networks standard (OSI) [1]. According to the authors of [1], the analogy between logistics and ICT was that both involve networks, are heterogeneous and in addition to data, logistics networks also include physical goods. Thus, are information driven. Logistics however, involves the movement of physical goods, not only data packets.

In [1], Montreuil, Ballot and Fontane proposed to adopt a seven-stage layered structure, that would be in-line with the OSI with the prospect of refining them in the future or even potentially unifying some layers, similarly to the evolution of the TCP/IP model. The TCP-IP model shares, or better said, has some layers that correspond to the OSI model such as the Physical, Data Link, Network, Transport Layer. On the other hand, OSI's Session, Presentation and Application layers are all encompassed in the TCP/IP's Application layer.

Communication Layers

According to the initial proposal of Montreuil, Ballot and Fontane [1] the OLI consists of the following layers:

1) Physical Layer

   This layer handles the operations related to the Physical Internet. With the purpose of optimising logistics networks, the Physical Internet connects different private and public heterogeneous logistics networks [4]. Thus, the Physical Layer includes the PI means (vehicles, conveyors, etc) that transport and stock PI containers [2]. It “specifies the layouts and relative positioning of entry and exit points, gripping mechanisms and interlocking mechanisms. It monitors the π- means, aiming to detect and correct their physical dysfunctions such as the loss of integrity of a π -container having been dropped, unsealed without client agreement, or whose temperature control is malfunctioning.” [1, p4]. For the aforementioned reason, RFID was proposed as a solution for information exchange.

2) The Link Layer

This layer handles node to node transfer. It is responsible for monitoring and correcting errors that happen at the physical layer. “This is being done by checking consistency between physical operations and their specifications.” [2, p.3]. For example, the road is blocked, conveyor is malfunctioning, PI container is lost or damaged [1]. The link layer attempts pre-emptively protect or take corrective measures against dysfunctions. This is a crucial layer because in the digital word it is relatively easy to pre-emptively protect against dysfunctions, and in extent, to recover from errors. If an information packet is lost in the it can be easily, promptly, and cost-effectively be resent. In contract the aforementioned do not apply when a physical container is lost, thus many standards shall be defined and applied.

3) The Network Layer

The Network Layer deals with the interconnectivity, integrity and interoperability of networks within the Physical Internet. It is responsible for providing the means of routing the PI containers across the network(s). It provides a quality of service that is requested by the Routing Layer. Within this layer, the composition of the PI containers is defined [1], [2].

4) The Routing Layer

The routing layer is in charge of routing the PI containers from starting point to their destination. It attempts to achieve this in the more efficient and reliable manner possible. It manages inter-node transport and handling services to the upper layers while factoring their specifications (e.g. environmental, economic, service priority) [1]. This layer defines and controls the PI routing protocols. “It monitors the status and service capability, capacity and performance of all π-means within each π -network” [1, p.331].

5) The Shipping Layer

The Shipping layer enables the efficient and reliable shipping of PI containers but providing the functional and procedural means. It is responsible for all administrative aspects during the shipping process, including delivery acknowledgement [2]. It establishes the type of service that is used, such as normal, express. It defines the protocols for monitoring, verifying, adjourning, terminating and diversion of shipments [1]. “It gets shipping requests from the deployment layer and it requires transport services for its shipments from the transport layer” [1, p.331].

6) The Encapsulation Layer

The Encapsulation or Deployment Layer links products to PI containers. It handles moving and storing products in PI containers and monitors and validates the properties of PI nodes and PI means (such as capacities and performance) [1], [2].

7) The Logistics Web Layer

This layer provides the interface between the Physical Internet and the users of the logistics services [1]. It provides the necessary applications to the users in order to utilise the Physical Internet [2]. “It monitors contracts, stocks, flows, service provider capabilities, capacities and performances by exploiting an informational synchronization with the encapsulation layer" [1, p.331]. Supply chain, logistics, operations and enterprise resource management software operate within this layer.

New Open Logistics Interconnection (NOLI)

The OLI model [1] was further refined by Colin, Mathieu and Nakechbandi [2], as the NOLI model. NOLI adjusted the seven layers of OLI, as in Table 0-1 Comparison between the layers of the TCP/IP, OSI, OLI and NOLI Models [2, p.6]. The proposed layers of NOLI are presented below in more detail:

Table 0-1 Comparison between the layers of the TCP/IP, OSI, OLI and NOLI Models [2, p.6]

<table>
<tr><th>TCP/IP
Layer Name
(Internet)</th><th>OSI Reference Model
Layer Name</th><th>OLI
Layer Name [1]</th><th>NOLI
Layer Name [2]</th></tr>
<tr><td rowspan="3">Application</td><td>7. Application</td><td>7. Logistics Web</td><td>7. Product</td></tr>
<tr><td>6. Presentation</td><td>6. Encapsulation</td><td>6. Container</td></tr>
<tr><td>5. Session</td><td rowspan="2">5. Shipping</td><td>5. Order</td></tr>
<tr><td>Transport</td><td>4. Transport</td><td>4. Transport</td></tr>
<tr><td rowspan="2">Network</td><td rowspan="2">3. Network</td><td>4. Routing</td><td rowspan="2">3. Network</td></tr>
<tr><td>3. Network</td></tr>
<tr><td>Network Access</td><td>2. Data Link</td><td>2. Link</td><td>2. Link</td></tr>
<tr><td>Physical</td><td>1. Physical</td><td>1. Physical</td><td>1. Physical Handling</td></tr>
</table>

Communication Layers

1) The Physical Handling Layer

The Physical Handling Layer defined the characteristic of the PI means that physically move the PI containers (e.g. ships trucks, cranes, conveyors) [2].

- It manages the states and locations of the PI means (e.g. availability of cranes, trucks, conveyors) and of the PI containers (waiting, carried, etc).
- It receives shipments of PI containers and the identification of the π-mean allocated to each shipment, from the Link layer.
- It schedules the arrangement of PI containers to PI means. For example, ensuring that a conveyor is within the maximum weight it can hold.

- Instructs the PI means.
- Signals PI means problems to the link layer. [2]

2) The Link Layer [2]

This layer "manages the individual steps of movements of π-containers on π-means" [2, p4]. Any point to point movement is considered to be a step. The Network Layer sends blocks to the Link Layer with their start and ending location. The Link Layer, divides and combines blocks accordingly, and allocates a PI mean for shipment for this step. This can also be a virtual move instead of a physical suck as the handling of a block from one operator to another.

3) The Network Layer [2]

The Network Layer receives loads of π-containers from the Transport Layer, with an initial starting and a final ending location for each load. The Network Layer divides and/or combines the received loads into "blocks". The Network Layer computes and manages the routing of each block from its initial starting location to its final ending location. The Network Layer manages and maintains the data structures necessary to compute the best paths for the blocks.

4) The Transport Layer [2]

The Transport Layer receives orders made of π-containers from the Order Layer, with an initial starting and a final ending location for each order. The Transport Layer divides and/or combines the received orders into "loads". The Transport Layer manages the end-to-end trip of each load from its initial starting location to its final ending location. It checks that the final ending location can handle a load shipped there. It signals to the Order Layer the initial departure, the current location and the final arrival of each π-container. The Transport Layer ensures that deadlines are respected.

5) The Order Layer [2]

The Order Layer receives sets of π-containers from the Container Layer, with an initial starting and a final ending location for each set. The Order Layer establishes the "dispatch note" associated to each π-container of each set. It also records priorities and deadlines of π-containers. The Order Layer divides and/or combines the sets into "orders" (according to deadlines, characteristics of π-containers, clients wish such as sub-orders, etc.). The Order Layer checks the possible problems (for example, does the final ending location accepts dangerous material? etc.) The Order Layer manages transactions. They can be simple complete orders, or more complex ones, such as sub-orders that may trigger intermediate payments if completed, etc. It signals damages to, or loss of, π-containers to the above Container Layer, and also received π-containers with no known consignor nor consignee.

6) The Container Layer [2]

The Container Layer defines the physical characteristics of the π-containers allowed on the Logistics Network. The Container Layer receives π-containers from the Product Layer, with contracts information.

The Container Layer checks the physical integrity of received π-containers, and of the goods inside. The Container Layer combines the received π-containers into "sets". It also covers specialized nodes for the management of π-containers (empty containers, damaged containers testing, specialized containers maintenance). Finally, it manages received π-containers with no known consignor nor consignee.

7) The Product Layer [2]

The Product Layer defines the possible products or goods that can be transported inside a π-container by the Physical Internet, and their characteristics. The Product Layer fills empty π-container with the products. It establishes the contract for each filled π-container, and gives the filled π-containers and their contracts to the Container Layer. It receives filled π-containers

The Layered Protocol Analogy of the Internet and PI

Rod Franklin [17], proposed the idea of using the Internet Protocol Stack as the basis for the PI architecture. The Internet Protocol Stack consists of five layers and is used for message transmission over the Internet. The analogies of the five layers between the Internet and the PI are analysed bellow:

i. The Application Layer – this layer is where goods to be shipped are prepared for shipment and human readable information about the goods is created. [17]

   - It is at this layer of the PI that all data relevant for ensuring that the shipment arrives at its final destination, is handled per its quality of service requirements, and that its general cost structure is encoded (this is the information that is “read” at each node through which the shipment moves so that its movement can be controlled)
   - As with the Internet, this packet of information and physical goods (the shipment) is our “message”

ii. The Transport Layer – at the transport layer shipments are broken up into sizes that are transportable by standard sized containers or the selected means of transport. [17]

   - In addition, the transport layer provides services that ensure delivery of the shipment and manage flows between the sending location and destination
      - These services include tracking, forwarding, cost accounting, and reporting services among others
   - The standard loads that are shipped out from the transport layer are our “segments”

iii. The Internet Layer – this layer takes the “segments” constructed in the transport layer and manages all services required to deliver these “segments” to their destination. [17]

   - This layer defines how all nodes between source and destination will respond to information contained in the “datagram” that it constructs to move the “segment” (shipment) from source to destination

iv. The Link Layer – the link layer takes the “datagram” from the Internet layer and passes it from the current node to the next node in the network. [17]

- The services that the link layer provides depends on the mode of transport between nodes
- The encapsulated “datagram,” which includes all information on how the transport mode is to handle the shipment, is called a “frame”
- All QoS, cost, etc. necessary for the transport of the shipment is provided to the transport means through the services of this layer

v. The Physical Layer – this layer of the Physical Internet actually moves the “bits” of a shipment between the linked nodes. [17]

- The services provided are both link and mode dependent and depend heavily on mode, carrier, regulatory bodies, etc.
- This is the layer that includes roadways, rail operations, rivers, sea and air lanes

Analysis of information entities and flows in the OLI model

To understand the lifecycle of a π-unit and how the π-network supports its movements through it, is useful to rearrange the layer of the OLI model (Fig. 4) to a more lifecycle focused view as illustrated in Figure 0-5 OLI layers re-aligned.

