MeshBox®

SmartMesh®

HyperMesh IoT Architecture for Value Internet

Use-Cases, Requirements and Architecture

2021 August

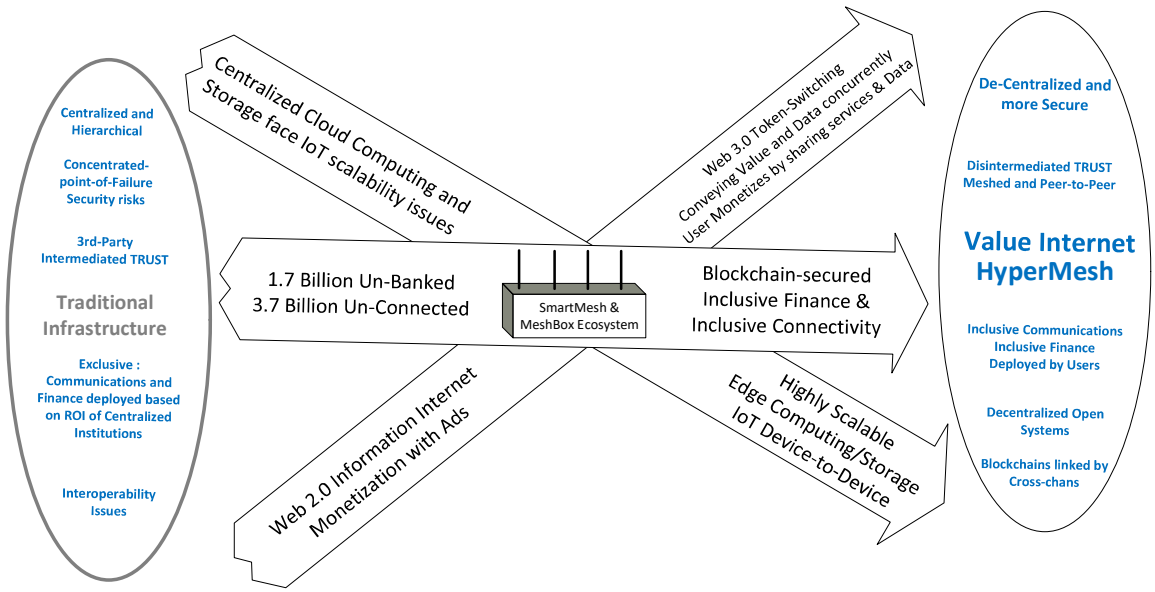
Table of Contents

1 Executive Summary

1.1 Internet of Everything, for Everyone

The Internet has evolved from Web 1.0 in the 1980s to Web 2.0 in the early 2000s – an **Information Internet** era. The emergence of Blockchain, Edge-Computing, Internet of Things (IoT) and Artificial Intelligence (AI) calls for the next stage of the internet.

Web 3.0 is a decentralized, open and trustless network where peers can transact directly with each other without centralized intermediaries, and users are the rightful owners of their data – ushering in the **Value Internet** era. Participants have been expanded to include machines as part of the **Internet of Things (IoT)**. While the Information Internet uses Packet Switching, the Value-Internet supports the concurrent transfer of both information and **value** based on **Token Switching**. The transformation to that of a Value-Internet can leverage the HyperMesh Architecture, benefiting society through inclusivity, giving users ownership of their data, and helping them to monetize their participation.



While the Internet is crucial to personal and societal well-being, a Digital Divide persists.

- 3.7 billion people are still not connected to the internet, with 63% of the rural households lacking internet access (UNESCO 2019)
- 1.7 billion people are without banking services (the Global Findex Database 2017), with many more lacking access to basic financial services like payments, borrowing and insurance.

1.2 IoT Opportunity and Challenges

IoT and associated applications are regarded as another wave of information technology progression in Internet evolution. According to IDC, by 2025 there will be approximately 42 billion

devices connecting to internet with 73 zettabytes of data having been generated. Coupled with AI, devices are becoming smarter and autonomous in decision making by interacting with each other. However, large-scale deployment of IoT is still constrained by several fundamental factors.

- limited scalability
- interoperability challenges with fragmented, proprietary technologies
- security and privacy risks
- exclusion of underserved peoples due to non-inclusive ROI analysis
- lack of incentivization to accelerate deployment

1.3 Scalability

1.3.1 Challenge: Applications running at Data-centers are far from IoT devices

A centralized network architecture, in which IoT applications are concentrated at data-centers, far from the IoT device, heavily loads the communication network. A fundamental issue lies with the existing client-server model. Here, device-to-device (D2D) communications must nevertheless jump through many hops to communicate with cloud hosting the application, creating unnecessary traffic loading and increased latency.

5G standards and Narrow-Band-IoT (NB-IoT) attempt to mitigate this issue with Cloud RAN. However, Telecom carriers are challenged to bill for D2D services because such communications do not run through their equipment, making monetization difficult, and stalling deployment. Also, for underserved and sparse areas, the bleeding-edge throughput and latency performance of 5G and small-cells is over-kill, sometimes being too heavy and costly to deploy.

1.3.2 Solution: Decentralization

The HyperMesh Architecture is proposed as a decentralized, synergized blend of networking, computing, data storage/access, blockchain, and cyber-physical IoT infrastructures. The HyperMesh™ is built from the ground-up, in a decentralized, fault-tolerant manner, incentivized by blockchain cryptocurrencies, powered by renewable transactive energy, and connected to the existing telco backhaul and/or world-wide satellite constellations with local peer-to-peer (p2p) mesh communication via MeshBoxes on the ground.

The need to support higher efficiency and lower latency for IoT applications drives the migration from cloud-based client-server models towards Edge-Networking, Edge-Storage, and Edge-Compute functions. Conventional blockchains are usually implemented in the data-center, due to heavy Proof-of-Work and large number of trivial transactions stored on traditional ledgers. On the other hand, the Spectrum Blockchain and Photon payment network are uniquely well suited for the edge and IoT, due to being lightweight, and for offloading trivial transactions from Spectrum to Photon.

Photon is a high throughput, low-latency layer-2 architecture, on top of the Spectrum blockchain, enabling quick P2P transfers, and scaling Transactions-per-second (TPS) performance, mitigating the classical TPS issue with blockchain transactions. The Photon Network also supports off-chain and offline transfers, i.e., without internet access.

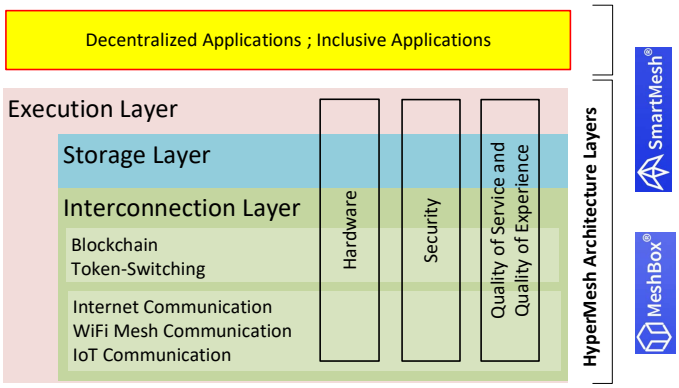
1.4 Interoperability – Standardizing Web 3.0 Value Internet

1.4.1 Challenge: A comprehensive architecture for IoT is lacking

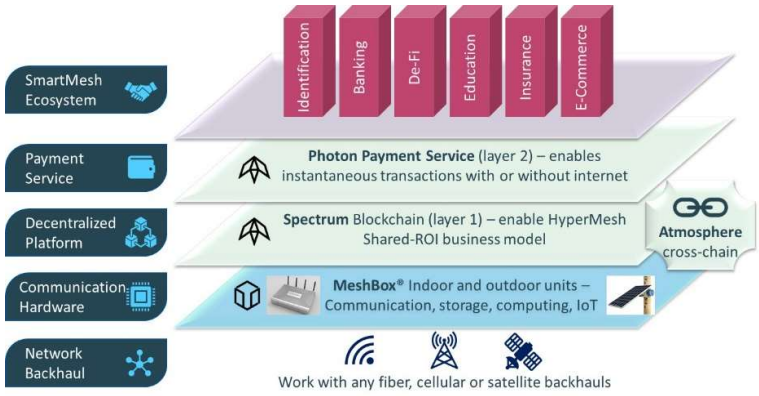
Traditional business models and technologies, applied to IoT results in multiple company's products not being interoperable and therefore competing with each other. Also, centralized architectures are inefficient and not cost-effective in supporting IoT applications, and are challenged to satisfy inclusive requirements.

1.4.2 Solution: Vertically integrated and horizontally broad architecture

A new comprehensive architecture is needed, consisting of decentralized hardware, communications, blockchain, storage, and task execution. Upon such an architecture, useful decentralized applications can thrive. The HyperMesh Architecture includes the Interconnect (composed of networking and blockchain transactions), decentralized storage, and decentralized execution layers. Spanning these layers, the HyperMesh hardware infrastructure, Security, and QoS/QoE functionalities are supported.



The HyperMesh Architecture encompasses a comprehensive vertical stack with mutually synergistic hardware and software components.



The hardware consists of wired and wireless (satellite) backhaul connection, WiFi wide-area mesh network nodes (using MeshBoxes), decentralized Edge-Storage and Edge-Computing hardware, IoT access points, and the IoT devices themselves.

The software stack includes QoS support for IoT and video streaming communications; security and transparency via hardware security platforms and Spectrum blockchain technology; Token-switching via Photon payment network; Atmosphere cross-chaining; and decentralized storage and decentralized computing layers.

The HyperMesh Architecture, with IoT security support, and Token Switching enables Web 3.0, can horizontally disrupt many industries including supply chains, finance, healthcare, and energy.

1.5 Security and Privacy

1.5.1 Challenge: Centralized architectures have concentrated-points-of-failure

IoT data is mission critical since important decisions are made with the data. The devices and the data which they generate must be securely communicated, stored without tampering, and timestamped. Also, users should own their own data and be able to monetize sharing of such data to applications such as big-data for machine learning and AI.

1.5.2 Solution: Blockchain based solutions with token-switching

Token-Switching is a new Value-Internet protocol based on the simultaneous conveyance of crypto-tokens and data securely. Such a protocol, used by on-chain and off-chain smart-contracts, enables applications in which users control how their data is shared, and the monetization of such sharing. IoT data transfers are secured crypto-graphically, simplifying the management of synchronization between information transfer and a token payment, associated with the information transfer.

Token-Switching adds a new dimension to the information internet by integrating value transfers in the same protocol as the data transfer. This results in an efficient billing function (and thus user monetization) built into edge and IoT applications. Security is enhanced in that DDOS and Spam attacks can be mitigated, due to the (optional) cost associated with the sending of data. The Who, What, When, Where, and How the Hypermesh Architecture achieves IoT security is discussed in this paper.

1.6 Inclusivity

1.6.1 Challenge: Exclusion of underserved peoples due to non-inclusive ROI analysis

There is a fundamental issue with the **centralized business models** for network deployment, where business decisions for infrastructure deployment are made by centralized service providers. For underserved communities that require much-needed internet-based services, service providers find it difficult to make a good business case to justify the high capital expenditure.

1.6.2 Solution: Priority placed on inclusive requirements

For inclusive applications, HyperMesh Architecture nodes must be light in terms of energy, storage, and deployment/maintenance costs. Due to the **intermittency** of the Internet and electricity in developing areas, the HyperMesh Architecture must be robust in providing **high-availability** support communication and e-commerce services in such challenging environments. The HyperMesh deployment must likewise be cost-effective and simple. In developing areas, some people are not well educated, which requires the Inclusive services and applications to be simple to use, with **automatic configuration and failover** in the presence of network faults.

1.7 Incentivization for deployment

1.7.1 Challenge: IoT Infrastructure deployment is expensive to deploy

IoT deployment is composed of the IoT devices, communications infrastructure equipment, cloud equipment, data gathering and processing, often with AI. Also, a billing function is needed, which is decoupled from the data communications in a centralized business model. Such a costly infrastructure is needed in order to start generating ROI. Yet, without the proper funding, it is difficult to begin infrastructure deployment.

1.7.2 Solution: Decentralized finance and token economics

The HyperMesh Architecture Layers support a **revolutionary decentralized business model for economic inclusiveness**. HyperMesh enables vast monetization opportunities, by enabling asset-light, application-rich and highly secured services, including Internet Access, Payment Networks, Decentralized Identification, Data Storage, Micro-Insurance, Transactive IoT, etc.

HyperMesh is designed to promote local economic development, in which community members share internet bandwidth and storage services with each other. Tokens are earned for providing such goods and services and therefore generate incentives and ROI to be shared between MeshBox owners, investors, and operators in the community. Such economic benefits have previously been exclusive to large enterprises and out of reach for micro-SMEs in the outdated centralized business model.

1.8 Whitepaper Organization

Section 2 examines IoT sectors and use-cases. Various Value Internet IoT use-cases are considered in detail, including Green Agriculture, Supply Chain, Smart Home, Unmanned Aerial Vehicles (UAV), Internet of Vehicles, electric power grid and transactive energy, and intelligent medical networks.

Based on the use-cases' needed functionality, Section 3 discusses the HyperMesh Architecture Requirements for IoT. From the Requirements, a solution for networking aspect of the HyperMesh Architecture is discussed in Section 4, including IoT, Edge-Networking, and Space-Ground Integration Network (SGIN).

Based on the use-cases, requirements, and the HyperMesh network architecture, an overview of the HyperMesh Architecture Layers is given in Section 5.

Who, What, When, Where, and How of IoT Security is discussed (in Section 6 ???)

The resulting business processes which are facilitated by the HyperMesh are examined in Section 6. Finally, Sections 7, 8, and 9 detail the Interconnect, Storage, and Execution Layers respectively.

2 IoT Application Use Cases

2.1 IoT Application Sectors

In the Internet of things, the combination of blockchain and other technologies can be effectively applied to the Edge-Computing architecture of the Internet of things. A new system Architecture design is needed, including consensus algorithm, online and offline smart contracts, location-based services, timestamping, data storage structure, etc. in order to efficiently address the pain-points surrounding IoT deployment. Issues related to resource consumption, scalability, cost, long processing time must be solved, while preserving the security and privacy for users and their data.

2.1.1 System architecture

The development of Internet of things architecture has gone through various stages, from server client to open cloud center, and then to distributed P2P applications via Edge-Computing. Traditional IoT applications based on cloud server suffers from inherent security risks. If servers fail or are attacked, performance and availability of applications will suffer. In addition, if IoT devices are attacked, and used to generate false or large amounts of data, it may degrade the whole network through distributed denial of service (DDOS) attacks, thus affecting network security. In contrast, a distributed P2P network architecture based on blockchain does not rely on a central node or cloud server, and transactions are protected and verified through various technologies such as cryptography, TEE, and encryption. Also, with Token-Switching, IoT-related messages can be required to carry tokens of value, which dramatically reduces the possibility of DDOS attacks, due to the cost of sending each message. Therefore, the case of malicious nodes can be mitigated.

The following use-cases are examples of how IoT applications can benefit society. Such use-cases will be used to generate requirements for HyperMesh, which will further drive the HyperMesh architecture.

2.1.2 UAV Control

Unmanned Aerial Vehicles (UAVs) face security, data privacy and other issues. Blockchain can provide a higher level of transparency, security, credibility and efficiency for UAV control. For example, on August 1, 2019, Wal Mart announced a pending patent named "cloning UAV Using blockchain", which aims to use blockchain technology to ensure the data integrity and security of UAV package delivery systems, realize digital signature through hash algorithms, and provide an encryption method for image and sensor data collection, A general, extensible and easy to manage UAV access control system is implemented based on the blockchain. It is believed that in the near future, such technology can be implemented on small consumer friendly UAVs and decentralized servers running on mobile devices such as smartphones.