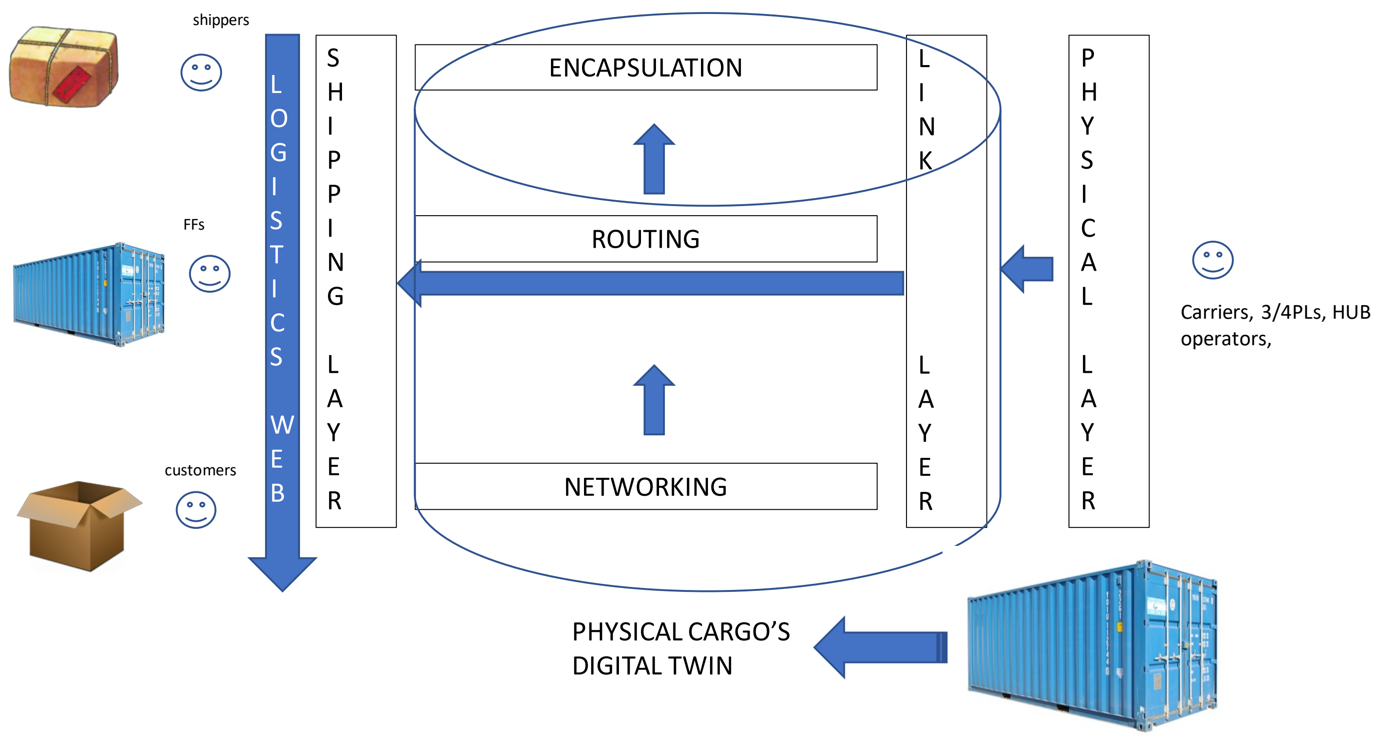

Figure 0-5 OLI layers re-aligned

Figure 0-5 OLI layers re-alignedreads from left to right shows the different types of PI actors and the PI systems corresponding to the OLI layer. In the middle of the diagram are systems that allow the synchronisation between business logistics operations and PI related functions. This is achieved through ‘digital twins’ of the Physical Internet entities (shipments, equipment, hubs, etc.) that help to keep in sync the customer side of PI (shippers, consignees and their agents) with the operators of the PI (carriers, π -hubs and other PI related actors).

Logistics Web Layer

Overview

According to OLI, the Logistics Web Layer monitors and validates the capabilities, capacities, prices and performances of π -nodes and π -means of π -service providers, as well as the status of signed contracts and of deployed π -containers. At this layer reside certain current EDI operations. In the context of this report we interpret the above as follows:

A shipper or its agent will be primarily concerned about the cost and service guarantees of shipping its products via the PI. The Freight Forwarder – assuming that the shipper does not enter in contract directly with the PI hubs and carriers- i.e. it acts as a broker itself- is the intermediary between the shipper and the PI. An FF as in its traditional (non-PI) role is able to obtain wholesale freight prices by negotiating with the PI service providers (carriers, hubs, possibly other types of actors that run services on the various PI subnetworks/segments.

In general, the shipper will be seeking a firm price quote for the entire transport through the PI, rather than separate prices for each transport segment. Possibly it will want to obtain different price quotes for different service options (for different transport means and routes through the PI, or for different service levels such as 'economy', 'express' etc.). This is what the OLI paper refers to as 'capabilities, capacities, prices and performances of π -nodes and π-means'.

Once the shipper selects the transport option that is optimal under the shipper's criteria (e.g. price to service ratio), a contract is signed between the shipper and the PI broker (e.g. the FF). From then on, a Transport Contract and a Transport Execution Plan are established. Sample Transport Execution Plan in UBL 2.0 format is shown in D1.10. The transport execution plan may provide the basis for additional subcontracts to be established between the PI broker and the other involved PI service providers. Or it can be used as the basis for charges and billing to be calculated by the different parties.

Thus, the Logistics Web Layer from the perspective of the shipper is the first entry point into PI, at which the shipper agrees to ship products via the PI, as well as the basic contractual terms with the PI contact point/broker.

Information Model of the logistics web layer

The main information entities are the transport instruction, the transport contract and the transport execution plan. They all make references to the products that need to be shipped. The type quantity and other physical characteristics of these products are recorded by the inbound logistics handling system of the PI-gateway node. Of particular importance for efficient packaging of the products into P-units are the following physical characteristics:

- Net weight, in order to obey rules as to maximum weight carried by the various transport equipment (containers, trucks, etc.).
- Net volume, as to calculate the optimum number and types of P-units to be used.
- Loading weight.
- The Product category, to identify for example perishable or dangerous cargo.

Services required by the logistics web layer

The Layer will require information such as quotations from the ERP, transport management/execution and similar enterprise systems, used by the PI logistics service providers. Standards such as GS1 EANCOM, GS1 XML and GS1 UN/CEFACT XML [15] could be of relevance in this context.

Services provided by the logistics web layer

Once the shipper agrees on the terms and conditions of shipping with the FF or another PI broker, this layer exports the shipping instruction to the encapsulation layer in order for the bundling of the shipping items to π-units to be planned and then for the loading of π-units to π-containers in the nearest participating π-hub.

Relevant Standards and Technologies

GS1 and UBL 2.0 standards [15] can be used to model the key information entities described above.

Encapsulation layer

Overview

OLI defines this layer as providing the means for efficiently encapsulates products of a user in uniquely identified π-containers before accessing the π-networks. It allows linking product supply, realization, distribution and mobility taken at the upper Logistics Web level with their π - container deployment implications. It transposes decisions about moving and storing products into decisions about moving and storing π -containers. It proceeds first to encapsulation assignments of products within specific π -containers.

We interpret the above definition of the Encapsulation Layer as follows. This layer helps to maintain traceability between the original consignment/shipment and its PI specific encapsulation. As the original shipment items will now be split over possibly several π-logistic units of possibly different types and bundled as part of one or more π-containers, traceability at the original level of packaging e.g. box, carton and other retail packaging unit types. The encapsulation layer must provide information as to how the original shipment was transformed into an equivalent π-type shipment. This is important information for the following parties:

The shipper and/or the shipper's agents need to be able to trace how the original shipment flows through the π-network. In fact, this is information managed by the Shipping Layer of OLI as per Figure below.

The final recipient of the shipment (while still in the π-network needs to know the physical format of the shipment so that it can unbundle the contents from the π-units and (possibly) re-bundle them for the last leg shipment to the final destination (outside the π-network).

Information Model of the encapsulation layer

The information schema of the encapsulation layer resembles that of a packing list. It shows how the shipped items are packed inside π-units. There can be multiple levels of packaging as products can be stored for example inside their original packaging in the π-unit.

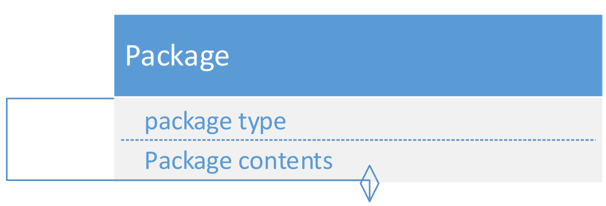

Figure 0-6 UML model of a package.

Services required by the encapsulation layer

The packing layer requires the knowledge of the original packing list of the shipment as it is generated by the shipper e.g. the factory. These services can for example be provided by the ERP or other enterprise system that is used by the shipper.

Services provided by the encapsulation layer

The encapsulation layer exposes information about the packaging of the shipped items into π-units, this is hierarchically nested information that contains at each level:

The numbers and types of π-units used (outermost layer).

For each π-unit the number, package types and content type of each package (outermost-1 layer)

…etc., until the minimal shipping unit is reached- i.e. the smallest quantity that the manufacturer ships which can be a single item, or a packing unit.

Relevant Standards and Technologies

Standards and technologies used for product labelling and unique identification for example GS1 standards [15] such as GTIN (Global Trade Item Number) can be utilised in this layer.

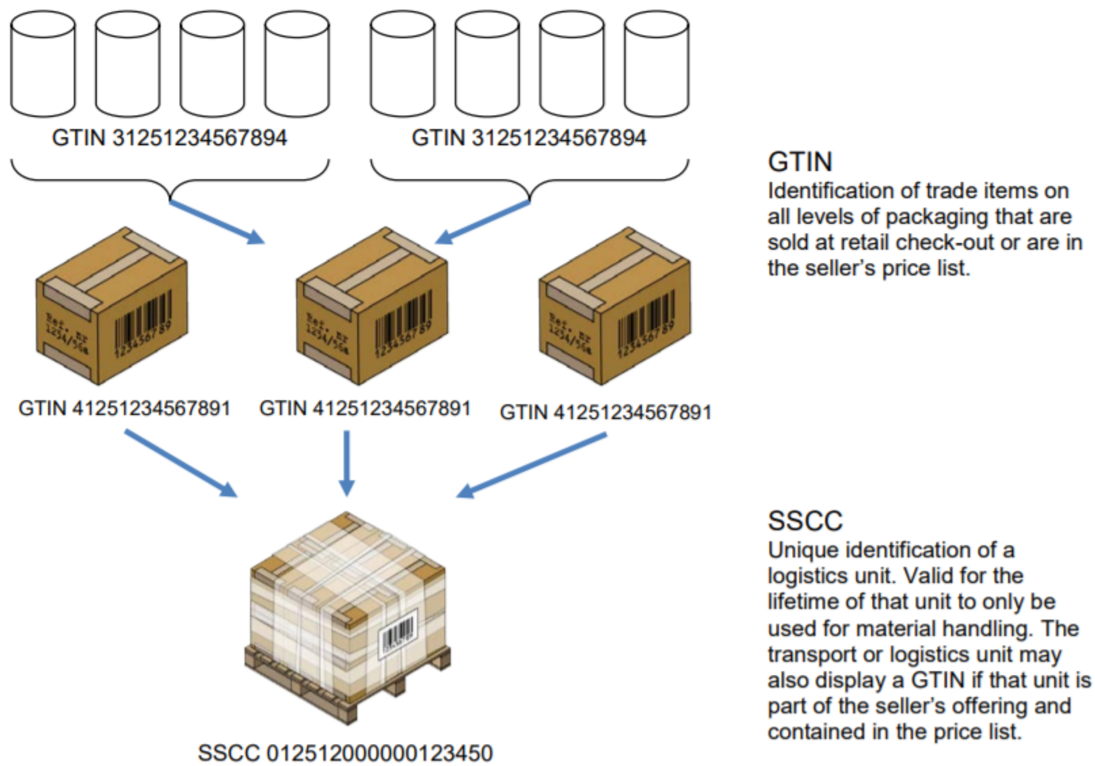

Figure 0-7 GS1 Identification standards for packages and products

As shown in Figure 0-7, multiple identification standards can be applied to describe the types of packages and their contents. For example, GS1's Serial Shipping Container Code [15] can be used by companies to identify a logistic unit, which can be any combination of trade items packaged together for storage and/ or transport purposes; for example, a case, pallet or parcel. Bar codes, Universal Product Codes, and RFID labels are also relevant standards.

Physical Layer

Overview

According to OLI, this layer is concerned with the physical objects of PI involved in handling and transporting cargo. These include π-containers as well as a variety of Physical Internet means such as vehicles, carriers, conveyors, stores and sorters. This layer validates that the physical elements are operating according to specifications, that for example a π - conveyor indeed allows moving π -containers between its entry and exit points.

We interpret the above definition as follows: The Physical Layer is the digital representation ('digital twin') of the physical entities comprising the Physical Internet. As such it is perhaps the most diverse and extensive of all OLI layers as it has to represent the vast variety of physical devices used in logistics. Moreover, many of these devices are not yet totally or at all digitised, requiring an intermediate layer of technologies such as Internet of Things technologies in order to be modelled and monitored by the Physical Layer.

Information Model of the Physical Layer

For each physical item utilised in the performance of PI functions, its digital counterpart ('twin') is maintained by this layer. This means that the layer models and executes 'active' digital 'objects' of physical resources such as:

- Transport Means such as trucks, ships, airplanes.
- Transport Equipment such as trailers, intermodal containers, wagons.
- Returnable Transport Items such as pallets, roll-containers, crates.

Services required by the Physical Layer

The Physical Layer requires the services of the automation infrastructure in the logistics equipment used for transporting handling storing and otherwise manipulating the logistics units. Data for these services are provided by automation systems for logistics operations in warehousing, logistics yards and so on.

Services provided by the Physical Layer

The Physical Layer reports the status of the physical resources to the Link Layer. This includes for example the location (both absolute and relative locations against set landmarks such as routes, π-means and π-hubs). Other types of information report for example the temperature of a shipment unit, its speed, vertical acceleration (level of g-shocks received) etc. This information is interpreted by the Event Engine of the Link Layer and might result in notifications sent (to other Layers) or actions taken by the decision-making modules operating at the Physical or other Layers.

Relevant Standards and Technologies

The Physical Layer need to be able to unambiguously identify the object/entity reported about. It also needs to unambiguously decode any contextual information i.e. what is the type of this information, the unit of measure (quantitative or quantitative) used and the actual value reported. It means that the information needs to be encoded in a structured and self-describing manner. Various standards are potentially useful at this level include:

- The use of Locodes for the unique identification of locations such as π -hubs.
- All shipped items for example start their lifecycle as a trade item (merchandise) identified by a Global Trade Item Number (GTIN).
- Internet of Things standards such as EPC/RFID.
- Internet of Things technologies such as sensors and accompanying infrastructure (routers, gateways, databases) that allow information about the physical objects to be captured, digitised and processed.

Link Layer

Overview

The link layer focuses on the detection and possible corrections of unexpected events form the operations at the Physical layer by checking consistency between physical operations and its digital mirror. It notably allows to detect and to engage protection against, or correction of dysfunctions such as a road segment or a conveyor being blocked, a π -container lost while being sorted, breakdown of security tracking of π-container moving along the π -link, or yet the appearance of an unknown security-threatening π -container. This layer is especially essential to ensure hand-over of a π-container from an operator to another and to avoid error propagation through the physical network.

We interpret the above definition of the Link Layer as follows: The Link Layer stores the digital trajectories of the PI logistics units and it compares information it receives from sensors and from logistics information systems (via the Physical Layer of OLI) with its own data in order to detect any discrepancies.

The layer implements functionalities of an event processing engine where the detection of abnormal events triggers rules that activate decision making systems at this layer or send notifications to such systems in other layers.

A state transition machine relates to the composition/decomposition of the container when π-units get bundled/unbundled from the container. This occurs at designated π-hubs. The event of changing the composition of the container needs to get communicated to other OLI layers.

Information Model of the Link layer

Like the Physical Layer, the Link Layer maintains digital twins of the physical objects of PI. These have to be actually identical to or mapped to the digital twin models of the Physical Layer. The Link Layer however implements additional digital models of the *context* in which the physical entities exist, for example models of the locations of the digital objects and of the actors that handle the physical PI entities at various time points.

This allows the Link Layer to reason on the state of the physical objects and their context and to implement action rules when certain conditions occur.

Services required by the Link layer

This Layer acts as a notification/alerting service to other OLI layers, most notably to the Encapsulation and the Logistics Web Layers. It possibly needs to service the Logistics Web layer via the Encapsulation Layer only, as conditions that occur on PI entities need to be traced back to business logistics data in order to determine which business entities (consignments/shipments) and business actors (shippers/consigners, consignees) they involve.

Services provided by the Link layer

This layer requires the services of the Physical Layer to obtain the status of relevant PI physical entities.

Relevant Standards and Technologies

The Link Layer can utilise the same standards as the Physical Layer for the digital representation of PI entities. In addition, standards for representing context (possibly ontologies and other formal models of location, time, action, state) could be useful for this Layer.

Additionally, EPCIS, a GS1 standard [15] that enables trading partners to share information about the physical movement and status of products as they travel throughout the supply chain can be considered.

Network Layer

Overview

According to OLI, the network layer focuses on the interconnectivity, integrity and interoperability of networks within the Physical Internet. It provides the functional and procedural means for insuring that π-containers can be routed within a π-network and across π-networks while maintaining the quality of service requested by the routing layer. It provides the protocols for π -containers assignment to means (handlers, vehicles, etc.) across the networks of the Physical Internet, similarly as TCP in the Digital Internet. It engages the triple-level assignments of π -containers to π -means on π -links according to the route provided by the routing layer. It monitors the π -containers as they flow across the Physical Internet, identifies routing errors and engaging in minimizing their impact, and complementarily identifies punctual routing opportunities and reacts so as to take advantage of them. This layer also defines the composition and decomposition of π containers, the assignment and control of flows of π containers across π -networks.

Each segment of this route must also be feasible and where possible, efficient. Feasible means that there exists a feasible navigable link (a road, a rail track), connecting two neighboring π-hubs, and at least one transport service capable of carrying the shipment between the two π-hubs. Moreover, the start and end π-hubs must be able to handle the shipment, i.e. constraints imposed by the shipment must not make it infeasible to handle the shipment.

Efficiency means that if there are alternative paths between the origin and destination in the shipment, the path that maximises (or minimises) some variables must be selected if possible. The variables to be optimised will be typically relate to the Quality of Service level agreed with the shipper.