2.1.3 Internet of vehicles technology

In the Internet of Vehicles, vehicles need to collect and share data to improve driving safety and increase traffic throughput. The introduction of blockchain technology addresses the issue that vehicle owners are reluctant to upload data to a centralized infrastructure because they are worried about single point of failure and data privacy. Several initiatives utilize blockchain to

establish a distributed database to manage vehicle data; deploy smart contracts to ensure the safety and efficiency of roadside units (RSUs) for data storage; and select more reliable data sources through reputation based data sharing schemes to improve data credibility. Vehicles can choose high-quality and reliable data providers to ensure the security of data storage and data sharing. The results show that, compared with the traditional methods, such schemes have great advantages in improving the detection rate of dangerously driven vehicles and ensuring the security of data sharing.

2.1.4 Electric power grid and transactive energy

In the future energy infrastructure, in addition to the main grid, distributed energy resources (DERs) and micro grids will also become important sources of energy. The combination of blockchain and smart grid creates a more efficient system to match supply and demand via energy trading in real-time (transactive energy) or through business processes (Renewable Energy Credits) [GridWise] [PowerMatcher] [TeMix].

Through a safe, transparent and distributed energy trading model based on blockchain, the monopoly of the traditional energy market is challenged. Existing projects, such as Energo, evaluate the possession and consumption of energy in the form of tokens, and adjust the trading rules and grid switching strategy through smart contracts. Based on blockchain and local microgrids, a decentralized system of clean energy measurement, registration, management, transaction and settlement is realized. At present, the system has been promoted and deployed in Southeast Asia and Australia.

2.1.5 Intelligent medical networks

Intelligent medical applications need a new generation of life science technology and information technology to be supported, in order to provide comprehensive, thorough, accurate and convenient services for patients.

Health care, especially the processing of electronic medical records, is an area which can benefit from IoT and blockchain applications. Effective sharing of medical data can improve overall health and reduce cost for patients. Medical data sharing is a sensitive topic, which is a pain point in the development of medical industry applications. This is mainly due to the need to protect the privacy of patients with sensitive personal information.

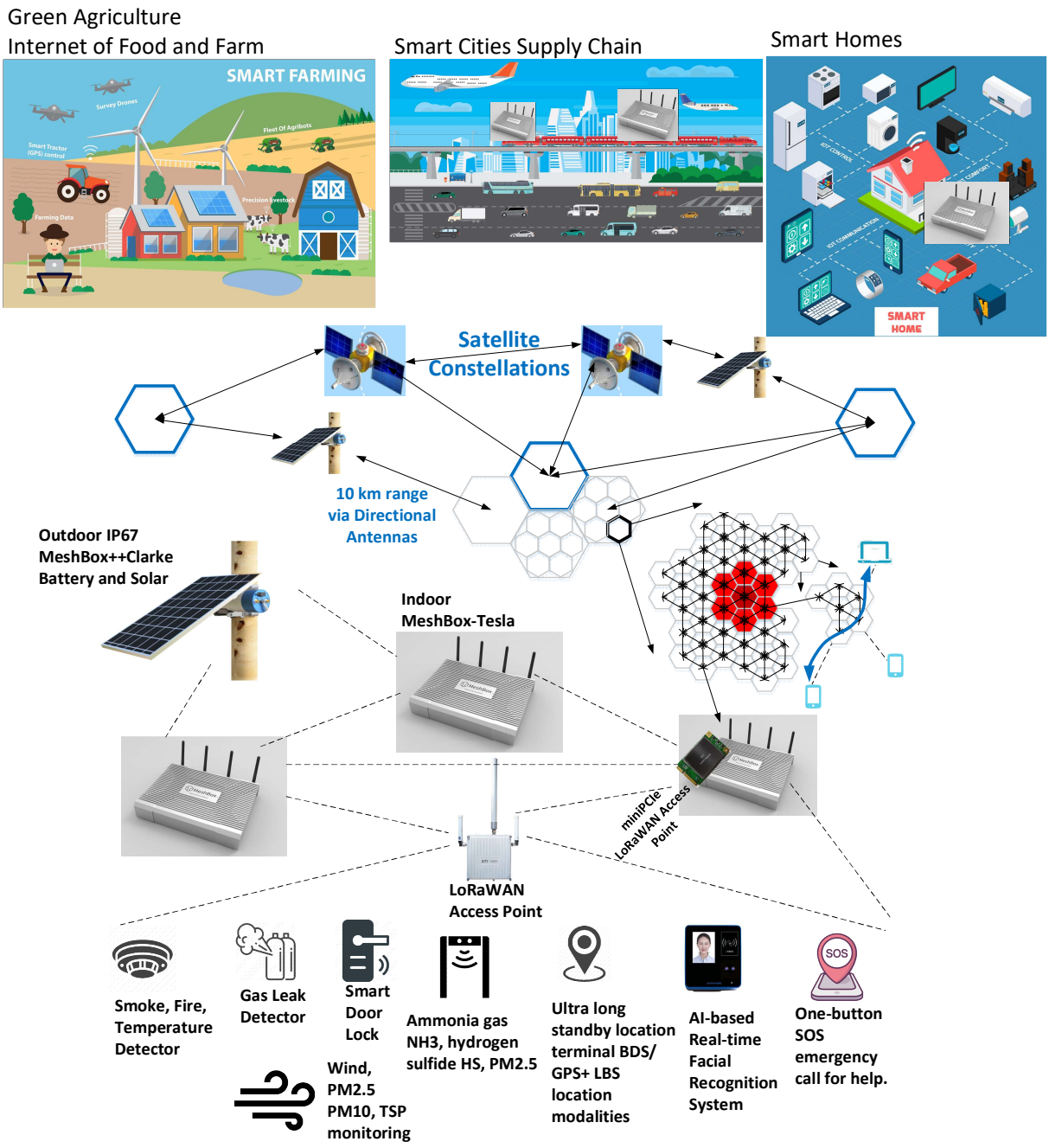
Blockchain and secured IoT can enable solutions to the problem of medical data sharing. The historical medical records of patients in different medical institutions can be uploaded to a HyperMesh Architecture Storage Layer, secured by blockchain technology, and different data providers can authorize users on the HyperMesh to access the data through their given permissions. This not only reduces the cost, but also solves the problem of trust. A typical application of blockchain in the medical field is chronic disease management. Medical regulators, medical institutions, third-party service providers and patients themselves can share sensitive information in a protected ecosystem, coordinate and implement the integrated chronic disease intervention mechanisms, and improve the control of diseases.

2.1.6 Application of blockchain in the Internet of things

Conventional Blockchains, by themselves, suffer from shortcomings when trying to address IoT applications, such as low Transactions per Second (TPS) performance, which cannot scale to

support the high data-rates across billions of IoT devices. Some consensus algorithms, such as POW for Bitcoin and Ethereum are not fully decentralized (often implemented in centralized data-centers), requiring heavy compute and storage resource and a great deal of energy to operate, and thus making them unsuitable for deployment in edge and IoT spaces.

The following details three other common IoT use-cases. The first starts with an agricultural, smart farm use-case. The second focuses on a supply-chain, which can, for instance, transfer the agricultural produce to the end-customer. The third looks at the end-customer, located in their smart-home.



All use-cases discussed will be used to derive the requirements and then develop the HyperMesh architecture.

2.2 Use-Case 1: IoT for Green Agriculture

2.2.1 Overview

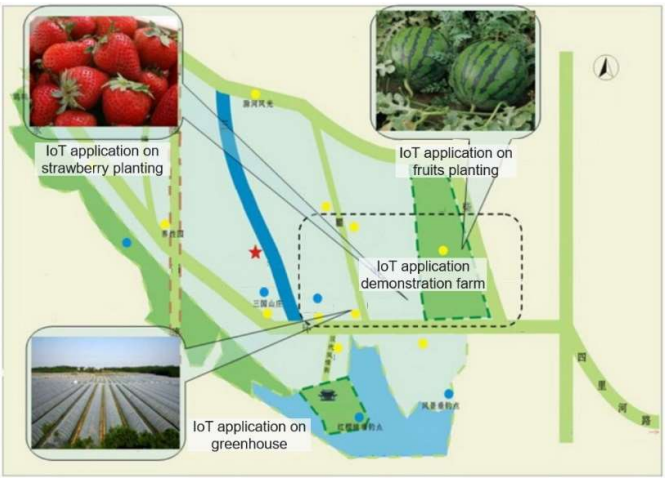
The following illustrates a Green Agriculture IoT system to realize intelligent and green agricultural production. The system solution uses various types of sensors, automated control equipment, IoT access points, and MeshBox wireless networking to form a smart IoT agriculture solution.

The solution is dedicated to improve agriculture production by monitoring air and soil temperature, humidity, light intensity, and carbon dioxide concentration. The collected data is saved on the MeshBox edge-server and potentially transferred to a cloud server. A distributed database is established to provide farmers and other managers with the measured data points, and allow for the interactive control of the agricultural process to improve production yield and quality.

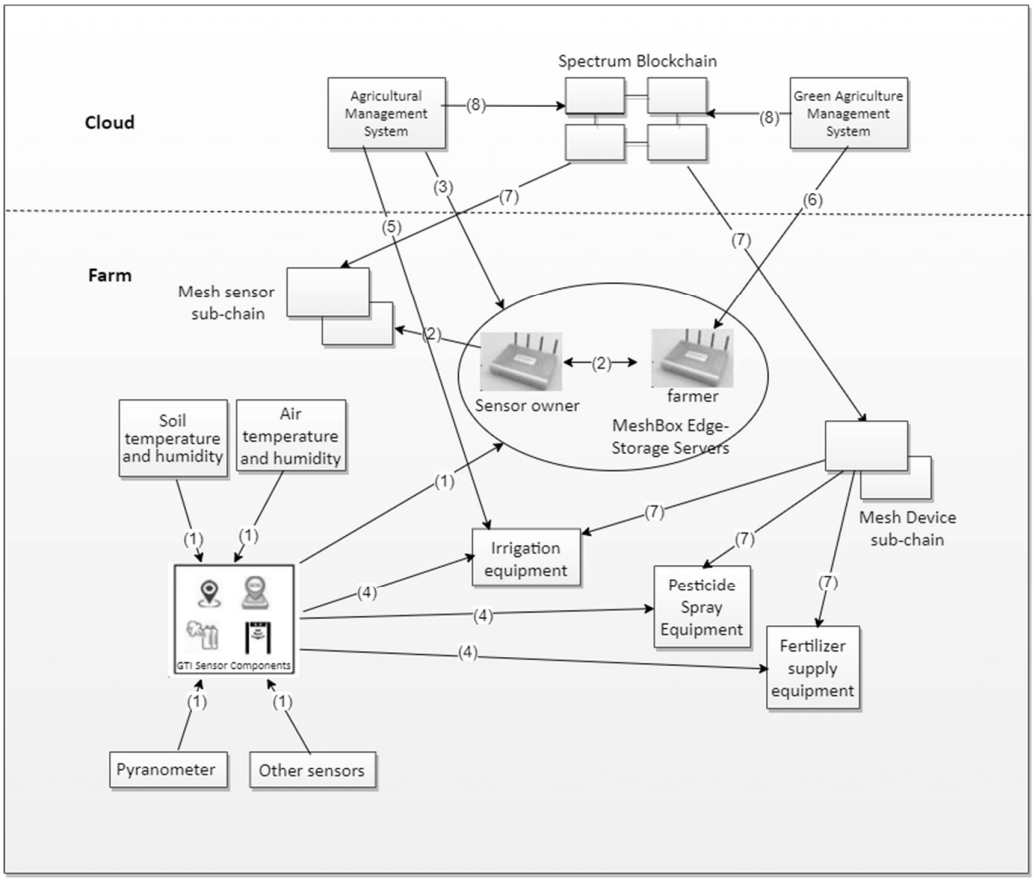
An IoT control system is needed to securely gather data, which benefits from blockchain technology. Based on data values, triggering and execution of tasks need to be performed. The goal is to reduce cost, such as green energy-saving management, improve yield, and run the system securely and efficiently.

The system components and configuration are as follows:

2.2.2 Functional Breakdown



The above shows an example of an agriculture farm, with the following logical diagram.



A brief description of the Use-case scenario follows, with the numbers below corresponding to the arrows in the figure.

(1) The **sensor nodes** regularly collect environmental data (soil temperature and humidity, air temperature and humidity, light intensity, etc.) and save the data in the MeshBox Edge-Storage servers.

(2) The **farmer** will send TransacKets to pay the sensor owner through the Photon. Thereafter, the **sensor owner** uploads the sensor data to the Edge-Storage nodes (MeshBoxes) with possible registration on the mesh sub-chain.

(3) The Agricultural Management System sends TransacKets to request and pay for sensor data, and processes the data.

(4) In a Decentralized management paradigm, sensors may report their measurements directly to the control equipment (potentially augmented with MeshBoxes), which has the intelligence to make decisions to start and stop operation (such as turning on/off the irrigation system).

(5) In a Centralized management paradigm, when enough sensor data values exceed a threshold, the Agricultural Management System sends the relevant TransacKet commands to the control equipment to perform the relevant tasks such as pesticide spraying, fertilizer

distribution, water drip irrigation, etc. When the measured data returns nominal values, the relevant TransacKet commands are sent to the control devices to stop operation.

(6) For Centralized management paradigm, the Green Agricultural Management System's expert system function synthesizes management strategies based on the purchased data to ensure that the farm's products are in a good state of growth. If the crops or farm animals face systemic issues such as illness, the farmers will be notified to take appropriate action as soon as possible. The expert system may sell consulting services to farmers (paid through Photon).

(7) In Centralized management paradigm, an oversight entity (such as Environment Protection Agencies or the Government) uses Atmosphere to collect the wide-area status using multiple mesh sub-chain sensor and equipment data for comprehensive observation.

(8) The Agricultural Management System and the Green Agricultural Management System can then use the data collected by Atmosphere in order to track green energy deployment, and control the equipment in order to mitigate the impact of agricultural production on the natural environment.

2.2.3 Interconnection Layer

The Green Agriculture IoT application benefits from the following components and functionality for the Interconnect Layer.

(1) IoT communication: Such as LoRaWAN, GTiBee, Sigfox, and other Low-power WAN (LPWAN) IoT communication protocols

(2) IoT sensors and control system: A variety of wireless sensor monitoring equipment such as air temperature and humidity; illuminance; carbon dioxide concentration; and soil temperature and humidity. Examples of IoT controls are water-saving irrigation water valves, automatic fertilization equipment, automatic spraying equipment, energy consumption management, etc.

(3) Edge-Networking: MeshBox WiFi router equipment and IoT Access Point (such as those supported by GTI IoT company) nodes support multi-hop mesh networking.

(4) Blockchain support of Token-Switching: Spectrum public main-chain, Mesh sub-chain, Photon payment network and Atmosphere cross-chain. TransactKets containing both data and tokens are transferred between Spectrum/Photon wallets and smart-contracts for IoT applications.

2.2.4 Storage Layer

The Green Agriculture IoT application benefits from the following components and functionality for the Storage Layer.

(1) Edge-Storage elements: MeshBox WiFi router equipment supports decentralized storage of Data on HyperMesh.

(2) IoT data cloud storage service: The Edge-Storage elements may transfer data to a cloud applicaton which gathers IoT data from various Edge-Networks and aggregates such information for machine-learning and AI purposes.

(3) Blockchain support of Token-Switching: Spectrum public main-chain, Mesh sub-chain, Photon payment network and Atmosphere cross-chain, support storage of IoT data on-chain and off-chain. Simple aggregation and dis-aggregation of data can be done by the Storage Layer.

(4) Agricultural system digital-twin: Gathering of geographical IoT information for machine learning, knowledge discovery, virtual reality and other technologies help in developing a digital-twin representation of the geographic area.

2.2.5 Execution Layer

The Green Agriculture IoT application benefits from the following components and functionality for the Execution Layer.

(1) IoT data cloud application: The (centralized) IoT application running in the cloud may perform real-time monitoring of agricultural production site status; long-term historical record keeping and predictive analysis of potential future issues; as well as remote status tracking and management of agricultural machinery and equipment.

(2) Blockchain support of Token-Switching: Spectrum public main-chain, Mesh sub-chain, Photon payment network and Atmosphere cross-chain, must support control task triggering scheduling, and execution, which supports the analysis of IoT data. Task triggering is based on received on-chain and off-chain messages and TransacKets (containing aggregated IoT sensor measured values).

(3) Green agriculture management system: Based on the agricultural site's digital twin, and the decision-making of the agricultural expert system, the farm owners can improve the production of crops and/or aquaculture products. Artificial intelligence, multimedia, virtual-reality and augmented-reality tools utilize the digital-twin of an agricultural site to help with production management and developing strategies for optimizing the agricultural business. Many farm processes such as fertilization, pesticide spraying, and the energy of agricultural machinery and equipment are managed and cost-optimized to increase revenue and mitigate adverse impacts on the climate and the natural environment.

2.3 Use-Case 2: IoT for Supply-Chain

2.3.1 Overview

The next IoT application discussed is the HyperMesh Supply Chain, which is a decentralized network composed of suppliers, intermediate processing/storage facilities, transportation, and consumers. Products and services are produced, sold, and physically transferred, from the initial supplier to the end user. A basic supply-chain system usually involves food or raw material suppliers, manufacturers, logistics companies, shipping, warehouse storage facilities, and end-customer retailers.

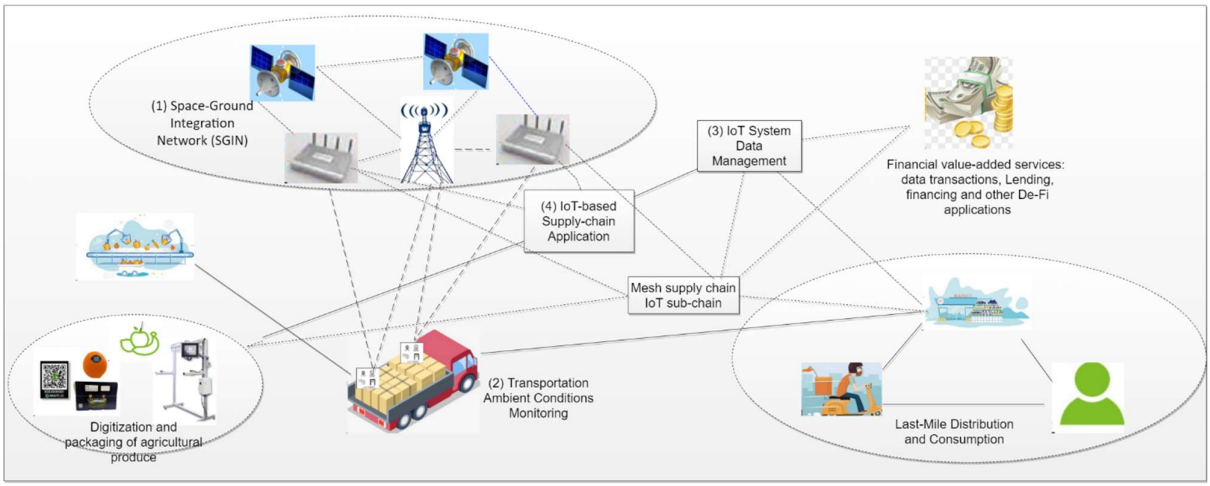
Compared to the Agriculture IoT application, a supply-chain adds the complexity of physical transport of items, and the need for multiple business entities to transact with each other.

At present, the inefficiency and opacity of the supply chain management system have been plagued by paper-based management and human errors. Other issues stem from:

- privacy protection hinders data sharing

- lack of traceability on the status of goods across the entire chain
- subjective determination on the quality of the goods (such as for perishables)
- difficulty in ensuring transaction legality and compliance
- lack of data transparency and dynamic adaptability to faults in the supply chain
- etc.

Some of the above pain points can be mitigated through blockchain privacy protection, anti-tampering protection of data via blockchain, blockchain trusted verification (including digital signatures), smart-contract automatic reconciliation, etc. Therefore, blockchains are widely applicable in the supply chain field. Specific to supply-chain use-cases related to IoT, HyperMesh with Token-switching supports useful features for end-to-end monitoring and visualization of the full supply-chain process.



The above example of an IoT system for supply-chain consists of the following parts:

(1) Space-Ground Integration Network: Due to the international scale of Supply-chains, various backhauls including satellite, cellular, and fiber networks are used with wide-area MeshBox WiFi and IoT networks to provide worldwide location-based services, observability and controllability of IoT and embedded machines.

(2) Transportation ambient conditions monitoring – sensors can be deployed in transport vehicles or container enabling remote real-time query on ambient conditions during the transportation of goods.

(3) IoT system control: Using the HyperMesh Architecture for IoT, the measurement and adjustment of temperature in a transport vehicle or container can be performed. HyperMesh must support data analysis and early warning triggers, and other distribution logistics feedback.

(4) IoT-based supply-chain application: Tracks all parameters of the goods being transported, including real-time IoT sensor readings on temperature, freshness, etc; as well as real-time contraints on the goods, such as perishable goods' production date, shelf life, and time

spent during each step of the shipment process. Tracked information must be correct, complete, accurate, effective and traceable.

2.3.2 Interconnection Layer

An IoT-based supply-chain benefits from solutions for the following, in terms of the Interconnect Layer.

- Wireless sensor network communication network (receiving, monitoring and early warning based on IoT data values. Examples include LoRaWAN and NB-IoT cellular networking.
- Multi-factor authentication of locations and parties which store and process goods.
- Location tracking including cellular location-based-services, visual tracking, GPS sensors, motion tracking; edge mesh network tracking via MeshBoxes, and LoRaWAN triangulation
- Fault tolerance and self-organizing network function to mitigate node failures
- Management and authentication of IoT devices which, for example, measure the temperature and humidity of the ambient environment to ensure that the products are kept in good condition (such as freshness of agricultural products).
- Transportation vehicles' ambient conditions monitoring

In scenarios such as product traceability and supply chain management, IoT technology, with various sensing devices are able to track, monitor, and relay key information about the status of the goods transmitted in near real-time. Decentralized Edge-Networking and Blockchain provides a secure environment for such IoT functions. Such a combination helps ensure the timely and comprehensive monitoring of the state of the supply-chain, to meet delivery expectations and minimize cost.

Goods in a supply-chain must be tagged in order to be traced. Also, the storage and processing entities must be authenticated in order to trust the assertions made by each entity on the status of the goods. A common method is to associate QR codes and/or camera data with the goods in order to provide accurate data on the (potentially varying) quality of the goods during the transportation process. On this basis, HyperMesh must support monitoring of environmental conditions (temperature and humidity) and dynamic adjustment of the ambient conditions during transit.

2.3.3 Storage Layer

The distributed storage, anti-tampering, and consensus mechanism of blockchain technology ensure that the key data, as well as the nodes which store and transfer such data between the various parties along a supply-chain are authenticated and trusted. IoT technology can improve the comprehensiveness of the data and support the HyperMesh Architecture. The combination of the two improves the data coverage and data authenticity in the upstream and downstream flows of the supply chain.

An IoT-based supply chain benefits from solutions for the following, in terms of the Storage Layer.

- Data related to observability and controllability during transit must be communicated to/from applications running at the Data-center cloud server. However, communications outages may occur, as a function of geographic location (moving out of communication coverage area) and as a function of time (such as during inclement weather for wireless

networks). In such cases, Edge-Networking nodes, such as MeshBoxes, placed strategically along a route, or within the vehicle, can gather and analyze the data. Then, when an internet connection is available, the MeshBoxes can transfer such data to Fog and/or cloud-based applications, if needed.

- Digitalization of products, such as through QR tags providing digital identity of agricultural products and packaging, to be transported over the supply-chain. IoT cameras capture QR codes and images of agricultural products and packaging to register and store digital assets.
- IoT data Storage Layer must developed in a manner which facilitates subsequent analysis and actions performed by the Execution Layer.

2.3.4 Execution Layer and related applications

The IoT observability and controllability data associated with the transit process are useful for business processes such as insurance and guaranteeing the timely delivery and quality of the goods.

An IoT-based supply chain benefits from solutions for the following, in terms of the Execution Layer.

- Based on issues which occur during transit, dynamic decisions related to changes in the remainder of the supply chain may be best made at the edge, close to the goods themselves.
- Regarding perishable and/or time-critical deliveries, adjustments can be made based on varying issues which arise. For instance, if any leg of the supply chain encounters additional delays, the route and/or the type of vehicle, and/or the processing times at intermediate warehouses for the remainder of the transit can be reduced.
- In terms of ambient conditions which can change due to equipment malfunctions, or the weather, relevant data is gathered and the values can be used to trigger certain remedial actions. For instance, the thermostat of refrigeration units can be adjusted, in order to adjust the temperature.
- In the case of refrigeration equipment malfunction, the driver of a transport vehicle can be directed to a nearby warehouse with refrigeration. All such data, which can be captured as a digital-twin of the physical state, can be analyzed and reviewed in a machine-learning application. The security of such a digital-twin representation of the supply-chain must be ensured via blockchain and associated technologies provided by HyperMesh.

In addition, the observability and controllability processes can be incentivized at a fine-level of granularity via Token-switching, supported by the Execution Layer. For instance,

- Each leg of a supply chain, operated by a company, can be rewarded immediately for delivering the goods on time and in good condition.
- Multiple resources e.g. warehouses, can bid, through Token-switching to win business by offering the best features, guarantees, throughput, and latency.
- Large amounts of IoT data, stored in the Storage Layer can be queried, with a token payment to Storage Layer for each query.

- Smart-contracts, both on-chain, and off-chain will execute transparently, securely and according to QoS scheduling, in order to mitigate the need for legal counsel to write legal agreements, and try to enforce such contract breeches through legal action (suits).

2.4 Use-Case 3: Smart Home

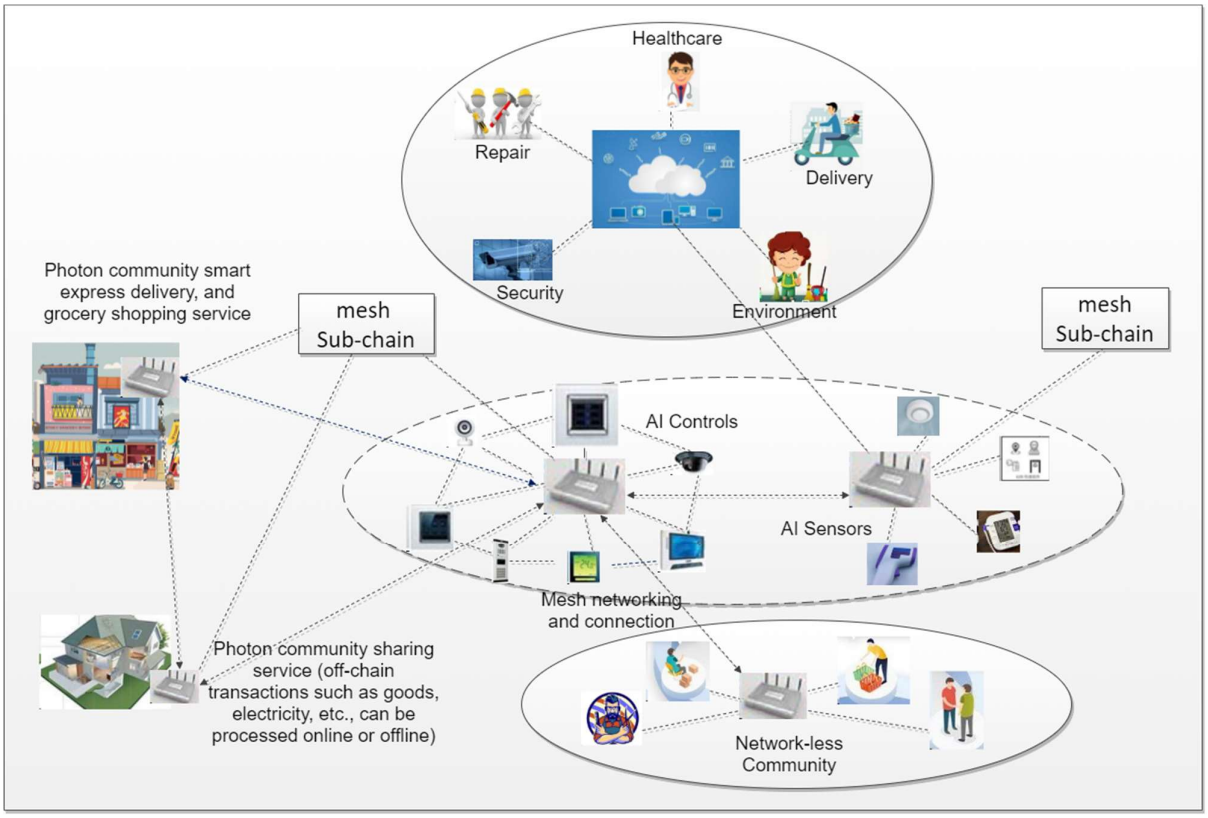
2.4.1 Introduction to Smart Home Ecosystem IoT

At present, smart home applications are different for each manufacturer, and work mainly on the IoT devices produced by each manufacturer. So, solutions are fragmented, do not interoperate well, such that a unified standard for applications does not exist. HyperMesh is designed to provide such a standardized Layered architecture.

Security is also incomplete, in that various family members need to have different permissions to access the home network. For instance, adults should have more permissions than children. Also, Smart homes are often not usable when the Internet is unstable.

Further, there are no applications which support the payment of tokens for services. And interactions between smart homes is missing. For instance, an electric vehicle may wish to pay a neighbor's solar panel to buy electricity if that electricity is cheaper than that available from the grid.

Similar to the Agriculture and Supply-Chain use cases, the Smart-Home use case benefits from IoT and blockchain enabled HyperMesh. However, the Smart-Home emphasizes end-user experience, which relates to Quality of Service (QoS) and Quality of Experience (QoE).



2.4.2 Interconnection Layer

The scope of the Smart-Home solution for Interconnection Layer includes:

- Home: A digital-twin of the state of the home is maintained. This is composed of potentially AI-controlled smart-appliances, lights, multimedia entertainment, electric vehicles, heating and air conditioning. These should be observable and controllable.
- Community: Neighbors in a community can share resources (internet backhaul, electricity, etc), with the associated token payments supported by the Interconnection Layer (blockchain and payment network).
- Home to Businesses: Family members can transact with businesses such as healthcare, delivery, environment, security, home repair, utility providers, etc.

2.4.3 Storage Layer

The scope of the Smart-Home solution for Storage Layer includes:

- Home: A digital-twin of the state of the home is maintained. This is composed of both data and value (e.g. tokens held in a wallet) pertaining to smart-appliances, lights, multimedia entertainment, electric vehicles, heating and air conditioning (HVAC), etc. Such state information should include both the observed values, as well as the control values.
- Community: A digital-twin of the states of Neighbors in a community and the history of such state information can be used for the Home to optimize its business with the community.
- Home to Businesses: A digital-twin of the states of businesses (usually companies selling goods and services to the Home) and the history of such state information can be used for the Home to optimize performance and value for the Home in terms of dealings with such Businesses.