Additionally, the Network Layer receives notification events from the Link Layer about the condition of π-means (the status of logistic services) or π-links. For example, disruption in a service scheduled to be used in the routing plan (i.e. delays in a ship arrival/departure) or on a π-link (delays on a road segment due to accidents or unscheduled maintenance work) will trigger conditions of rules in the Network Layer that require rerouting a shipment. Such conditions need to be transmitted to the Shipping Layer and from there to the Logistics Web Layer (through the Encapsulation Layer) in order for the final business decisions to be made.

Information Model of the Network Layer

The information model of the Network Layer can be seen as a network of state machines and their transitions, where a state consists of a π-means (e.g. a truck/trailer or a train/wagon) and a π-route and a transition consist of a change of π-means and/or π-route. This occurs for example when the container is trans-loaded to another π-means at a π-hub. There is one state machine per container.

Services required by the Network Layer

This Layer requires the services of the Physical Internet Entities that help it to establish a network model(s) and to make routing decisions. This can be static information for example the service that identifies the presence of a π-hub within a geographic region and the existence of a transport service linking two π-hubs.

In addition, this layer requires dynamic services, e.g. status updates about the PI entities. These can be obtained via notifications/alerts received by the Link Layer.

Services provided by the Network Layer

The Network Layer needs to update upper OLI layers with its routing decisions, including any re-routing decisions. As with the case of other OLI layers, some of the Network Layer's decisions need to be translated based on information available on other layers (such as the Encapsulation Layer) in order to make sense to the Shipping Layer. This for example involves any rerouting decisions and the impact they might have on shipper/customer related QoS variables such as Estimated Time of Arrival (ETA) or costs.

Relevant Standards and Technologies

This Layer can utilise technologies and standards for modelling logistics and transportation systems for example network models, representations of service timetables, as well as routing algorithms. Tracking and tracing technologies and standards are also of use, for example for tracking the location of π-containers along the π-networks.

Routing Layer

Overview

The routing layer provides the functional and procedural means for getting a set of π-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 the upper layers according to their environmental, economic, and service priority specifications. Stated otherwise, it defines for a π -container its best path according to networks status. It is at this layer that π -routing protocols are defined, put into action and controlled. It monitors the status and service capability, capacity and performance of all π -means within each π -network. It does the same at an aggregate network level. For example, it monitors the current accessibility of a given π-network.

We interpret the above functionalities of the Routing Layer as follows: The Network Layer selects the feasible/optimal routes (out of those identified by the Networking Layer) through the PI that connect the origin of the shipment (i.e. the initial π-hub handling the π-units comprising the shipment to the final destination/ π-hub that will handle the shipment. So, while the Networking Layer defines all possible routes between origin and destination of the shipment, it is the Routing Layer that at transport execution time selects the feasible/optimal ones.

Information Model of the Routing Layer

For the purpose of routing, the Routing Layer utilises a model of the π -network(s) provided by the Network Layer. This can be centralised or distributed, i.e. a single model of the whole network is held centrally, or each PI node such as a π -hub maintains a model of its own local network. These local models can of course be synchronised and propagated across the whole π –network where ultimately each node maintains a copy of the whole π -network. In addition, each node (e.g. a π-hub) maintains a routing table describing all logistics services available to reach the π hubs it is connected to. Additionally, each π-hub can maintain a link-cost table for each of its neighboring π-hubs.

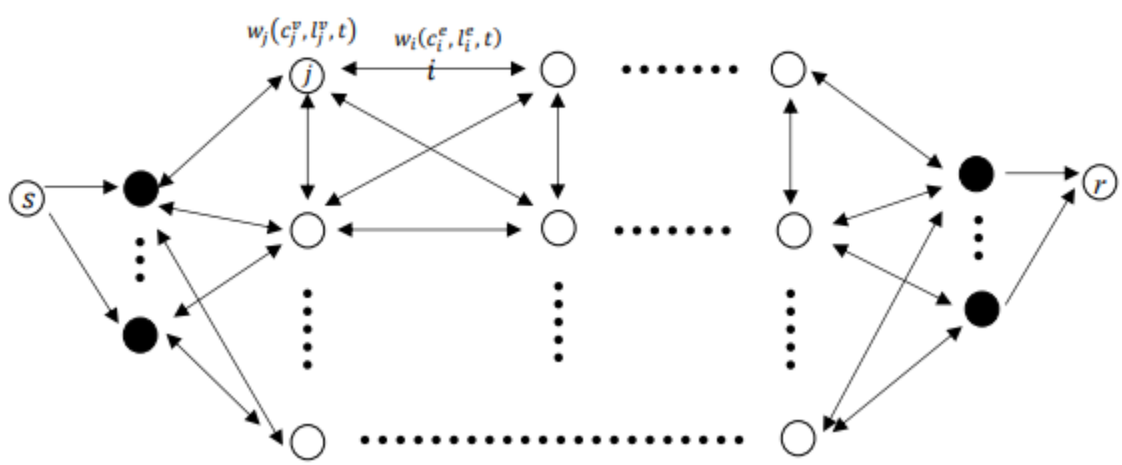

Figure 0-8 A simple network model for the Physical Internet [11]

A model that could potential apply to the Routing Layer for optimization as well as to the Networking Later to address the reachability problem is the Simple network model which covers the reachability and optimality problems of a network [11]. Heuristic algorithms [8] could also improve the relocation rate in container yards, the determination of the storage locations for relocated containers. Moreover, the routing layer could benefit in optimising routing by utilising Algorithmic Game Theory which “typically models applications via concrete optimization problems and seeks optimal solutions, impossibility results, upper and lower bounds on feasible approximation guarantees” [9, p.1].

Services required by the Routing Layer

The Routing Layer requires models of the π-network in order to calculate suitable routes through it. These are provided by the Network Layer who must be ensure that an accurate model of the underlying physical network is maintained.

Services provided by the Routing Layer

The Routing layer informs the Shipping Layer about the route that a particular shipment (in terms of the π-units containing the shipment items) will follow. Information about the route or segment such as the transit time, means of transport, ETD and ETA will need to be delivered by the Routing Layer in order for the Shipping Layer to appraise the proposed route in terms of the original transport contract and its service level agreements.

Relevant Standards and Technologies

The Routing Layer information sets need to be compatible with those of the layers immediately above and below it. Thus, as per the previous layers, GS1, UBL [15] and other related standards such as UN/LOCODE can be utilised.

Shipping Layer

Overview

According to OLI, the shipping layer provides the functional and procedural means for enabling the efficient and reliable shipping of sets (corresponding to orders for instance) of π -containers from shippers to final recipients. 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.

It gets shipping requests from the deployment layer and it requires transport services for its shipments from the transport layer.

We interpret the above definition of the OLI Shipping Layer as follows. The Shipping Layer represents the interface between the business side of the Pi contract established by the shipper and the PI broker, and the operational side (the π-network). The terms and conditions of the contract must be fulfilled by the π-operators. All service quality agreements for example regarding shipment status notifications need to be met by the π operations. All such notifications must be pushed through the π-network to the stakeholders i.e. the shipper, FF/broker and customer. So, procedures must be established and remain operational throughout the execution of the transport in order to monitor its progress and to help enforce/maintain the agreed service level standards.

Information Model of the Shipping Layer

The Shipping Layer utilises an entity centric view of the shipment which can be implemented in a state transition diagram with events triggering state changes. Events originate in the π-network and through layers such as the Link Layer are propagated to the Shipping Layer that is responsible for making some decisions for example regarding the termination or (approval of) diversion of a shipment.

Services required by the Shipping Layer

The shipping layer receives updates of the status of the shipments through the routing layer. This includes information about routes to be followed, ETAs (total and for each route) and any deviations from the original transport plan in terms of time, costs, etc.

Services provided by the Shipping Layer

The Shipping Layer needs to receive services from the Logistics Web Layer (via the Encapsulation Layer), regarding instructions on how to proceed with a shipment. Correspondingly, it needs to instruct the Routing Layer about such decisions, as the Routing Layer operationalises the transportation plan and needs to instruct appropriately the lower operational layers to for example reroute, adjourn or cancel a shipment.

Relevant Standards and Technologies

The Shipping Layer can utilise the same logistics information standards as the layers above and below it in order to ensure interoperability.

The following segment of UBL 2.0 XML document shows a sample shipment. Information about this shipment will be updated by the Routing Layer when the route (or a segment of a route) are determined by the Routing Layer.

```
<cac:ShipmentStage>
  <cbc:ID>normalizedString</cbc:ID>
  <cbc:TransportModeCode>normalizedString</cbc:TransportModeCode>
  <cbc:TransportMeansTypeCode>normalizedString</cbc:TransportMeansTypeCode>
  <cbc:TransitDirectionCode>normalizedString</cbc:TransitDirectionCode>
  <cbc:PreCarriageIndicator>true</cbc:PreCarriageIndicator>
  <cbc:OnCarriageIndicator>true</cbc:OnCarriageIndicator>
  <cac:TransitPeriod>...
  </cac:TransitPeriod>
  <cac:CarrierParty>...
  </cac:CarrierParty>
  <cac:TransportMeans>...
  </cac:TransportMeans>
  <cac:LoadingPortLocation>...
  </cac:LoadingPortLocation>
  <cac:UnloadingPortLocation>...
  </cac:UnloadingPortLocation>
  <cac:TransshipPortLocation>...
  </cac:TransshipPortLocation>
  </cac:ShipmentStage>
```

Annex II: The OLI Protocol Stack and the Living Labs

Introduction

The LLs provide the testing ground of the innovations and capabilities to be offered by the PI concept while at the same time identifying areas needing further study and verification.

With the Use Cases and set up activities completed for the four LLs, deliverable D3.1 has laid out the specific processes and elements required for each LL to compliment the architectural layers and logistics services to ensure interoperability and data flow. The report specifically addressed the protocol stack and layered architecture and its relevance to the deployment of the LLs. More specifically a closer look on the services required by each layer but also the services provided by each layer to layers above or below in the architectural stack.

One of the core objectives of the LLs has been identified as the need for continuous communication and feedback to/ from WP1 and WP2 to further enhance and improve the design work related to PI framework and platform/ reference architecture.

LLs and the OLI model

It is important in this second version of the report, aiming to satisfy requirements laid out by DoA through task T1.6, to have indeed a closer look on the proposed-to-inspire OLI model, the architectural layered stack and its effect on the LLs framework. The LLs set up and requirements will guide further the architectural considerations on an on-going basis.

Therefore, based on work in D3.1, a quick reminder of the LLs use-cases and how the OLI model affected the design of each LL, is provided below:

LL1 PI Hub-centric corridor

<u>Use Case</u>

The LL1 will implement and validate PI concepts in the complex transport landscape of the area of Antwerp, composed of three port mega-hubs (Antwerp, Gent and Zeebrugge), each of which (due to its size) can be considered as a PI Hub-centric network. The maritime and continental hubs and terminals of these ports will be considered as the primary PI Nodes, whereas trains, trucks and barges will be the PI Means, and the respective train, road and barge lines/services will be the PI Links.

The goal of the PI-centric approach in this LL is to streamline the mega-hubs’ operations, reducing congestion and bottlenecks in the flow of goods, especially in left/right bank trips. The LL provides the opportunity to simulate and study PI concepts and network operations at two different scales: intra-facility inter-center network and intra-country inter-state network.

<u>LL1 and the OLI Layers</u>

The specific use cases will inform the various layers as table below shows:

Table 0-1 LL1 and the OLI layers reference

	Layer	Description	LL's reference

1	The Physical Layer	Operations related to the Physical Internet	This Layer provides all the different physical means involved in handling and transporting cargo i.e. trains, wagons, containers, vessels, cranes, trucks and various operating units (e.g. loading units) or even Logistical systems, Blockchain solutions in the Port. These are not all expected to be fully digitized at this stage so monitoring by the Physical Layer and IoT devices will play an important part.
2	The Link Layer	Node to node transfer	The maritime and continental hubs and terminals of the ports will be considered as the primary PI Nodes. Intercommunication and cross referencing of data will ensure the consistency between the physical entities and their digital specifications. Possible dysfunction on the transfer status of PI Units within the PI Network will have to notify other OLI layers but most notably the Logistics Web Layer and the Encapsulation Layer.
3	The Network Layer	Interconnectivity , integrity and interoperability of networks	Based on the destination and delivery date of the PI-Containers the Networking Layer provides the available links for (re)-routing containers within the port. Furthermore, the Network Layer provides all the available transport means and routes (road, rail etc.) to the Routing Layer.
4	The Routing Layer	Routing of the PI containers from starting point to their destination	Based on the available PI-Means and routes that are provided by the Networking Layer, the Routing Layer optimises road and rail journeys. This is achieved by bundling of wagons for the same hub on the same train while taking into consideration pre-defined parameters and constraints (e.g. initial location, destination, delivery date etc.). Moreover PI-Containers are bundled into wagons on similar principles as the aforementioned.
5	The Shipping Layer	Reliable shipping of PI containers	Through the collection of information (e.g. status of shipment, ETA, costs) from other OLI layers, this layer establishes the efficient shipment of orders and as per the contract, service agreements and standards already in place between clients and agents/ brokers/FFs or generally PI services providers. Bundling of Wagons /Containers, re-routing of consignments, modal shift to rail mode are all examples of other OLI layers' services to shipping layer to ensure accurate shipping notifications and acknowledgements.
6	The Encapsulation Layer	Stuffing /Unstuffing products to PI containers	This layer will provide this LL with all the necessary information on the bundling of wagons and containers in an effort to optimize the provided services and flow of goods at the port amongst the different hubs/ PI nodes. The information flows relate to the products themselves, the PI means and PI networks and retain the path followed of a certain product unit from client/ shipper end to destination end.

7	The Logistics Web Layer	Interface between the Physical Internet and the users of the logistics services	The main information will relate to the transport instructions and execution plan (different modes and service levels) as well as information of the actual products to be shipped (weight, volume etc.). The optimization of the Port's infrastructure with emphasis on railway services while enhancing communication and efficiency will provide the required services under this layer.

Living Lab 2: Corridor-centric PI Network

Use Case

Focusing on the North Sea – Mediterranean Corridor, smart-sensors will be engaged on the existing transport infrastructure. LL2 will examine the applicability of IoT through progressively transforming typical transport corridors into PI corridors, with the emphasis to enhancing the reliability of intermodal connections, paving the way to implement synchromodality at an operational level, and ultimately understanding decision making characteristics with regards to delaying or pulling forward loads or modal shift.

LL2 and the OLI Layers

The specific use cases will inform the various layers as table below shows:

Table 0-2 LL2 and the OLI layers reference

	Layer	Description	LL's reference
1	The Physical Layer	Operations related to the Physical Internet	This Layer incorporates the physical entities and in this LL these include trucks, rail, wagons, containers and other supply chain operating units (e.g. cranes, conveyor belts) used to transport for example a PI load unit from entry point to exit.
2	The Link Layer	Node to node transfer	In this LL the Layer provides services through the use of smart IoT devices that are installed on PI-Containers. This layer will detect discrepancies through a number of Trackers and Sensors which all record and transmit data, referencing, amongst others, the position, the condition and other important parameters of the PI containers and goods.
3	The Network Layer	Interconnectivity, integrity and interoperability of networks	The focus of this LL lays on the services of the Networking and Routing Layer. The Networking Layer provides the interconnection and interoperability of Road and Rail networks. In this case the focus will be in two major Corridors: (1) Mechellen to West Thurrock and (2) Mechellen to Agnadello. In addition, it provides the functional and procedural means (Trucks and Trains) for insuring that π-containers can be routed within a π-network and across π-networks while maintaining the quality of service requested by the routing layer. This layer is utilised by the Routing Layer to select the optimal transportation mode for product transportation between its origin and

			destination and (re)routing and (re)prioritizing containers across based on requested delivery date.
4	The Routing Layer	Routing of the PI containers from starting point to their destination	In conjunction with the Network Layer, this layer (re)routes and (re)prioritises containers across from its source to its destination based on requested delivery date. Several factors are taken into consideration within the scope of routing such as Total Transit Time, Waiting Time, Delivery Date, Average Speed, Cost, etc. Algorithms and other models will be used to optimise alternative routes on the North Sea (Mediterranean Corridor) based on P & G's business criteria. Possible re-routing will have to notify other OLI layers for further adjustments.
5	The Shipping Layer	Reliable shipping of PI containers	This layer will ensure the successful operational activities within the shipment of containers from supplier to customer via the use of services from other layers. The enhanced, by the use of IoT devices, Supply Chain visibility throughout the chosen PI corridors will enable the monitoring and verification of the shipping service to the client.
6	The Encapsulation Layer	Stuffing /Unstuffing products to PI containers	This layer does not provide any services as it applies to stuffing and unstuffing of containers.
7	The Logistics Web Layer	Interface between the Physical Internet and the users of the logistics services	The synchromodality through the internet corridor services, offering different capabilities through the PI nodes, PI means and PI Links is a major ingredient of this Layer in this LL, providing the shipper/ agent with possible PI solutions and benefits.

Living Lab 3: e-Commerce centric PI Network

<u>Use Case</u>

LL3 will demonstrate the application of PI principles in optimizing the Fulfilment of e-Commerce Purchase Orders, utilizing local stores as PI Nodes, by reducing lead time, travelling/fulfilment time and stock-outs, in SONAE's logistics network. A consumer driven approach will be adopted to increase the use of environmentally friendly Service Points optimized

<u>LL3 and the OLI Layers</u>

The specific use cases will inform the various layers as table below shows:

Table 0-3 LL3 and the OLI layers reference

	Layer	Description	LL's reference
1	The Physical Layer	Operations related to the Physical Internet	This Layer will provide the physical means like delivery vehicles, unloading forklifts, conveyor belts, pallets, picking vehicles (e.g. reach trucks), unloading equipment all operating in the PI nodes and adding value to the supply chain. IoT technologies like sensors will feed this layer with readable information regarding for example the position and status of the goods in the PI network
2	The Link Layer	Node to node transfer	This Layer will provide services from IoT devices on trucks, vans and stores from different PI means and PI nodes to track parcels and goods delivered to clients and therefore ensuring the integrity of the flow in the PI network of urban distribution. Stock out warnings and traffic information can detect disruptions of the routing models to be developed and again inform other OLI layers of such disruptions.
3	The Network Layer	Interconnectivity , integrity and interoperability of networks	This Layer will allow the interconnection with distribution networks consisting of company owned and open distribution centers/fulfilment hubs operated by third party fulfilment service providers (FSPs). Interconnecting with other networks to utilise on demand dynamic/mobile facilities, renting/leasing and sharing is crucial.
4	The Routing Layer	Routing of the PI containers from starting point to their destination	Within SONAE's network the Routing Layer measures the cost of order preparation that consists of different routing related variables such as delivery cost. This layer should optimise the best time windows to offer in the different delivery regions in order to facilitate more effective and efficient delivery operations, while making delivery more sustainable. The Routing Layer is used in assessing the lead time of the fulfilment order in order to be within the objective time frame. Furthermore, "Picking efficiency" can also depend on the items/routes.
5	The Shipping Layer	Reliable shipping of PI containers	This Layer will provide all operations, technologies and procedures to ensure the reliable shipping of parcels and e-orders to clients. The PI nodes network of SONAE including local stores and regional warehouses together with optimum distribution routing will result in cost and time savings guaranteeing agreed client service levels. The Layer will also decide on needed operations based on information received from Logistics Web layer on how to proceed with a shipment.

6	The Encapsulation Layer	Stuffing /Unstuffing products to PI containers	This layer will provide information on pallets stored in Warehouses which are transformed into PI-Units when the SKUs are stored/unpacked on the shelves of each type of store which serves as a PI-Node. Moreover, fragmentation and de-fragmentation of orders to enable cost efficient delivery of products through geographically dispersed nodes, must preserve product traceability to enable keeping track of the status of the products.
7	The Logistics Web Layer	Interface between the Physical Internet and the users of the logistics services	In this LL the eCommerce channel orders fulfilment through SONAE's PI nodes and network of stores will provide the client with a range of solutions as to which service best serves their interest while at the same time avoid stockouts at stores and achieve desired efficiency and optimum client service.