2.4.4 Execution Layer and related applications

The scope of the Smart-Home solution for Execution Layer includes:

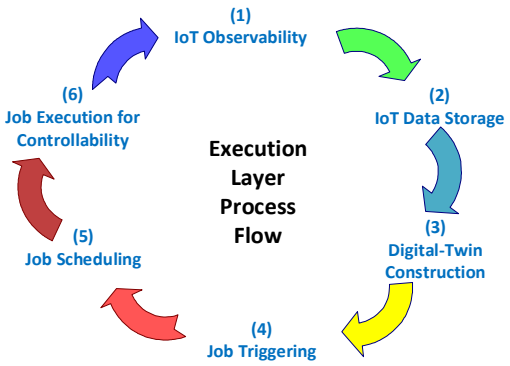
- Home: Using the Digital-Twin maintained by Storage Layer, the Execution Layer benefits from the triggering and scheduling of tasks based on Digital-Twin state variables. Upon such a Layer, various decentralized applications can be built, such as
    - Monitoring the health of the family members and visitors of the Home. If there are hazardous chemicals; poor or dangerous air quality (gas leak, carbon monoxide); or a disaster such as fire, flood, radiation or earthquake detected, appropropriate action can be taken to warn the household and alert the authorities (fire-fighters, hospital) for help.
    - Monitoring the welfare of the family members and visitors of the Home. If a thief or un-welcome person tries to cause problems, cameras can alert the household of the situation and alert the authorities (police) for help.
    - AI and Big-Data algorithms can learn about the preferences of household members, and proactively take action to make the household members more comfortable.
- Community: Execution Layer supports applications in which Neighbors in a community can share resources (internet backhaul, energy availability, etc). This is done with the Digital-

Twin maintained by Storage Layer and the associated Token-switching TransacKet-based communication and payment supported by the Interconnection Layer (blockchain and payment network). AI and Big-Data algorithms can be a big help in learning about the preferences of those in the household, and proactively taking action to optimize interaction with the community.

- Home to Businesses: Execution Layer supports applications in which Family members and/or smart processes in the Home can transact with businesses such as grocery stores, utility providers, and health care institutions, for the benefit of the household members. AI and machine learning algorithms intuit the preferences of those in the household, and proactively take action to optimize the benefits and costs for the household. For instance, if the AI entity for the Home determines that a particular kitchen soap is often purchased by the household, then when the soap goes on sale, the algorithm may make the purchase online.

2.5 Use-Cases Generalization for HyperMesh Process flow

Based on the above IoT application use cases, the generalized process flow diagram illustrates the basic functionality of HyperMesh.



(1) IoT Observability: IoT Devices gather data; transmit to LPWAN Access Points. This is done via a LPWAN wireless protocol such as LoRaWAN.

(2) IoT Data Storage: Edge-Storage nodes store data and may perform simple aggregation of such data, if so instructed.

(3) Digital-Twin Construction: A digital-twin of the Mesh-Subchain area-of-interest is constructed using Internet information, measured IoT data, and pre-determined triggering conditions.

(4) Job Triggering: When state variables in the Digital-Twin meet the triggering conditions, the associated Jobs become Triggered.

(5) Job Scheduling: Based on potential QoS and real-time deadline targets, the Task will be scheduled to run at some future point in time.

(6) Job Execution for Controllability: the execution of the Tasks or Jobs is performed, and results in Control actions (such as charging an electric-vehicle, or adjusting the temperature).

The above process flow will be used to generate the requirements and architecture of HyperMesh.

3 IoT Driven Requirements for HyperMesh Architecture

Blockchain technology offers trust for peer-to-peer transactions without a centralized third-party entity, which solves many security issues. For IoT, peers can be sensors, electric vehicles, solar panels, batteries, etc. Blockchain enables IoT peers to conduct business, such as double auction buying and selling, amongst themselves, under machine learning and AI guidance, and with human approval, or potentially autonomously.

Transactive IoT business models will dramatically reduce costs, optimize performance, and deliver killer applications to users, thus improving the Return-on-Investment for IoT deployment. Various decentralized data storage applications, with Blockchain technology address the authenticity and protection of data, such that users own their data and can control and monetize the release of such data. This forms the basis of the Value Internet.

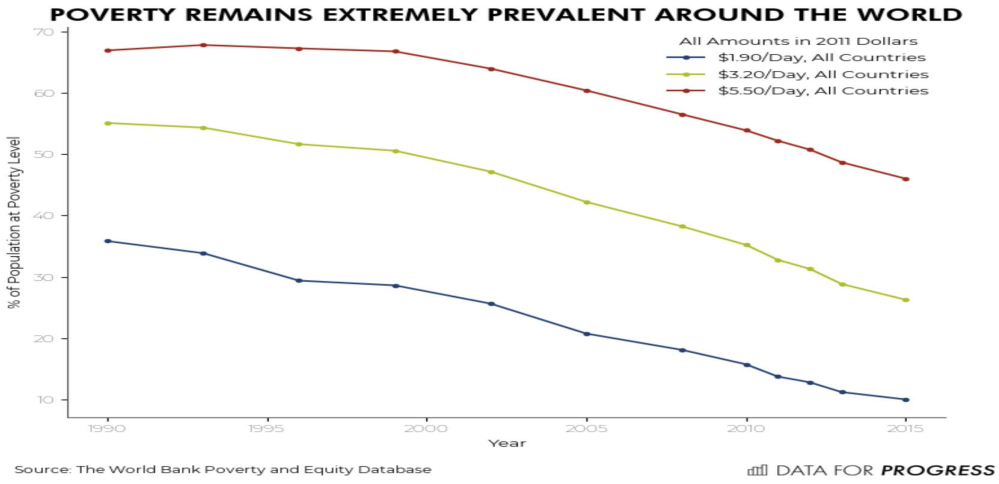
3.1 Blockchain Requirements for IoT Applications

The following compares SmartMesh Spectrum with other blockchains in terms of requirements which are important for IoT applications.

Requirements	Conventional Blockchains	SmartMesh Solutions
Fault-Tolerance to Communication Failures	Not supported. Bitcoin, Lightening, Ethereum and Raiden **require Internet connectivity at all times**.	**Photon runs** safely with **Intermittent connection** to the Internet (and thus to Spectrum)
Smart Contract Support	Supported by **Ethereum** and derivative Blockchains. Difficult to support on Bitcoin	Supported on Spectrum Ethereum smart contracts run on Spectrum.
Green, Eco-friendly consensus algorithm which can run on low-cost hardware	Bitcoin and Ethereum run on **Proof-of-Work** consensus, which is expensive in terms of hardware costs and energy. Therefore these are not suitable for Edge-Computing and low-energy IoT applications.	Spectrum's Proof-of-Capability consensus runs on low-cost CPU/memory resources. Photon is similarly **eco-green** and runs on **mobile smart-devices**.
Transaction costs	Proof of Work (PoW) requires expensive servers and large amounts of energy. Transaction gas fees are prohibitively high for inclusive applications.	Spectrum transaction gas fees are low enough to be feasible compared to poverty-levels. Allows for micro-entrepreneurship through sharing of Disk, CPU, Energy, and IoT Data.
Interoperability features.	Cosmos, Polkadot, Ubin	Atmosphere bridges Spectrum with other blockchains Wormhole Universal Channels bridges SMT/MESH with other Tokens

3.1.1 Inclusivity Requirement

[R1] Inclusivity: The HyperMesh Architecture components must be cost-effective enough to be feasible in inclusive environments in which community members live at the poverty level.



	Layer-1 Transaction Fee	Layer-2 Fees	Assumes
Bitcoin	~ $1 to $5 USD	Base fee + 1% of amount transferred = 2000 Satoshis + (0.01 * amount transferred) 1 Satoshi = 10 nano BTC	2020.06 Bitcoin at $10k. Min and Max Mining Fees
Ethereum	~ $0.40 to $2.10 USD		ETH from $1000 to $5000
SmartMesh Spectrum	$0.00000126 to $0.00126 USD	Photon Direct Transfers are Free. Mediated Transfers depend on Mediator nodes' policy. For instance, 0.01% of transfer amount	SMT from $0.003 to $3.00 Growing by 1000x

[R2] Potentially tiny transactions for IoT: For transactive IoT, machines are paying other machines for resources and services, which can be a very tiny amount (e.g. a few cents). This means the fees must correspondingly also need to be tiny. From the above, it is clear that the fees for Bitcoin and Ethereum transactions are too high for tiny transactions, which are needed for some aspects of IoT applications.

3.1.2 Light-footprint CPU and Storage

Spectrum blockchain is light-weight, running on low-cost mobile CPUs in MeshBox®. This is due to Spectrum's Proof-of-Capability consensus algorithm which securely, and fairly gives all Spectrum nodes an opportunity to Sign blocks. Since Spectrum is deployed with Photon from the

beginning, most would-be Spectrum transactions are instead moved to transfers over Photon, which minimizes the transaction loading on the Spectrum blockchain, therefore minimizing the disk-space storage requirements of Spectrum.

Thus, Spectrum (using a few GBytes) is much lighter than other blockchains, such as Bitcoin and Ethereum, which require hundreds of GBytes of storage in order to run the corresponding blockchains. With such a low storage foot-print, Disk Space on MeshBox® can be primarily used to store Data and Content, rather than being used to support the Spectrum blockchain.

The Photon network offloads most of the transactions from Spectrum by establishing State-Channels between Photon nodes and allowing for a high number of peer-to-peer transfers to take place between two Photon nodes, without needing to be recorded on the Spectrum blockchain.

3.1.3 Scalable TPS

Spectrum blockchain takes care of securely Creating and Settling State-Channels, and then allows Photon to manage all transfers between Photon nodes. Since only the creation and settling of Channels requires Spectrum consensus, and many transfers can flow between the Photon nodes which use the State-Chanels, a majority of transactions are off-loaded from Spectrum and results in the entire Spectrum + Photon architecture being able support millions of TPS.

As the number of MeshBoxes grows, the Spectrum+Photon TPS scales very well. This is because the peer-to-peer transfers between Photon nodes can usually execute concurrently, and are thus highly parallelizable. Thus, the more Photon nodes there are, the higher the TPS scales.

3.1.4 Multi-blockchain and Tokens

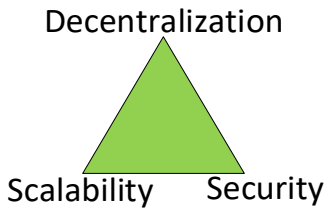
Not only does the SmartMesh® blockchain support the SMT (SmartMesh® Token) coin and MESH token, but other blockchain tokens are interoperable, including ERC-20 tokens from Ethereum and BTC from Bitcoin. This is done with Wormhole Universal Channel technology and the Atmosphere architecture being developed by SmartMesh®.

Due to the unique advantages of SmartMesh Spectrum blockchain in meeting the Inclusivity and tiny transaction amount requirements for IoT, this report assumes the use of Spectrum.

3.2 Tradeoff between Decentralization, Scalability, and Security.

Vitalik Buterin, inventor of Ethereum, said,

Blockchain systems have to trade-off between different properties. And it's very hard for them to have three things at the same time, where one of them is decentralization. The other is scalability, and the third is security".



The above challenge is termed, Buterin's Blockchain Scalability Trilemma.

The following characterize the Spectrum Blockchain and Photon Layer-2 architecture in terms of decentralization, scalability, and security requirements.

Goals	Blockchain (Layer 1)	Photon (Layer 2)
Security **Proportional to the cost needed to compromise the network**	Spectrum is permissionless, public Blockchain.	Photon Channel creation and settlement are recorded and secured on Spectrum. Photon uses signatures and receipts to secure Peer-to-peer transfers.
Decentralization **Degree of diversification in sharing of value and influence in blockchain operation.**	Spectrum Blockchain, is **permissionless**, running on ~750 nodes worldwide. Spectrum nodes are light (running on low-cost hardware with small disk space (few Gbytes)) are easily accessible due to low barrier-to-entry. Thus, highly decentralized.	Peer-to-peer transfers between Photon Nodes work even without connection to the internet, or Spectrum. Thus, highly decentralized.
Scalability of Transactions **Ability to support increasing number of nodes, at high transaction rates.**	Spectrum runs at low (~15) TPS, but with blockchain consensus security.	**Photon** is a Payment Channel Layer 2 Scalability solution running on Spectrum, implemented on mobile devices. Photon **Transactions per Second (TPS)** for tokens transfers is close to linearly **scalable** in the number of Photon Nodes.
Internet of Value: Token Switching version 1.0: **Secure Message and Token transfer in same protocol**	Spectrum transactions can contain both meta-data and tokens of value. However, blockchain is not efficient for storing large amounts of data in general.	Supports sending of messages (data) with the Photon Token Transfer itself (1) In-band message-transfer = Up to 256 Bytes of Message carried in the Photon protocol (2) Out-band message-transfer = Pointer (Hash) to Message (in decentralized File store), carried in the Photon protocol
Business Processes	Task Graph Represent business processes implemented as Smart-Contracts on Spectrum. Nodes = Spectrum Wallets, corresponding to business entities Arrows = Spectrum Transactions	Job Graph Corresponds to the Task Graph, in which Jobs, corresponding to Tasks, execute at high rate, with peer-to-peer security. Photon equivalent of Smart-Contract = Off-chain Smart Contracts Nodes = Photon node with Message Triggering, Job readiness, and Job execution. Arrows = Photon Channels

Goals	Blockchain (Layer 1)	Photon (Layer 2)
IoT related Business Processes (e.g Supply-Chain)	Specific Business Process mapped to physical world via IoT. Nodes = Spectrum Wallets, mapped to physical business locations (e.g. Farm sensors and equipment, household appliances, Warehouse equipment)	Nodes = Specific entity where goods are stored/processed. Arrows = movement of goods between entities, and the corresponding transfer of tokens. Job readiness and scheduling can be used for end-to-end real-time Supply Chains. Supply-Chains can be optimized for resource-sharing and meeting QoS constraints (e.g. deadlines).

3.3 Business and Cyberphysical Process Representation Requirements

In order to optimize business processes, graph theory is used to represent both logical and physical implementations, with the following requirements. Task Graphs describe logical processes, while Job Graphs represent finite executions of a Task Graph.

	Processes representable by Task Graph	Processes representable by Job Graphs
On-Blockchain execution	Required, via On-Chain Smart Contracts	Required for channel setup, deposit top-off, deposit withdrawal, and channel settlement.
Off-Blockchain execution	Not Required	Required, via Off-Chain Smart Contracts
Support logical (cyber) business processes	Required	Required
Support cyber-physical business processes between peers and machines, such as supply chains.	Not Required No QoS nor Fault-Tolerance guarantees	Required With QoS and Fault-Tolerance guarantees
Fractal Scalabilty	Required. At each level of the HyperMesh hierarchy, the architecture should be similar, so that solutions developed for one level can be applied to other levels.	
Trusted Security	Spectrum nodes are secured by Spectrum consensus mechanism	IoT nodes are managed by an IoT Access Point which adheres to a Trust framework
QoS		
Transaction throughput = Transactions per Second (TPS)	15	Highly scalable TPS, with scaling of hardware resources
Transaction finality latency	17 blocktimes (= 4 minutes = 17*14 seconds nominally)	1 second to a few seconds
Real-time guarantees	Not Required	Required

	Processes representable by Task Graph	Processes representable by Job Graphs
Delay Tolerant Networking	Not Required	Required
Fault-Tolerance		
With intermittent Internet	Not Required	Required, Photon transfers can execute when nodes are not connected to the Internet, once channels have already been established on Spectrum
With un-reliable communication channels	Required, Spectrum node pauses and synchronizes once connection is re-established	Required With distributed Photon network routing. IoT communication with Cloud support
Service availability in the presence of network faults	Required	Required

3.3.1 Interconnection Layer Requirements

The following are requirements for the Interconnection Layer, which supports transfers of TransacKets.