Living Lab 4: Warehousing as a Service

Use Case

Stockbooking (SB) will identify available spaces into the warehousing facilities and will combine the latest with client's requests for logistics services on-demand and dynamic basis. To test the PI solution, SB will study different scenarios (business cases) with various specificities and define whether such user cases fit the needs of PI distribution and could gain savings and optimization or not. On top of that, more complex scenarios will be formulated like the capacity of the stores to provide a more relevant PI business function

This LL aims to investigate the potential of e-Warehousing as a key enabler for the PI concept. Hence, this LL will serve as the testbed for testing and improving warehousing services structured under the PI concept. The LL provides the opportunity to simulate and study PI concepts and network operations at the scale of an intra-center inter-processor network. The location of warehouses can strategically benefit the smooth flow of containers along PI Corridors achieving at the same time savings and quality of services offered.

LL4 and the OLI Layers

The specific use cases will inform the various layers as table below shows:

Table 0-4 LL4 and the OLI layers reference

	Layer	Description	LL's reference
1	The Physical Layer	Operations related to the Physical Internet	The physical layer provides the physical means which generally manipulate the logistics units. In this case relate to the operation of the PI nodes i.e. the SB stores so these are items like containers, forklifts, pallets, trucks, storing systems, sorting equipment, loading/ unloading vehicles and so on. Data for this layer will originate from devices collecting information from these assets. Data will be forwarded to the Link Layer.

2	The Link Layer	Node to node transfer	This layer will ensure the uninterrupted flow of containers in the PI designed-network of stores of SB and attaining the targeted service levels of clients. This to be achieved by IoT devices in containers and stores as well as sensors indicating the store capacity (in a PI network of PI nodes) and forecasted positioning of goods to ensure maximised benefit for the customer and the hubs' own productivity
3	The Network Layer	Interconnectivity , integrity and interoperability of networks	While, currently, the Stock-Booking's network does not interconnect with other networks, the networking layer can be utilised in order to connect with external networks. Interoperability with other warehousing networking within the Physical Internet can offer utilization of "external" warehouses to increase storage availability and enhance transportation. At the same time, Stock-booking's warehousing and transportation networking can be utilised as a service by other networks for additional revenue while improving nodes' efficiency and productivity through better planning
4	The Routing Layer	Routing of the PI containers from starting point to their destination	The Routing Layer provides the essential services to LL4. Firstly, the Routing Layer is used to provide the list of suitable options for warehousing storage that will consider various variables/constraints such as type of goods, weight, value, source, destination(s), date received, delivery date, etc.). Secondly the Routing Layer will be able to suggest a new PI-Hub to improve the aforementioned characteristics based on some criteria and through monitoring the status and service capability, capacity and performance of all PI-means within the network.
5	The Shipping Layer	Reliable shipping of PI containers	The Layer will manage the shipping of the orders through the Logistics services (to be) provided by SB as far as the warehousing offerings are concerned. Services and information from other layers like Logistics Web layer become important (e.g. resources-in-logistics-means available, store capacity, stock turnover, Blockchain solutions and WMS systems) to make sure the shipping contracts' terms and conditions are met.
6	The Encapsulation Layer	Stuffing /Unstuffing products to PI containers	This layer will provide traceability of original shipments e-warehoused at possibly different locations or PI nodes or depending on products-nature even different shelf locations within the same PI node. Optimised transportation flows within a warehouse buffer and routing PI service, maintain PI unit's visibility at all times aided by the use of Technology like Blockchain solutions for trusted transfer of distributed ledgers.

7	The Logistics Web Layer	Interface between the Physical Internet and the users of the logistics services	The client (stock-booking) will be presented with information regarding the most suitable warehouse for the goods' storage. The Web Layer receives information (input) the various variables/constraints such as type of goods, weight, value, source, destination(s), date received, delivery date, etc.). The user will receive information regarding storage availability, cost (at every step of the process). The output will be a warehousing network map along with the proposed warehouse(s) for storage along with the time period. The system will be multiple options with varying costs and quality of service (e.g. delivery date, special storage conditions, etc.) as well as proposed scenarios for improved KPIs for the company.

Annex III: Digital networking technologies as PI enablers

Digital to Physical Internet

This section examines the applicability of existing Digital concepts and technologies to the Physical Internet, in order to gain further inspirations for the design of PI services.

We break down the discussion of digital networks into a number of sub-areas:

- *Network architectures*, in particular the architecture of the (digital) Internet, and how the Physical Internet architecture can map to it.
- *Networking*: What are the building blocks that when connected form a network. How the building blocks/components of the digital network and their types of connections correspond to the building blocks and connections in the Physical Internet.
- *Routing*: How information flows through the digital network? What are the rules/protocols for routing such information? How the routing concept applies to the flows of physical objects through the Physical Internet?

We discuss concepts of modern computer network architectures such as Software Defined Networking (SDN) and Network Function Virtualisation (NFV) and the potential inspiration they can provide to designing the Physical Internet. We also consider special types of networks and routing techniques such as mobile ad-hoc networks and content-based routing. These could also have potential applicability in some areas or applications of the Physical Internet. In addition, as per Figure 4.1 we discuss desirable properties of networks in general: fault tolerance, survivability and dependability, and how these properties apply to the Physical Internet.

The next Section discusses switched and packet-based networks of which (the later) the digital Internet is a member. Section 4.3 presents a view of the Internet as a network of autonomous systems. Section 4.4 presents the concept of routing and the main routing approaches and protocols used by the Internet. Section 4.5 discusses Software Defined Networking (SDN) and Network Function Virtualisation (NFV). Section 4.6 discusses network properties such as fault tolerance, robustness and survivability. In all these sections, there is also a consideration as to how the discussed principles and topics may apply to the Physical Internet.

Connection Oriented and Connectionless Networks

Packet- and Circuit-Switched Networks

The concept of circuit-switched networks is based on fixed circuits that establish a single route for data between nodes of the network that does not change for the life of the connection. Circuit-switched networks are therefore connection-oriented. IBM's Systems Network Architecture (SNA) and Asynchronous Transfer Mode (ATM) are two examples of circuit-switched networks.

All early data networks were circuit switched. However, the fact that packet-based networks permits the interconnection of far more nodes into a single network, made packet-based networks more popular than circuit-switched ones for many applications. Packet based networks also facilitate the interconnection of networks into an inter-network. This is one of the main principles of Internet.

Overall however there are benefits and drawbacks in both packet and switch-based networks. Some of these are:

- In Packet-oriented networks the destination address is encoded in the packet itself, making routing more flexible with regard to paths.
- Circuit-switched networks may require additional setup time if the circuit is not established on a permanent basis. Once the circuit is established, however, no routing information or decisions as to how the data need to move through intermediate nodes is required.

- Packet oriented networks are generally more economical than circuit-switched networks because of their ability to share traffic.

The equivalent concept in transport and logistics is that of point to point transport (I.e. the switch-based concept) versus transporting via intermediate terminals and transhipment/consolidation of shipments (with a shipment corresponding to one or more data packets). While point to point transport may be the most (time) efficient approach, the consolidation of shipments into larger units may require intermediate stops and additional operations but may be a more economic option for the shippers.

Network Layers

Networks can be considered at different levels of abstraction, or in terms of layers. In a layered network architecture, the lower layers of a network can be connectionless, but the higher layers can establish a logical connection. The Transmission Control Protocol (TCP) over the Internet Protocol (TCP/IP) is an example of such a layered network approach. The opposite is also possible, i.e. where the lower layers establish a connection and the upper layers do not. The later type of connection is of relevance to PI as for example the distribution centres within a (sub) network of PI are connected to each other in a direct manner, but to other PI nodes via π-hubs.

The Layered OSI model was briefly discussed in Section 3. In this section we focus on Layers 2 and 3 of OSI and discuss their relevance to the Physical Internet. Layer 3 works on top of Layer 2. Data bits are transferred over a variety of medium, cables, ports etc. Frames are used to define the data between two nodes on a data link, and when there's more than two nodes, higher layers of the OSI help to address route and control traffic.

Layer 2 defines the protocol to both establish and terminate a physical connection between two devices. Layer 3 works with IP addresses, while Layer 2 works with MAC addresses (unique identifiers for the network adaptor present in each device). IP addresses are therefore a layer of abstraction higher than MAC addresses. Also, unlike MAC addresses, IP addresses can be dynamic, i.e. 'leased' or assigned generally by a DHCP server.

Layer 2 networks forward all their traffic, so data transmitted by one device on L2 will be forwarded to all devices on the network. This type of broadcast is very fast, but as the network increases in size it creates congestion and leads to inefficiency over the network. In contrast, Layer 3 restricts broadcast traffic. Administrators on L3 can segment networks and restrict broadcast traffic to subnetworks, limiting the congestion of broadcast on large networks. Layer 3 networks therefore run on top of Layer 2 networks and are therefore one layer of abstraction higher than Layer 2. Layer 3 networks route using IP addresses and are therefore better for managing network traffic over multiple sites and through the internet. According to the OSI, the main difference between a Layer 2 switch and a Layer 3 switch is the routing function. A Layer 2 switch only works with MAC addresses, not with any higher layer addresses, such as an IP. A Layer 3 switch, on the other hand, can also do static routing and dynamic routing, which includes IP and virtual local area network (VLAN) communications.

In the context of the Physical Internet, it is interesting to consider the roles and correspondences of PI nodes to the Layer 2 and 3 switches and routers of the digital networks. Clearly, π-hubs act as routers as they interconnect different transport/logistics networks. These hubs therefore are packet not switch oriented, at least regarding their Physical Internet facing interfaces. They need to make routing decisions based on the address information available on the (physical) packet. At the same time, π-hubs act as switches as they have (fixed) connections to other Physical Internet nodes such as local terminals, consolidation/distribution centres, warehouses, etc. In this context, π-hubs do not need to make routing decisions as each physical packet is directed to a fixed path, i.e. similar in concept to a MAC address.