	Via Spectrum Blockchain	Via Photon Payment Network
On-Blockchain support	Required. Transactions and Execution are transferred between wallet addresses, via Spectrum	Not Applicable
Off-Blockchain execution	Not Applicable	Required. Execution are transferred between Photon nodes, via Photon
Support logical (cyber) business processes	Required	Required
Support cyber-physical business processes between peers and machines, such as supply chains.	Required	Required
Trusted Security	Spectrum nodes may be implemented on any computer, including MeshBoxes.	Interconnection Layer nodes secured by TEE mechanism.

	Via Spectrum Blockchain	Via Photon Payment Network
	Spectrum nodes are secured by Spectrum consensus mechanism.	Photon nodes may be implemented on any computer, including MeshBoxes. MeshBoxes support mesh communication, IoT access points, and interface to internet backhauls. MeshBoxes support Trusted Execution Environment
QoS		
Transaction throughput = Transactions per Second (TPS)	15	Highly scalable TPS, with scaling of hardware resources
Transaction finality latency	17 blocktimes (= 4 minutes = 17*14 seconds nominally)	1 sec to a few seconds
Real-time guarantees	Not Required	Required
Delay Tolerant Networking	Not Required	Tokens and TransacKets can be transferred, even with long communication delays between mediated transfer Photon nodes. This allows for Photon transfers on other moons and planets as well as between earth and off-world locations.
Fault-Tolerance		
With intermittent Internet	Not Required	Required, once channels have already been established on Spectrum
With un-reliable communication channels	Required, Spectrum node pauses and synchronizes once connection is re-established	Required With distributed Photon network routing. IoT communication with Cloud support.
Service availability in the presence of network faults	Required, Spectrum node pauses and synchronizes once connection is re-established	Required. Decentralized Photon routing supports route discovery from initiator to target.

3.3.2 Storage Layer Requirements

The following are requirements for the Storage Layer.

	Storage Layer
On-Blockchain execution	Required, Hash Key of Message contents is stored with Spectrum blockchain transactions in order to support timestamping and anti-tampering.
Off-Blockchain execution	Required. Message portion of Execution can be queued and/or stored and managed in the Digital-Twin representation.
Support logical (cyber) business processes	Required for Hash Key of Message storage on Spectrum.
Support cyber-physical business processes between peers and machines, such as supply chains.	Required The message portion of a TransacKet is stored in Storage Layer. The Token portion of a TransacKet is stored in a Wallet Address
Trusted Security	Storage Layer nodes are secured by TEE mechanism
QoS	TBD
Scalable amount of data storage capacity	Must be linearly scalable in amount of data stored, by adding more Storage Layer nodes (such as MeshBoxes)
Transaction finality latency	Latency must be lower than IoT Data generation frequency in order to keep up with periodic and/or streaming IoT data-rate.
Real-time guarantees	Required Storage and Retrieval rates must be guaranteed for real-time applications
Delay Tolerant Networking	Required for both Spectrum and Photon support of Storage Layer.
Fault-Tolerance	
Service availability in the presence intermittent, unreliable Internet backhaul connection	Required Edge-Storage in MeshBox mesh network should be supported
Service availability in the presence of Edge Wi-Fi mesh network faults	Optional Tradeoff between cost (hardware overhead) and availability requirements for Edge-Storage in MeshBox mesh network.

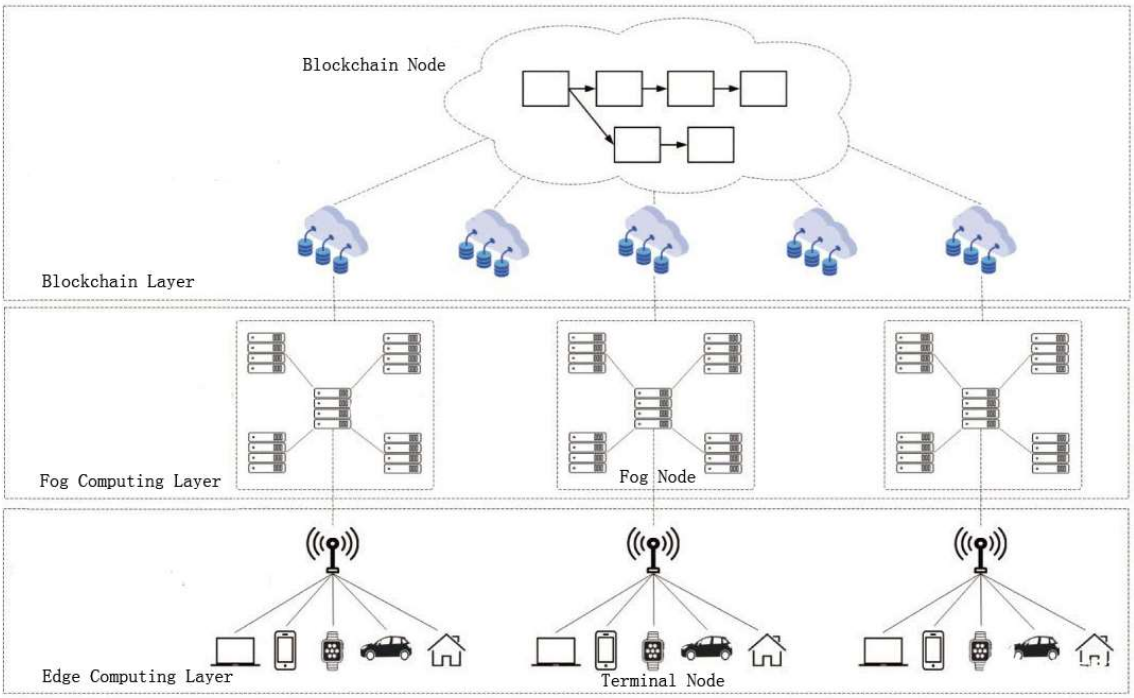
3.3.3 Execution Layer Requirements

The following are requirements for the Execution Layer.

	Execution Layer
On-Blockchain execution	Required, Via Spectrum Smart Contracts
Off-Blockchain execution	Required. Via Off-Chain Smart Contracts and Photon.
Support logical (cyber) business processes	Required. Via Spectrum Smart Contracts.
Support physical (cyber-physical) business processes between peers and machines, such as supply chains.	Required Execution Layer supports • Job triggering based on received (input) Execution, • Job scheduling for IoT QoS guarantees, and

	• Execution, with the transmission of (output) TransactKets. The Token portion of a TransacKet is stored in a Wallet Address
Trusted Security	Execution Layer nodes are secured by TEE mechanism

4 IoT and Edge-Networking with SGIN



IoT Network Layer Based on Blockchain

In the diagram, in the Edge-Computing layer, the IoT devices are connected via an IoT access point via a Low-power WAN (LPWAN) narrowband network. MeshBoxes can also be used at this layer to connect the IoT access points to the Internet and/or the Fog computing layer. The Fog-computing layer is sometimes defined as being the same as the Edge-Computing layer, which is a possible configuration. In this case, the Fog-computing layer is discussed as a decentralized higher layer, which has more powerful servers, which are capable of higher performance computing power than the Edge-Computing nodes. The blockchain layer, which runs on the Internet is shown at the top.

4.1 IoT Necessitates Edge-Networking

This section discusses how IoT relates to Edge-Networking. The issues with traditional networking architectures, both wired and wireless are compared and contrasted with the proposed Edge-Networking architecture. A Space-Ground Integration Network is introduced as an Internet backhaul connection technology.

Edge-Computing is the product of network evolution and the development of cloud computing technology. Its purpose is the extension and expansion of cloud computing beyond the centralized data center, which takes "edge-intelligence", "device-to-edge collaboration", and "edge-to-network collaboration" as the core capabilities. Blockchain technology provides security, while Edge-Computing supports high efficiency and availability. The two are integrated to achieve resource sharing, optimal configuration, security, and trust for business transactions and other collaborative applications.

The functional integration of Blockchain + IoT + Edge-Computing is mainly reflected in the following aspects.

4.1.1 Edge-Computing provides resources for blockchain services

A Blockchain enabled application can be deployed on an Edge-Computing Layer to provide blockchain services for network and industrial applications. In terms of resources, blockchain nodes and applications are rapidly deployed on Edge-Networks in the form of software, enabling the sharing of Edge-Computing node resources between business applications. For communications, blockchain and P2P Content Delivery Network (PCDN) technology are used to cache user account, data, other user data and business data to the edge nodes, so as to improve the communication efficiency and reduce the data transmission delay. In terms of capability, the HyperMesh Architecture, using blockchain, deployed at the edge nodes can serve as a general vertical solution of "information + trust".

4.1.2 Blockchain can provide trust for Edge-Computing.

With the help of the blockchain services built into edge nodes, the isolated islands of information between diverse Edge-Networks can be linked in terms of both information and token exchange, to produce cross network synergy. A sub-blockchain can run on Edge-Networking nodes (such as MeshBox) to establish the integrity of transactions for each Edge-Network as well as the decentralized authentication for data and tokens exchanged between diverse Edge-Networks, each potentially running a separate sub-blockchain. Such an HyperMesh Architecture can be used by applications to improve their commercial value.

4.1.3 Multi Edge-Networks data synchronization

Blockchain can aid in securing data storage, as well as storing some small amounts of data directly. Traditionally, IoT data is sent to the data-centers, where IoT applications process the data, and then push the decisions back down to the IoT control devices.

In Edge-Networking, Business applications require data synchronization between various locations (such as in a supply-chain) which are each served by separate, local Edge-Networks. IoT Access Points, Edge-nodes and Fog-nodes complement the data-center cloud nodes to store and process data, locally.

A blockchain distributed ledger, executing smart-contracts is used for maintaining consistency control for data synchronization and may be used for data authorization, usage tracking and data consistency verification. A message-passing programming model is considered, in which messages, containing data are transferred between tasks/jobs. Small amounts (preferably) of data can be transferred In-band, directly carried in a transaction on the blockchain, or transferred through Photon. Such In-band data can also be bound to a token, carrying value, in the case of using Token-switching via Photon.

Small and large amounts of Data can also be transferred Out-band, in which a pointer (such as a hash) to the data is transferred on the blockchain, or sent through Photon (as a TransacKet). When the pointer is transferred on chain, the data itself can be transparently registered, timestamped, and protected in terms of immutability (since modifying the actual data will result in an inconsistent hash with that stored on the blockchain).

4.1.4 Resource sharing in Edge-Computing

An Edge-Computing enabled HyperMesh maintains compute, storage, and payment network resources, which can be monetized, by dynamically allocating such resources to IoT applications. A third-party IoT application (such as a supply-chain) can submit a task or job graph to be executed on an Edge-Network, and the required performance metrics to be met (throughput and latency) to the HyperMesh Architecture. Through a centralized or decentralized scheduler (involving blockchain consensus), such a task/job graph will be mapped to the physical IoT and network resources in order to ensure that the performance metrics are met. The billing and payment for such resource reservation is also made via Spectrum or Photon.

Billing is based on the space-time utilization of the resources, including which resources are used, time duration, and duty-cycle, in order to support the most efficient sharing of such resources. In the case when any resources are over-subscribed (more demand than capacity), IoT applications may pay a higher price to receive preferential treatment.

4.1.5 Terminal edge device security certification

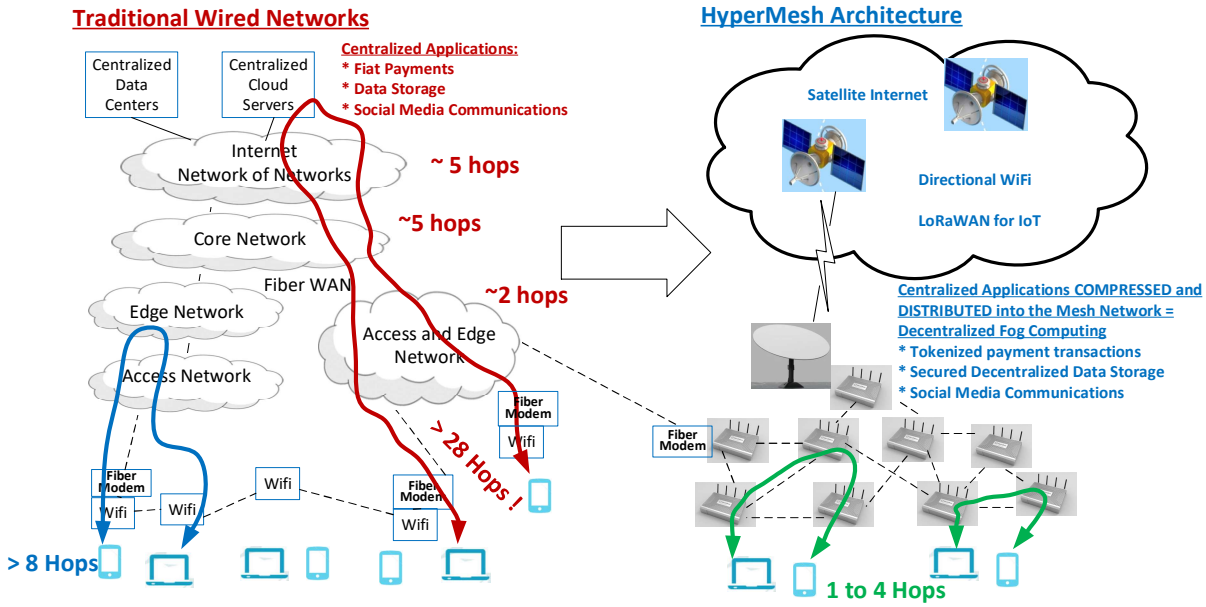
Each IoT device must be authenticated, and certified, in order to trust the data which it produces. The HyperMesh IoT Architecture must build a device identity authentication system based on the blockchain to ensure security for IoT applications.

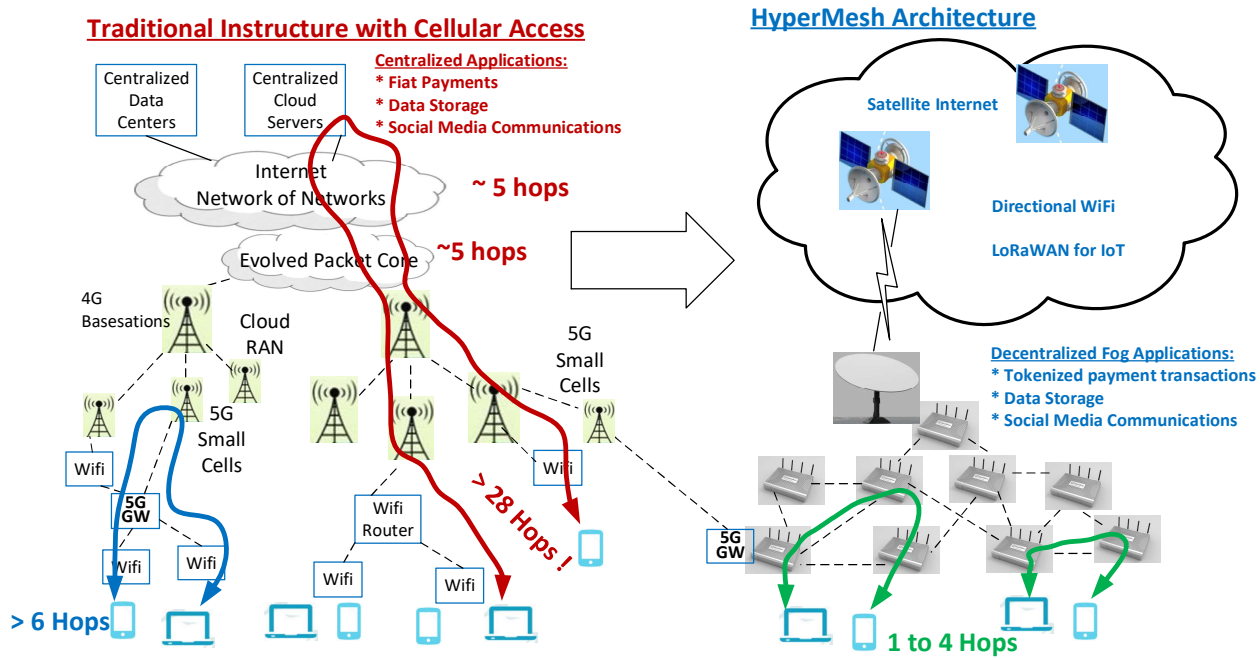
Each device manufacturer can authenticate the digital identity of deployed devices onto the blockchain, via stored certificates for such devices. The private key and digital signature of each IoT device is submitted to an Identity Authentication system, enabled by blockchain. After decentralized entities validate the IoT device, the IoT device's certificate is recorded in the blockchain. Edge-Computing applications also write their authenticated certificates onto the blockchain. After the connection between the user's terminal and the Edge-Computing application is established, the application can query the Identity Authentication system for IoT devices participating in application to ensure that all such IoT devices can be trusted.

Access authentication needs to be established between user terminals (cellphones) and Edge-Computing applications, and between Edge-Computing applications and cloud computing services. Thereafter, the data which the IoT devices generate must be secured as well. The secure exchange of data between diverse Edge-Networks can be done via crosschain technologies such as Atmosphere.

4.2 Edge-Networking for IoT

The following illustrates the HyperMesh Architecture, as compared to traditional Wired, and Cellular networks.





	HyperMesh Architecture	Traditional Architecture with 5G Cellular Access	Traditional Wired (Fiber) Architecture
Target Applications	Can be the last mile Access and Fog Network	Ultra-Low Latency Reliable Communication for advanced comm (gaming,	Very high Data-Rates compared to wireless.

	for Wired and/or Cellular networks Cost-effective for most commonly used applications (surfing, video streaming). Adequate access network latency for common applications ~10's msec.	VR, AR, MR, remote robotic surgery). Bleeding-edge QoS Access network latency < 10 msec	Can use wireless networks for last-mile, after Fiber plant terminates. Fiber need not be deployed all the way to subscribers' residences (can stop before last mile) to save fiber deployment costs
Wireless spectrum costs Deployment costs	WiFi uses Free (unlicensed) ISM band, with new 6GHz spectrum for WiFi 6 2.4GHz 2x2 + 5GHz 2x2 + 6GHz 4x4	Expensive, licensed cellular spectrum, for reduced interference advantages. 3G/4G spectrum, plus 5G spectrum up to 60 GHz.	Fiber and power cabling installation is very expensive, difficult to deploy, and not cost-effective for rural areas and islands.
Deploy-ment	• Much lower CAPEX and OPEX costs. • CAPEX < ROI ; OPEX < ROI, so can be deployed everywhere. • Outdoor nodes are easily deployable (within ~hours) portable (integrated solar and batteries) and thus need no wiring. • Mesh Topology: Outdoor and Indoor nodes mesh seamlessly and connect to Internet backhaul for maximum sharing.	• For the 3.7 Billion people without Internet, carrier-grade reliability and QoS is overkill and too expensive • CAPEX and OPEX > ROI in many remote areas. Thus not justified. • Base station and small-cell deployment require detailed simulations and tuning. Heavy and complex in hardware and software costs. Lengthy (months) deployment time.	• Usually high deployment costs due to trenching and laying wiring conduits. • Difficult and costly to lay fiber between small islands and to remote areas
Backhaul	• Any modem (fiber, cable, DSL) ; Cellular SIM support • Satellite Internet (dish modem) well suited for remote areas. • LEO Starlink supports lower	• 5G Fixed Wireless Gateway backhauls to base station/small-cell. • Conventional wired backhauls with modem. • High latency with tens of hops needed	• Fiber network connects seamlessly with Backhaul. • Conventional wired backhauls with modem. • High latency with tens of hops

	latency than Fiber networks due to speed of light in a vacuum being 1.5x faster than that in fiber. • 1/10 to 1/100 number of hops for Fog applications through mesh network reduces latency.	to reach Data Center.	needed to reach Data Center. • Higher, 1.5x latency (versus free space inter-satellite communication) for fiber
Data Rate	• WiFi 6E increases from 100's Mbps to ~ Gbps data-rate	• 10 Gbps to 100 Gbps (target) per base station	• Multiple Tbps. Theoretically 50 Tbps. • Much higher data-rate with DWDM, and multiple fibers in conduit.
Availability	• Lacks carrier class reliability and QoS, but this is unnecessary for people who don't have any internet. • Auto configuration and re-configuration when any nodes go down. • Due to Edge-Networking features, most local mesh network services available even with intermittent Internet.	• Carrier-grade hardware provides excellent QoS and reliability. But requires very complex and expensive OAM. • All-or-Nothing: Centralized applications require entire network infrastructure and connection to Cloud Data Centers at all times in order to function.	• High availability and reliability due to wired infrastructure. • All-or-Nothing: Centralized applications require entire network infrastructure and connection to Cloud Data Centers at all times in order to function.
Over-the-Top Applications	• Supported directly within MeshBox HyperMesh, allowing operator to monetize additional services (Fog Data-Storage, Fog Computing, Payment Networks,	• Out of reach for operator. • OTT Services monetized by centralized institutions (google, YouTube, banks for remittances, etc)	• Out of reach for operator. • OTT Services monetized by centralized institutions (google, YouTube, banks for remittances, etc)

	Transactive Energy) • DeFi model allows third parties to invest CAPEX costs		

4.3 HyperMesh Space-Ground Integration-Network (SGIN)

The following gives an overview of the Fractal network architecture of the HyperMesh Architecture. Due to the remote areas which must be covered by the HyperMesh, a Space-Ground Integration-Network (SGIN) is needed.

- **Space-Internet** for world-wide coverage, even in the remotest areas such as mountains, islands, rivers, oceans, and forests.
- **Ground Network**, linked with satellites, and providing Edge-Computing and Edge-Storage, to maximize the efficiency of the Satellite link.

Satellite Internet, SmartMesh and MeshBox technologies are synergized through the following:

- Satellites provide world-wide internet coverage, which is especially useful for non-urban, islands, rural, in-accessible, on-sea, and in-air locations.
- Satellites links are expensive, and must be optimized to best serve the people in remote areas.
- A wide-area WiFi mesh network (composed of MeshBoxes) scales the Satellite data-rate significantly through the use of Edge-Computing, Edge-Storage (caching), leveraging time-locality and space-locality of content accesses.
- Economy of Scale benefits using high-bandwidth satellite links, connected to low-cost ground-based Wifi Mesh networks for localized, last-mile deployment, even to far-flung locations and developing countries.

- Ground-based Mesh-networking provides high-bandwidth, high-density and sparse coverage, for both Indoor and Outdoor locales.
- Mesh network covers Non-Line-of-Sight (NLOS) locations, which Satellite signals do not cover well. Tradeoffs must be made due to higher satellite carrier frequencies providing higher data-rates, but are more susceptible to attenuation from physical obstacles and weather patterns. For instance, Satellite signals are attenuated significantly in passing through the adverse weather and atmospheric conditions, and are not expected to penetrate much into buildings.

Please see the Appendix and [SGIN] for more details.

5 HyperMesh IoT Architecture Overview

This document details the use-cases, requirements, and architecture of the **HyperMesh Architecture**, upon which **IoT applications** can be built.

Note that the HyperMesh Architecture Layers are not necessarily the applications themselves, but provide many useful tools and services upon which applications can be built. The HyperMesh Architecture makes certain (real-time QoS) performance guarantees and integrates autonomous constructs which help simplify application development and interoperability.

The following are brief definitions of some of the terms used in this document.

Tokens = A crypto-graphically protected digital representation of assets or identity.

Token Switching = A set of protocols which enable the exchange of user information and an associated intrinsic value of that information. When the user shares such information, there is a value accrued to the user in exchange for such sharing.

Micro Small and Medium Enterprises (MSME)

Digitalization and Identity Certifier = Technology provider for digital identity for people and secured digitization of cyber-physical items. Can be a group of decentralized entities.

Digitalization Program Operator = In a centralized approach, this is potentially a governmental, NGO, or banking entity which runs

HyperMesh = Next step of Information Technology composed of networking, and cyber-physical IoT, secured by blockchain technology.

TransacKet = Carried in Token-switching protocol. Represents the simultaneous conveyance of

- Transactions (holding Value, in the form of Tokens) and
- Packets, which contain messages,
- via Token-Switching protocols.

Spectrum Blockchain

Photon Payment Network

Task Graph = Used to represent business processes which can be executed as a Spectrum smart contract. Possibly Infinite executions. Not mapped to specific physical resources.

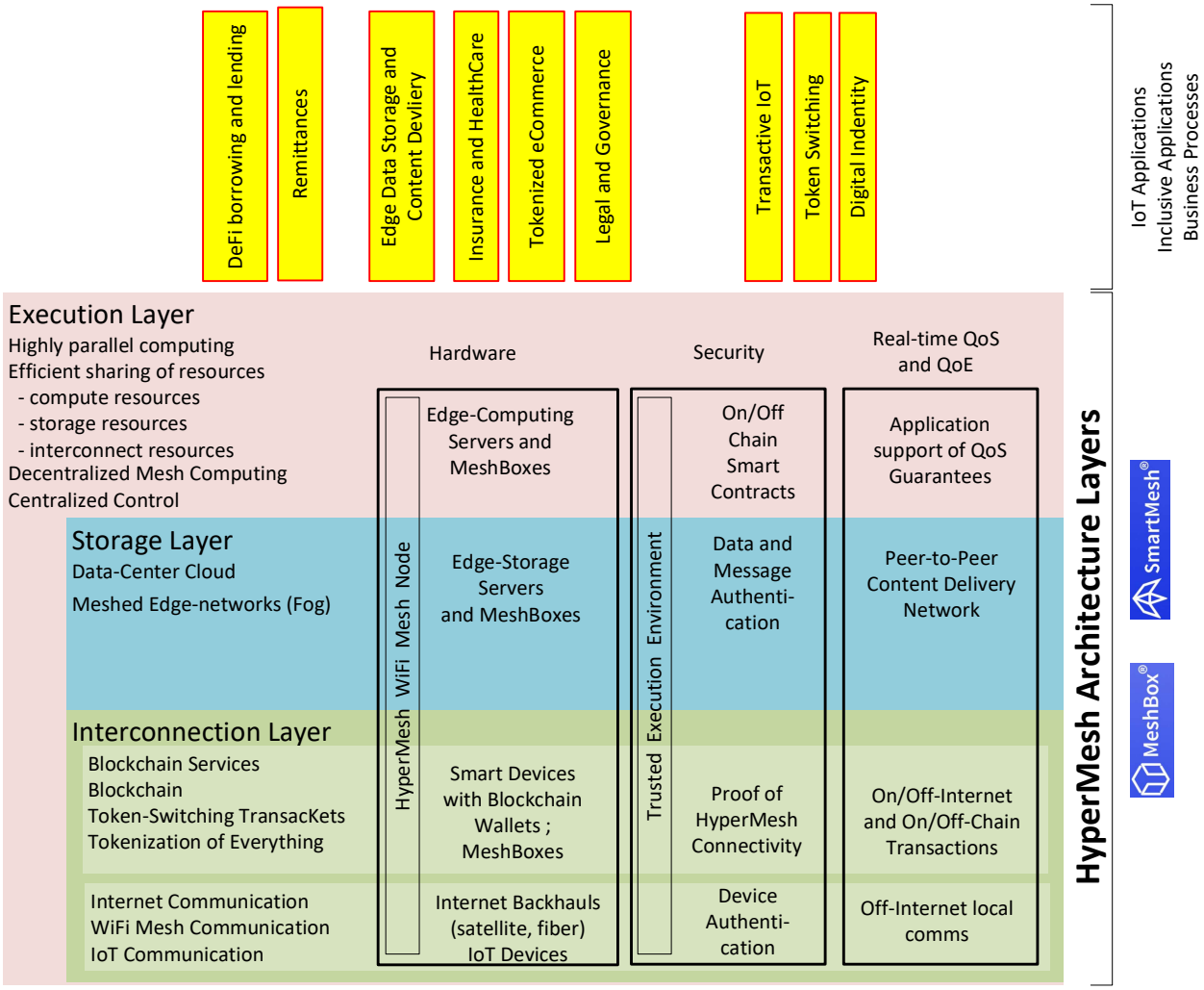
Job Graph = Finite (N) executions of Task Graph. Jobs Graphs are used to represent business processes which can mapped to a pool of physical resources, executed on Photon, with performance guarantees.

The proposed HyperMesh architecture is composed of the following Layers.

(1) HyperMesh Architecture Interconnection Layer

(2) HyperMesh Architecture Storage Layer

(3) HyperMesh Architecture Execution Layer



5.1 HyperMesh Architecture Interconnectivity Layer

The Interconnection Layer consists of

- Nodes: Connectivity nodes are represented by smartphones, IoT devices, autonomous machines, with security via a Trust protocol
- Token-Switching Network= Simultaneous conveyance of Value (Token) and In-band message-passing via Token-Switching protocols.
- Proof of HyperMesh Connectivity to ensure nodes are working properly, which enables token reward mining. Consists of two parts:
    - Proof of LPWAN (focusing on LoRaWAN) coverage or Proof of LPWAN
    - Proof of Wifi Mesh coverage (referred to as Proof of Mesh coverage, or Proof of Mesh)
- QoS guarantees, Real time and Delay tolerant protocols
- Security through cryptography, TEE, and blockchain

5.2 HyperMesh Architecture Storage Layer

The Storage Layer has the following characteristics

- Tokenization of Everything: fiat currencies, digital, and real-world objects can be represented/converted into Tokens.
- Messages: Encrypted data, with hash key access
- TransacKet = Data structure which contains Value (Token) and Data.
- Multiple-layer Edge-Storage mesh network allow for tradeoff between performance and security:
    - Database optimized for IoT data
    - Blockchain-secured storage of Messages
- Database optimized for IoT data
    - IoT data stored in the Database, have an associated timestamp and GPS location.
- Blockchain-secured storage of Messages
    - No globally addressed memory (for instance, URL type addressing is NOT used)
    - User owns their Messages by owning the hash key associated with the Messages.
    - Users may store their Messages on their own local smart device, or on the Storage Layer. When doing so, a user transfers the hash key to a proxy at the Storage Layer. The proxy is allowed to give requestors access to the messages if the requestor presents the correct hash key.
    - Storage Layer can perform simple management of messages, such as multi-dimensional, aggregation and dis-aggregation of messages to create new messages.

5.3 Decentralized Execution Layer

Decentralized Execution Layer has the following characteristics

- Execution Layer utilizes the Interconnection Layer and Storage Layer to support applications, such as transactive IoT
- The Programming-Model of Execution Layer utilizes Task Graphs and Job Graphs to support HyperMesh Architecture cyber-physical applications which automate
    - Task mapping to Jobs on physical resources
    - Job triggering based on regular expression conditions of input Execution, and execution of off-chain Smart-Contracts to produce output Execution.
    - Job scheduling in order to meet real-time guarantees.