High Level Architecture of Internet

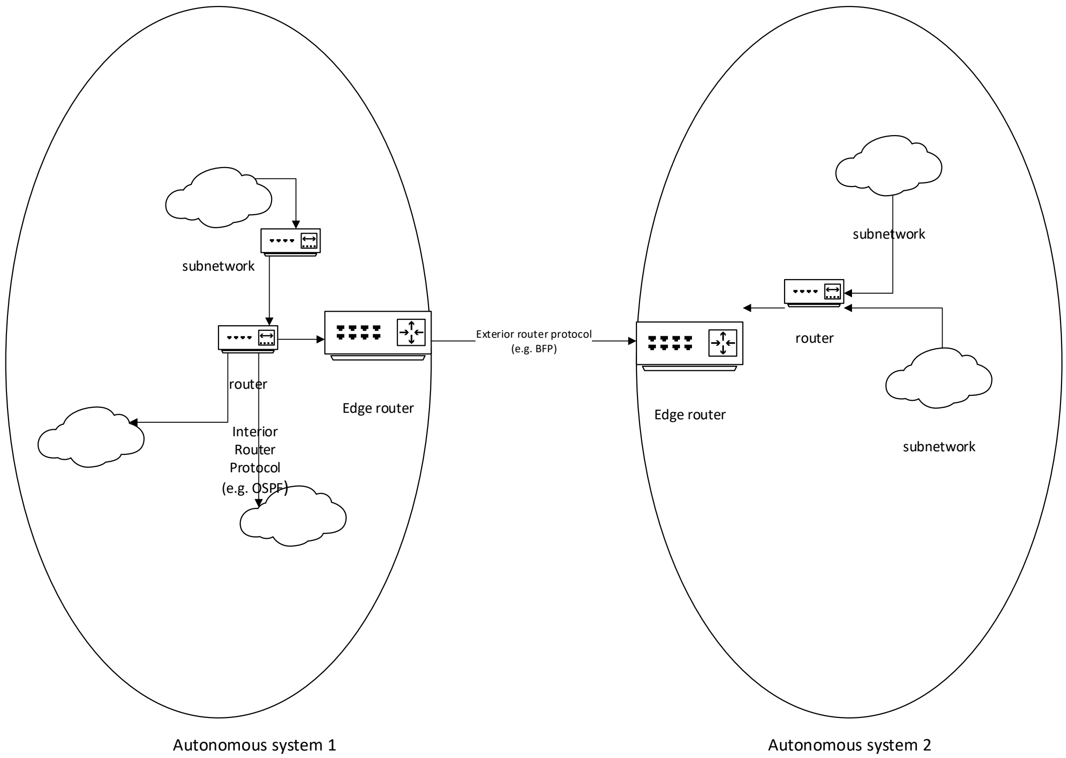

Figure 1-0-1 -Autonomous systems in the Internet Architecture

Autonomous Systems

The Internet can be viewed at a high level as a network of interconnected autonomous systems. An autonomous system (AS) is a set of routers and networks managed by a single organization. An AS consists of a group of routers exchanging information via a common routing protocol. Unless it is in a failure state, an AS is a connected graph, i.e. there is a path between any pairs of nodes.

As per Figure 1-1, edge routers (also called gateways) communicate and exchange routing information using an exterior router protocol such as BGP. BGP involves the performance of neighbor acquisition, neighbor reachability and network reachability.

Interior Gateway Protocols (IGP)

Interior Gateway Protocols (IGP) route Internet Protocol (IP) packets within a single routing network domain only. IGP protocols calculate the shortest path between the source and destination based on link bandwidth cost and then allows the network to send and receive IP packets via the shortest route. The most common interior routing protocols are discussed in the following section.

Border Gateway Protocols

RFC 1654 defines Border Gateway Protocol (BGP) as an EGP standardized path-vector routing protocol that provides scalability, flexibility, and network stability. BGP was designed primarily for IPv4 inter-organization connectivity on public networks, such as the Internet, or private dedicated networks. BGP is the only protocol for exchanging routing data between networks on the Internet. In IBGP protocols, peering relationships are created between edge routers. For edge routers to be able to establish peering, they must also run an IGP such as OSPF, RIP or ISIS (reviewed in following sections).

Relevance of AS approach to PI

Due to its size and heterogeneity (in terms of types of participants, business practices, logistics technologies, etc.) it has been proposed that the future PI will be organized as a collection of autonomous systems. An autonomous system in this context could be the transport network of a single organisation (e.g. a large shipper, logistics service provider, etc.) and as such will be administered by a single entity and have its own routing technology.

Routing

The key function of a router is to accept incoming packets and forward them appropriately (e.g. based on information contained in the packet's header). A router maintains forwarding tables, where a table shows for its destination, the identity of the next node (router). Additional information used for routing may include the source address, packet flow identifier and security level of packet. The destinations, the associated cost and the next hop to reach those destinations form the IP routing table.

Routers are responsible for discovering appropriate routes through the network. Alternatively (as we shall discuss in the section about Software Defined Networking-SDN), a network control centre may maintain a central forwarding table. As each router makes routing decisions based on knowledge of the topology and traffic conditions of the Internet, dynamic cooperation is needed amongst the routers.

Routing protocols like OSPF, calculate the shortest route to a destination through the network. The first routing protocol that was widely implemented, the Routing Information Protocol (RIP), calculated the shortest route based on hops, i.e. the number of routers that an IP packet had to traverse to reach the destination host. RIP successfully implements dynamic routing, where routing tables change if the network topology changes. However, RIP could not adapt its routing according to changing network conditions, such as changes to data transfer rates. Therefore, new dynamic routing protocol that could calculate the fastest route to a destination were required. OSPF is one of such protocols. It was developed so that the shortest path through a network was calculated based on the cost of the route, taking into account bandwidth, delay and load. Therefore, OSPF calculates the cost of each route on the basis of configurable link-cost parameters. OSPF was quickly adopted because it became known for reliably calculating routes through large and complex local area networks.

The above routing protocols are discussed in more detail below.

Routing Information Protocol (RIP)

Routing Information Protocol is a distance vector protocol that uses hop count as its primary metric. The term 'distance vector' refers to the fact that the protocol utilises vectors (arrays) of distances to other nodes in the network. RIP defines how routers should share information when moving traffic among an interconnected group of local area networks (LANs). RIP was defined in RFC 1058 in 1988.

RIP is a dynamic routing protocol that uses a distance vector algorithm to decide which path to put a packet on to get to its destination. The protocol only allows only 15 hops in a path- If a packet can't reach a destination in 15 hops, the destination is considered unreachable.

Each RIP router maintains a routing table, which is a list of all the destinations the router knows how to reach. Each router broadcasts its entire routing table to its closest neighbors every 30 seconds. In this context, neighbors are the other routers to which a router is connected directly on the same network segments this router is on. The neighbors, in turn, pass the information on to their nearest neighbors, and so on, until all RIP hosts within the network have the same knowledge of routing paths. This shared knowledge is known as convergence.

If a router receives an update on a route, and the new path is shorter, it will update its table entry with the length and next-hop address of the shorter path. If the new path is longer, it will wait through a "hold-down" period to see if later updates reflect the higher value as well. It will only update the table entry if the new, longer path has been determined to be stable.

If a router crashes or a network connection is severed, the network discovers this because that router stops sending updates to its neighbors, or stops sending and receiving updates along the severed connection. If a given route in the routing table isn't updated across six successive update cycles (that is, for 180 seconds), a RIP router will drop that route and let the rest of the network know about the problem through its own periodic updates. The Rapid Spanning Tree Protocol (RSTP) standards provides significantly faster spanning tree convergence after a topology change, introducing new convergence behaviors and bridge port roles to do this. RSTP was designed to be backwards-compatible with standard STP.

In enterprise networking, Open Shortest Path First (OSPF) routing has largely replaced RIP as the most widely used Internet Gateway Protocol (IGP), due to RIP’s inability to scale to very large and complex networks.

Open Shortest Path First (OSPF)

OSPF is an Intranet protocol that is, it is used within an AS (Autonomous System), i.e. an IGP type protocol as explained above. An OSPF network can be divided into sub-domains called areas. An area is a logical collection of OSPF networks, routers, and links that have the same area identification. A router within an area must maintain a topological database for the area to which it belongs. The router does not have detailed information about network topology outside of its area, which thereby reduces the size of its routing table.

Areas limit the scope of route information distribution. An area border router (ABR) is a kind of router that is located near the border between one or more Open Shortest Path First (OSPF) areas. ABR routers are used to establish a connection between backbone networks and the OSPF areas. An ABR stores and maintains separate routing information or routing tables regarding the backbone and the topologies of the area to which it is connected. The main function of ABR therefore is to summarize sub networks found throughout the OSPF system.

OSPF is a link-state protocol, where a link is an interface on the router. The state of the link is a description of that interface and of its relationship to its neighboring routers. A description of the interface would include, for example, the IP address of the interface, the mask, the type of network it is connected to, the routers connected to that network and so on. The collection of all these link-states forms a link-state database. As a link state routing protocol, OSPF maintains link state databases, which are network topology maps, on every router on which it is implemented. The state of a given route in the network is the cost, and OSPF algorithm allows every router to calculate the cost of the routes to any given reachable destination. Typically, the link cost of a path connected to a router is determined by the bit rate of the interface. A router interface with OSPF will then advertise its link cost to neighboring routers through multicast, known as the hello procedure. All routers with OSPF implementation send periodically hello packets, and thus changes in the cost of their links become known to neighboring routers. The information about the cost of a link, i.e. the speed of a point to point connection between two routers, is then cascaded through the network, using the process of synchronisation, in which OSPF routers advertise the information they receive from one neighboring router to all other neighboring routers. Based on this synchronised information, all routers with OSPF implementation continuously update their link state databases with information about the network topology and adjust their routing tables.