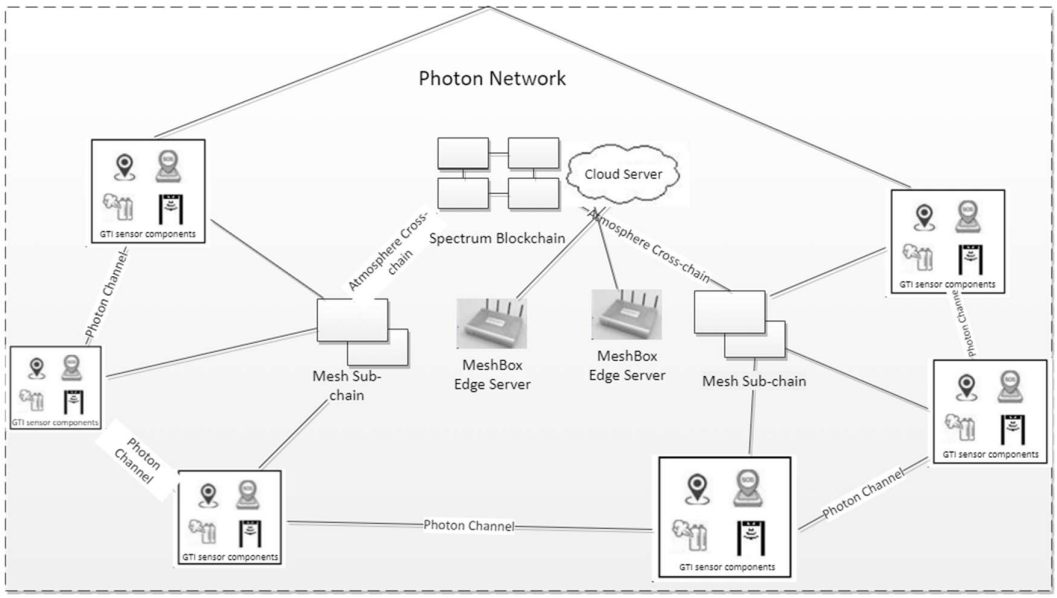
5.4 HyperMesh Technologies

The HyperMesh Architecture includes

- Spectrum public chain: Provides security through simple consensus, targeted for IoT.
- Mesh sub-chain : To improve scalability using sharded Mesh sub-chains.
- Photon layer-2 payment network: Enables transactive IoT

- Atmosphere cross-chain : Linking various blockchains to improve interoperability
- MeshBox Edge-Storage and Edge-Computing: Enable sharing of data and services, with token rewards for MeshBox owners.
- Sensor, actuator, and other IoT devices
- Cloud storage: For some IoT applications which reside in data-centers.

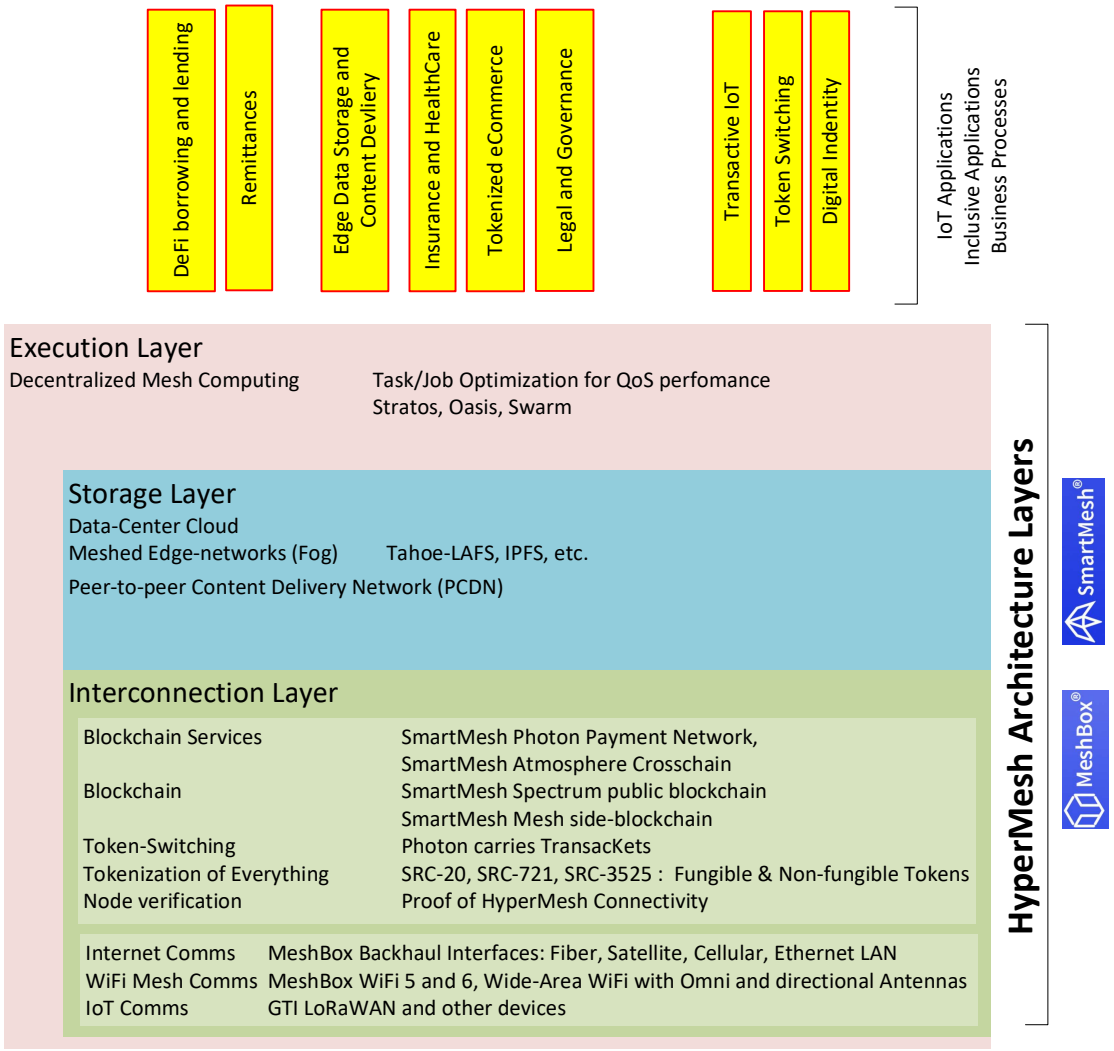
The goal of the HyperMesh Architecture is to standardize interfaces, in order to enable IoT application development; allow more devices to connect to the HyperMesh; and simplify and increase the amount of data which can be shared. Due to the benefits to those using the HyperMesh Architecture, more ecosystem partners will be incentivized to join, thus adding rich functionality and wider deployment of the HyperMesh Architecture, which supports Value Internet applications.



The HyperMesh consists of technology from various SmartMesh ecosystem partners such as MeshBox, etc. In order to provide a flexible and cost-effective solution. Other ecosystem partners are invited to integrate their technology into the general HyperMesh Architecture, which provides various interfaces for such integration. Here is the current state of technology which has been, or is planned to be integrated.

Various IoT Applications may be built on top of HyperMesh, including smart farms, smart supply chains, smart homes. In addition, with the expansion of the HyperMesh Architecture to include IoT-enabled DeFi, additional features such as data marketplace, equipment ownership exchange and cross-chain interactions can be supported, realizing the desired value for the participants.

The various technologies considered for use for the Interconnection, Storage, and Execution Layers are shown in the figure below.



6 HyperMesh Enabled Business Processes

New and exciting IoT business processes and models are supported by HyperMesh as a whole, and also independently for each of the Layers.

Multi-stakeholder and Multi-Revenue-Sharing approaches are now discussed to accelerate the deployment of the HyperMesh Architecture for a local community. Stakeholders include

- IoT and Edge-Networking device manufacturers
- IoT and Edge-Network network operators who deploy and maintain the IoT devices and Edge-Network.
- Investors who pay for the IoT and Edge-Networking equipment
- IoT and Edge-Networking equipment owners (which can be local community members) own the data which their devices generate, and can thus monetize such data.
- Owners of the various locations and business venues where IoT devices are deployed
- Entities who develop and/or run IoT applications.

Each stakeholder can benefit, where appropriate, from the revenue sources described below. HyperMesh and Token-switching supports the billing and payment functions to implement such revenue-sharing, seamlessly, and at a fine level of granularity.

6.1 Interconnection Layer Related Applications and Revenue

- Sharing Internet backhaul bandwidth between community members. A few, high-spped Internet backhaul connections, shared through the WiFi Mesh network, provides economy of scale economic benefits for both stakeholders and users in the community.
- Blockchain Mining: Token staking on Spectrum generate mining token rewards for the staker.
- Photon Payment Network Staking: Token staking on Photon Channels generate fee revenue for the staker.
- Photon Mediated Transfer Fee: MeshBoxes in the HyperMesh network which act as intermediaries earn mediated transfer fees (see diagram below).
- Inclusive Ecommerce: Point-of-Sale (POS) applications, via MeshBoxes, are enabled, in which buyers pay sellers through tokens for goods and services. A purchase from a merchant, through Photon can be immediately rewarded with a token-reward for the buyer, thus incentivizing buyers to shop at the merchant location. Users may also convert their fiat currency for tokens at the POS (see diagram below).
- Tokenization of Everything: Tokenized versions of various cyber-physical entities can be made immutable on a blockchain, with a time-stamp, and queried for their properties. Examples of such cyber-physical entities include personal and legal documents; IoT data and post-processing results, art-work; HyperMesh Edge-Networking equipment; farm produce and livestock, etc. Such Non-Fungible Tokens (NFTs) can be sold and traded, and thus monetized.

6.2 Storage Layer Related Applications and Revenue

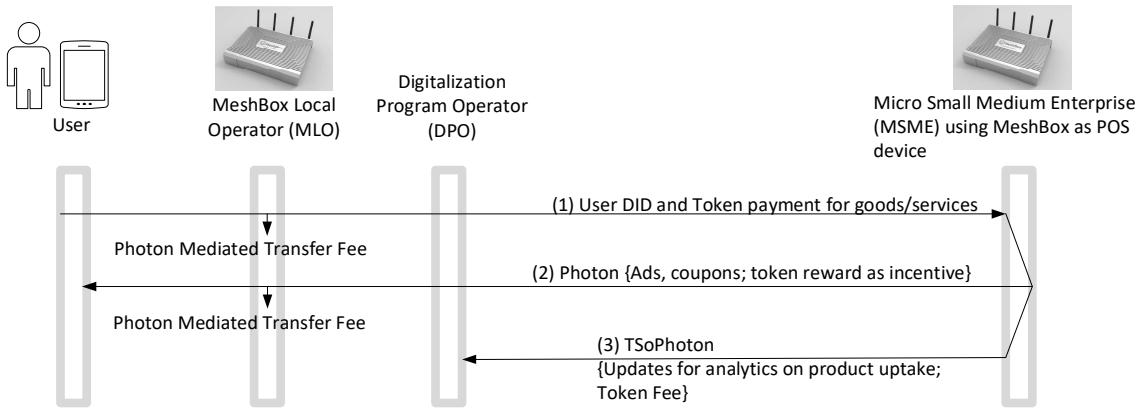
- Token-switching: This web 3.0 paradigm supports user ownership of their own data. This enables users to monetize the sharing or release of their data.
- User Data storage in the Edge-Network (instead of the cloud) can be monetized.
- Peer-to-peer CDN (PCDN) enables popular content to be cached via Storage Layer and shared amongst community members. Such sharing reduces the loading on the Internet backhaul (due to local caching) and improves the availability of such content during intermittent Internet outages.
- Data Analytics: With data stored at the Edge-Network, many types of (potentially AI/ML enabled) analysis, aggregation, compression, forecasting, querying, and triggering on such data can be performed, with the results being monetized.

6.3 Execution Layer Related Applications and Revenue

- IoT services: Observability and controllability of sensors and actuators for IoT applications can be monetized immediately through Token-switching TransacKets.
- Legal and governance: Services can be provided in which physically measured IoT data and conditions act as triggers for Smart-Contracts, which are crypto-graphically enforced legal contracts.
- Inclusive Community Building: E-commerce boosts the local economy, which is supported by the Edge-Networking services built on HyperMesh.
- Transactive IoT: Enables business processes between IoT devices in terms of machine-to-machine and machine-to-people transactions. An agent, using AI/ML tries to optimize benefits, maximize revenue generation, minimize cost, and improve the livelihood of those served within the Edge-Networking area.
- meshDAO: The investors of the IoT and Edge-Networking equipment receive a special DNET NFT token, which bestows to the owner, a seat at the meshDAO (Decentralized Autonomous Organization). Such meshDAO members can vote to determine the deployment plan and trade their NFTs to obtain revenue.
- Liquidity Mining: Providers can stake tokens (including utility and NFT tokens) into the system, and earn rewards, such as mining rewards, and other sources of revenue from the HyperMesh Architecture services.
- Cross-chain transactions: Assets from different IoT sub-chains or other main chains can be exchanged through cross-chain transactions, or through cross-chain communication. Smart contracts on the main chain and other IoT blockchains can be triggered to provide services (such as Value Internet Swap (VISwap)) across multiple Blockchain ecosystems.

6.4 Inclusive Ecommerce Example with Photon and MeshBox POS

The following illustrates how Photon is used in an e-commerce application, in which Merchants operate MeshBoxes as POS devices. As an incentive, the Merchant, may give an immediate token reward, such as a fixed percentage of the amount spent, to customers.



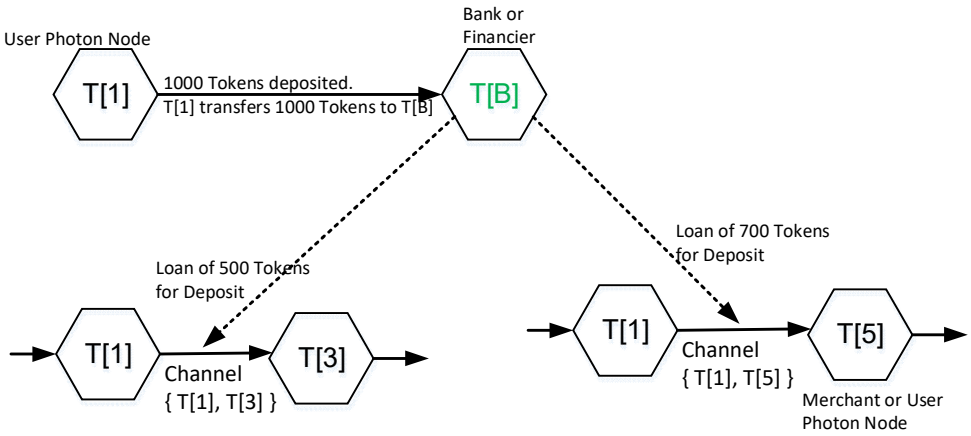
6.5 Photon Network DeFi Support

The following is another DeFi application which can enable Inclusive Finance. One disadvantage of Photon is that each Channel Deposit is locked, which may be difficult for the underserved since locked tokens cannot be used for other purposes. Also, it is difficult to manage many channels (logistically).

To address this issue, users may deposit (lock) tokens into a Single-Account, with a credit multiplier, which can be used as deposits on multiple channels.

Interconnection Layer enables DeFi on Photon, by allowing users to **stake** some amount of Tokens in a single account, as collateral (**investment**) and benefit from

- A **Credit** of a multiple (1.2x, 5x, etc) of the collateral tokens staked, is attributed to the user, to be used for e-commerce transfers
- User can take advantage of **Single-account staking**, with credit being usable across **multiple Photon Channels**.



The following is an example of how the mechanism works, based on the figure above.

- Photon node T[1] is a user with limited funds; T[3] and T[5] are other users or merchant nodes
- T[1] stakes a deposit to T[B], a 3rd party loaning node such as a Bank and transfers 1000 to T[B].
- T[B] takes the 1000 tokens as collateral; gives a 1.2x credit of 1200 tokens to T[1], and deposits 500 and 700 tokens onto Channels which T[1] has with T[3] and T[5] respectively.
- Thereafter, Spectrum transactions, and potentially Photon transfers, related to T[1]'s Photon Channels must have T[B] as a signatory (approval).
- At some point in time, the channel state between T[1] and T[5] becomes:
    - T[1] has available balance = 400 (T[1] owes 400 to T[5]) out of the 700 initially deposited
    - T[5] has available balance = 200 (T[5] owes 200 to T[1])
- If T[1] or T[5] wishes to close the channel
    - T[B] is required to also approve as signatory.
    - T[B] withdraws 700 from the Channel { T[1], T[B] }.
    - T[1] receives 200 plus 100 (unused balance from the channel)
    - T[5] receives 400
- At this point, T[B], has received 700 from T[1], has a remaining deposit of 300 from T[1] on Channel{T[1], but has a deposit of 500 on Channel{T[1],T[3]} for T[1]'s benefit. The credit from T[B] to T[1] is now at 1.66667x (= 500/300).
    - If this amount of credit is allowed by T[B] for T[1], then T[1] can continue to transact with T[3].
    - Otherwise, T[1] must make an additional deposit into Channel{T[1], T[B]} and transfer that same amount to T[B]. Before this occurs, T[B] will not sign any transfers between T[1] and T[3].