Other Routing protocols

Content Centric Networking

Content-centric networking is based on an addressing scheme wherein the send and receive communication primitives identify content rather than network locations. This addressing scheme is motivated by social, application-level considerations, as much as by technical, network-level considerations. At a high-level, communication can be more effective if information consumers can simply specify what content they intend to receive as opposed to from where that content might be retrieved. Content-centric networking proposes an addressing scheme that identifies content as opposed to location, to allow the network to operate more efficiently by duplicating and caching content around the network, as it is the delivery of content that matters, not where that content resides. Content-centric networking is therefore an approach to the problem of content distribution, especially for cases where users request named content.

Mobile Ad Hoc Networks (MANET)

Mobile ad-hoc networks (MANETs) have proposed forwarding and routing mechanisms for dynamic networks in which the connectivity among members is continually changing. These techniques apply also to more general delay-tolerant networking and disruption-tolerant networking in which stable end-to-end paths may never exist. Techniques that support routing in mobile ad-hoc networks include communicating as far as possible but reverting to store-and-forward when necessary, and mobile nodes carrying information, called store-and-haul, store-carry-forward, or ferrying.

The third contributor to new routing protocols is energy-constrained networks, exemplified by wireless sensor networks (WSNs), in which nodes with drained batteries can no longer contribute to network connectivity. In WSN, the routing protocols are responsible for maintaining the routes in the network and to ensure reliable multi-hop communication. Node deployment affects the performance of routing protocol. If the sensor deployment is deterministic the data is routed through pre-determined paths. In self organizing deployments, the sensor nodes are scattered randomly creating an infrastructure in ad-hoc manner, therefore, it is likely that a route will consist of multiple wireless hops. Routing messages from or to moving nodes is even more challenging as the routing stability is an important issue.

Variants, or specialised areas of the Physical Internet where some of the nodes are mobile can possibly utilise the routing protocols of ad-hoc and wireless sensor networks, but this area needs to be studied and verified through specific case studies and applications.

SOFTWARE DEFINED NETWORKING (SDN)

Motivation for SDN

SDN was proposed in response to the requirements of modern computer networks. These include:

- Adaptability of the network to changing business requirements, policy and conditions
- Automation of policy implementation in order to avoid expensive and error prone manual effort.
- Maintainability: introduction of new features and functionality to the network with minimal disruption.
- Model management: Conceptual overview of the whole network as a model, rather than of individual components.
- Mobility: Accommodation for mobile devices, virtualisation etc.
- Integrated security in the network rather than as an add-on solution

- On-demand scaling by adding or removing network functions as required.

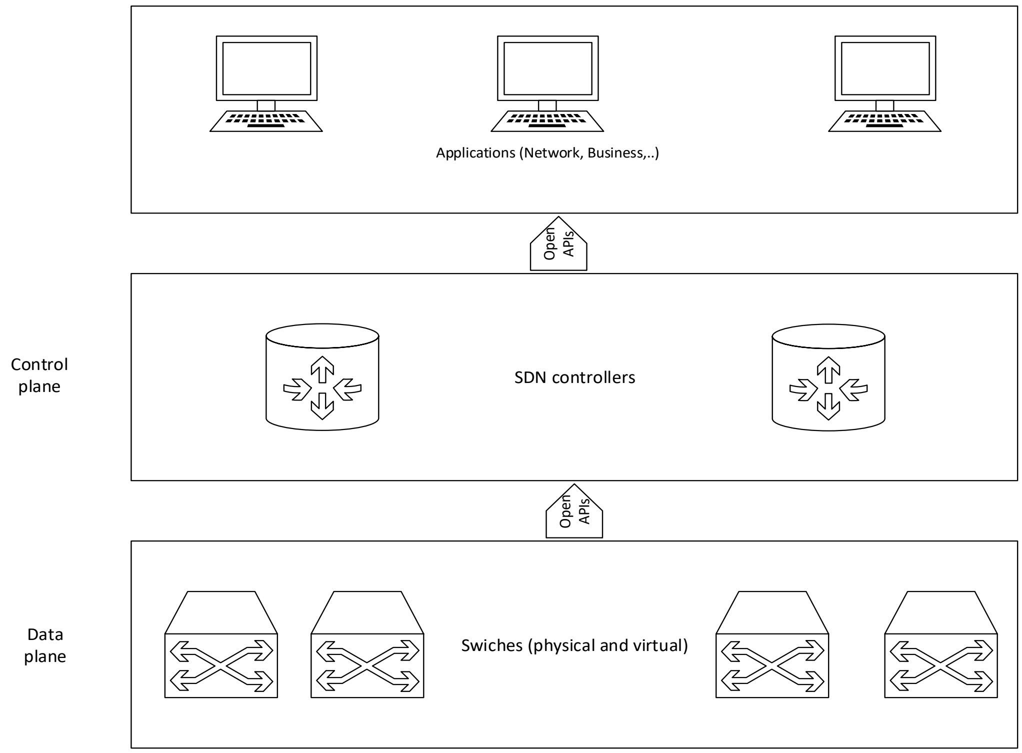

Figure 1-0-2 Overview of the SDN concept

Anatomy Of an SDN

As per figure 1-2, an SDN effectively implements one or more overlay networks on top of a single, easily managed network (the underlay network) that can be based on a Layer 2, or Layer 3 network topology.

A single SDN controller may control multiple logical networks. This approach decouples the forwarding domain from the physical technologies that implement them. Inside the forwarding domain, the SDN controller makes the forwarding, while the data plane is simply responsible for forwarding packets. Open interfaces (APIs) between the different layers allows the network devices to present a uniform interface irrespectively of the physical implementation. Similarly, APIs enable applications with networking requirements to communicate with the SDN controllers.

Routing Services for SDN

OpenFlow is a protocol between SDN controllers and network devices as well as a specification of the logical structure of the network switch functionality. The Routing Service consists of three modules:

- *Link Discovery*. The Link Discovery module is responsible for discovering and maintaining the status of physical links in the network
- *Topology Manager.* The Topology Manager builds and maintains the topology information in the controller and calculates the routes in the network. This module uses the neighbor database to compute the network topologies based on information received from the Link Discovery module.

- *Virtual Routing Engine.* The Virtual Routing Engine generates a virtual networking topology consisting of virtual machines that run traditional routing protocol. This module allows interoperability between SDN controller and existing networks consisting of traditional routers/switches running traditional routing protocols such as OSPF/BGP that were reviewed in earlier sections of this report.

In any of the above cases, the routing or switching protocol runs on the SDN controller. Each SDN forwarding domain may need to run its own routing protocol for interfacing with the external network. Or the SDN controller may run one or two instances and have virtual interfaces into each forwarding domain. The routing protocol can alternatively be implemented as an external application that talks to the SDN controller. In this case, routing updates would need to be forwarded from the switches to the controller and then to the external routing process. As per figure 4.3, the application API would be used by the external routing process to update the SDN controller's routing information base. A couple of core switches could be configured to run a routing protocol to exchange routing information with external systems. Internal to the SDN domains, the SDN controller would populate the forwarding information base. Only core switches would have routing information about the external destinations.

Network Function Virtualisation (NFV)

Network Function Virtualisation is the concept of virtualising network functions in software and running them in virtual machines (VMs). This decouples functionality such as Network address translation (NAT) domain name services (DNS), firewalls etc., from physical network devices. This allows network elements to become independent software applications that are flexibly deployed, and their capacity increased or decreased appropriately by adding or removing virtual resources (e.g. VMs).

Properties of Networks

A future realisation of PI must have certain desirable network specific properties in order to be acceptable by the T&L actors. These include resilience to disruptions and failures as well as adaptability to changing conditions. In other words, the PI must be a dependable T&L network. Such network properties are discussed below.

Failure and survivability

Resilient transport systems must be characterized and evaluated by the capacity to adapt to a variety of different stress scenarios. Current efforts in transportation resilience research have focused on framework development and quantification methods. These efforts include the specification of resilience indicators, such as total traffic delay, economic loss, post-disaster maximum flow, and autonomous system components. Other Resilience approaches to transportation networks use traffic network modeling to identify locations for critical buildings (for example, hospitals and fire stations), and to minimize trip distance and overall travel time across the system. Existing network resilience require information about resources for network behavior following a disruptive event.

In general, a service failure is a deviation of service from the desired system functioning to not meeting its specification or expectation. Network defenses may prevent challenges from triggering a fault and that many observable errors do not result in a failure. Disruption tolerance is one example of reducing the impacts of fault and errors on service delivery.

Fault tolerance

In systems engineering, fault tolerance relies on redundancy as a technique to compensate for the random uncorrelated failure of components. Fault tolerance techniques for both hardware, such as triple-modular redundancy, and for software, such as N-version programming exist. However, these apply to localised failures, and not to multiple, distributed correlated failures. Therefore, fault tolerance is necessary but not sufficient to provide resilience. Thus, fault tolerance can be considered a subset of network survivability.

Dependability

Dependability is the quantification of the reliance that can be placed on the service delivered by a system and consists of two major aspects: availability and reliability. The main measures of dependability are the MTTF (mean time to failure), which is the expected value of the failure density function, and the MTTR, which is the expected value of the repair density function. Availability is readiness for use, i.e. the probability that a system or service will be operable when needed, Reliability is continuity of service, that is the probability that a system or service remains operable for a specified period of time. These notions have been codified as standards by IFIP WG 10.4 and ANSI T1A1. The importance of availability and reliability depend on the application service. Availability is of primary importance for transactional services such as HTTP-based Web browsing. On the other hand, reliability is of prime importance for session- and connection-oriented services such as teleconferencing.

Robustness

Robustness is a network property that relates the operation of a system to perturbations of its inputs. In the context of resilience, robustness describes the trustworthiness of a system in the face of challenges that change its behavior. Robustness is often used as a synonym for resilience, survivability, and security.