6.6 Transactive IoT via Photon Token-Switching

The HyperMesh Architecture, based on Blockchain technology provides a highly secure environment for IoT application in terms of anti-tampering, traceability, and privacy protection of IoT data. Not only can IoT devices exchange data with each other and the IoT application, but it is also possible to support the exchange of value (in the form of tokens), along with the data.

6.6.1 Transactive IoT Token-Switching

The HyperMesh Token-Switching version 1.0 protocol is based on TransacKets being conveyed over the Photon payment network. TransacKets are carried in a Photon transfer, in which the data and a payment associated with the data are simultaneously transferred in the same protocol. This greatly simplies the billing functions associated with data transfers, since the HyperMesh Architecture takes care of such billing.

Token-Switching v1.0 can enable completely new applications including.

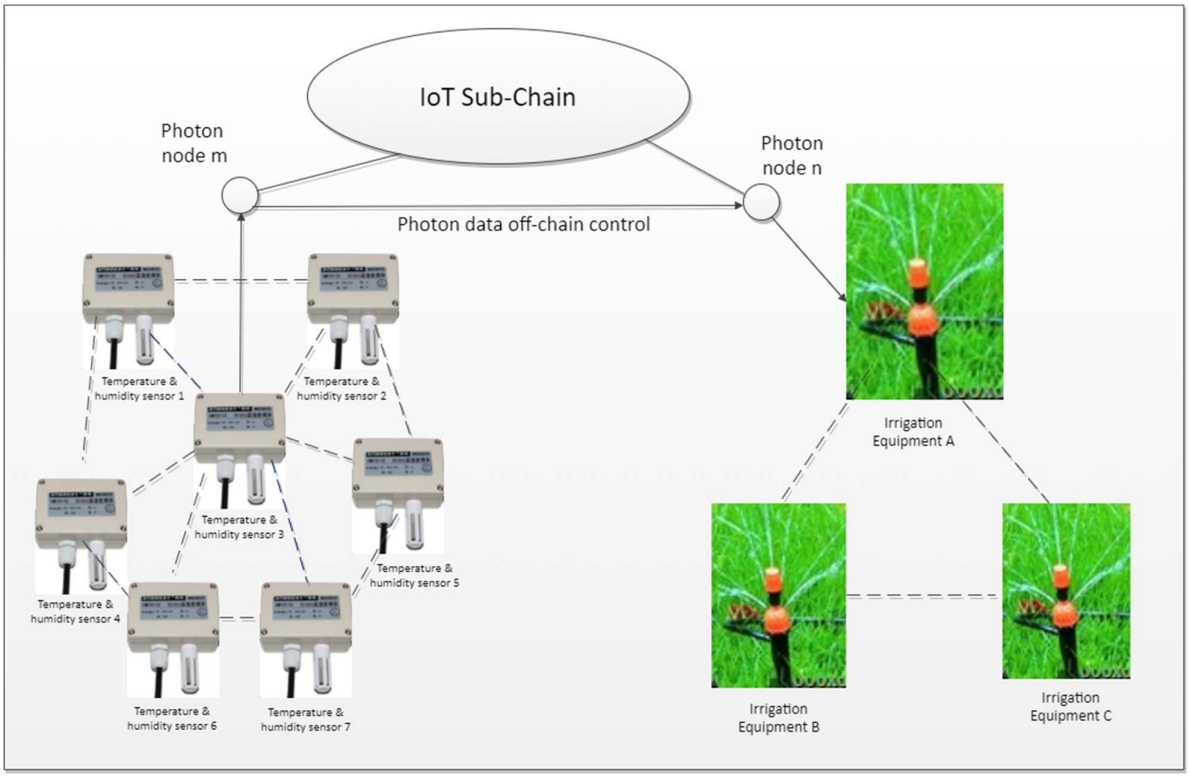
- Transactive-IoT (such as Transactive Energy (energy Internet)) uses blockchain and secure token transactions, for people and machines to participate in business processes.

- Quality of Service (QoS) and Quality of Experience (QoE) applications, in which people and/or machines pay more tokens for better QoS. For instance, if a person or machine is communicating with another entity, and the connection quality is poor (such as due to bad weather impacting wireless communications), the initiator or target of the communication could pay a few more tokens to improve the QoS of their call, at the expense of the QoS for other calls (which did not pay extra).
- Data from sensors, and commands to actuators are securely transferred through Photon, and can further be bundled with a token payment via the TransacKet. When tokens, containing value is transferred with the data/message this greatly mitigates DDOS type of attacks since such attacks will be quite costly. This is in contrast to normal packetized DDOS attacks in the Information Internet in which many packets can be transferred at almost no cost to the sender.
- When an IoT sensor sends data to be stored on the mesh Storage Layer, the TransacKet sent through Photon will contain a small payment for the storage of that Data onto the decentralized Storage Layer.
- When an IoT application wishes to access the IoT data, the application can send a TransactKet with the request for the data (in a message carried by the Photon protocol) as well as a token payment for such data. Thus, revenue will accrue to the MeshBoxes' Photon wallet storing that data and/or to the Photon Wallet of the entity which owns that data.

6.6.2 Transactive IoT Application Use-Case

In a conventional centralized IoT system, IoT devices will send sensor data to an application running in a data-center cloud server. This requires heavy asset costs, long latencies, and reliability issues for remote areas. The application is composed of analysis and diagnosis functions which generate a corresponding set of actions in the control system, to be sent as commands to the IoT devices, such as actuators. The cost of such a system is relatively high, since all of the IoT equipment must be purchased and maintained by the company running the IoT application and the ROI may not be sufficient, which impacts the deployment of the application.

However, with the Photon-enabled Transactive IoT, IoT device purchase and deployment by community members can be incentivized in terms of tokens being paid in exchange for sharing their sensor data with the IoT applications. Thus, the deployment of IoT devices can be done as a grass-roots movement, saving the network operator equipment and operational costs, while benefiting the community residents, such as farmers, through token rewards.



In a traditional centralized mode, sensor data is unidirectionally transmitted to a centralized server in the cloud for processing.

However, in the Edge-Networking Mesh-network scenario, the process of data collection and control can be simplified, using asset light equipment, and execute with low latency and high availability (even when the internet connection is unreliable). Thus, the Token-Switching protocol of Photon greatly simplies the payment of rewards for those who help deploy the HyperMesh.

In the figure:

- The multiple temperature and humidity sensors collect weather and soil data, and communicate with each other and with the low-power WAN (LPWAN) access point (co-located with Photon nodes m and n).
- Initial processing on the IoT data can be done in the Edge-Network (such as in MeshBoxes, co-located with Photon nodes m and n).
- When enough sensor values cross a pre-determined threshold, a photon node (residing at an access-point) can send a TransacKet to a control-process (also potentially residing at an access point, co-located with the corresponding Photon node). The TransacKet may consists of tokens, measured data, and control information, and can be conveyed even if there is no Internet.
- After the photon node connected to the control device receives the transfer, the IoT sensors and the data is authenticated, the appropriate task is scheduled and run, and the command (in the form of an output TransacKet transfer) is conveyed to the appropriate drip irrigation devices to spray the plants with water.
- Meanwhile, the sensor devices continue to gather data. When the measured values reach another pre-defined threshold and remains stable, the events trigger another task at the

control-process, which produces yet another output command in the form of a TransacKet, which is conveyed to the appropriate actuators to reduce the irrigation or shut it off completely.

In this example of an off-chain control process, IoT sensor devices need to perform multiple communication interactions to obtain accurate data. Sensor and control devices interact through Photon via Token-Switching to achieve a lightweight, secure, and efficient local control of Edge-Networked IoT devices.

When the sensors detect a dry condition, a distributed IoT application running at the Access Point (such as MeshBox) will send a TransacKet, carrying tokens, to pay for the water irrigation, which greatly simplifies the billing and payment functions. Further, the owner of the farm can send TransacKets, containing tokens, to pay the owners of the sensors and actuators (as other stakeholders). Such simplicity in bundling data and tokens reduces overhead and boosts efficiency, in order to accelerate the deployment of such a Transactive IoT system.

Since most of the TransacKets conveyed in Photon are processed off-Blockchain (only a few transactions are on the Blockchain), the loading on the IoT sub-chain M_s is minimized.

Further, IoT data can be processed securely, with time-stamping and/or sequence numbers for each TransactKet conveyed. Important events, such as the analyzed result of all of data received in the past hour, can be hashed and recorded immutably on the sub-chain or on the main Spectrum chain itself. Such summary reports can also be sent to the cloud application for further processing and storage.

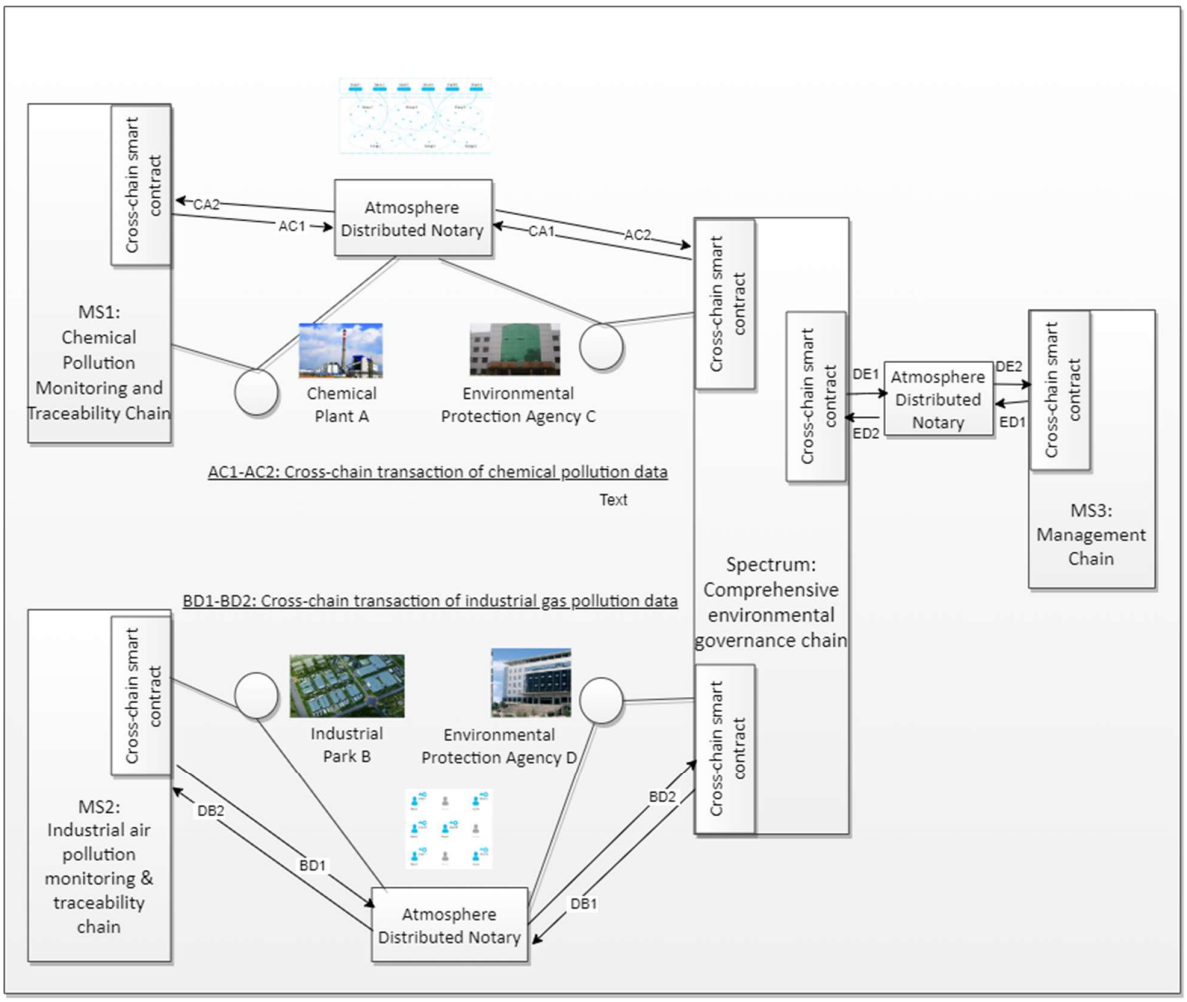
6.7 Cross-chain Business Processes via Sharing of IoT Data

IoT sensor data can enable ecosystem applications on multiple chains. Similarly, cross-chain applications of IoT data will also have multiple application use-cases.

The use of a single blockchain for governance may face governance blind spots. The use of cross-chain technology to connect the relevant chemical pollution-related blockchains and interconnect the governance system with the Spectrum main-chain can bridge such gaps.

The following example illustrates an application of IoT sensors in the comprehensive treatment of chemical pollution. The SmartMesh Atmosphere Crosschain architecture is applied to multi-chain integration to achieve cross-chain integration between multiple pollution monitoring chains (sub-chains) and comprehensive environmental management chains (main chain). This facilitates scientific decision-making support for the overall management of associated pollution.

The multi-chain integrated governance application scenario is constructed as follows.



As shown in the figure, chemical plant A and industrial park B belong to the miner node on the chemical pollution monitoring traceability chain (sub-chain 1) and industrial gas pollution monitoring traceability chain (sub-chain 2) respectively.

Assuming that the two areas are adjacent, but the monitoring of pollutants discharged are in different forms, for example: chemical pollution requires sewage monitoring after biochemical treatment (such as PH value, turbidity, etc.), and industrial gas pollution is related to (O_3, PM2.5, NO_2, NO, PM10, CO, etc.) pollution monitoring data. Since the two are detected by different nodes and uploaded to different sub-blockchains, the correlation caused by the superposition of the two types of pollution in the cross area cannot be reflected. We use Atmosphere cross-chain to link the two pollution traceability chains (sub-blockchains) to the integrated environmental governance chain (main blockchain) for the overall application.

Cross-chain smart contracts are deployed on the chemical pollution monitoring and traceability blockchain (sub-chain 1), industrial gas pollution monitoring and traceability blockchain (sub-chain 2), and the integrated environmental management blockchain (main chain) to observe and control and overall environment. The governance block chain runs on the main blockchain, while the

chemical pollution monitoring traceability block chain and the industrial gas pollution monitoring traceability block chain are sub-chains. Atmosphere cross-chain is used to realize the cross-chain interaction of multi-chain pollution monitoring IoT applications and the comprehensive governance on the main chain. The main participating roles are as follows:

(1) Sub-chain nodes: chemical plant node A, industrial park node B

(2) Main chain nodes: EPA node C, EPA node D

(3) Third party: Atmosphere distributed notary

Among them: the chemical pollution monitoring traceability blockchain is sub-chain 1, and the industrial gas pollution monitoring traceability blockchain is sub-chain 2. The two send cross-chain monitoring data to the main chain by interacting with the Atmosphere distributed notary, and Atmosphere distributed notaries can be deployed by the relevant environmental supervision departments. The geographic location and address of each node are stored in the comprehensive management database of the supervision department.

[AC1 – AC2] The process of sending data on the chain from the sub-chain to the main chain: Taking chemical pollution data to the comprehensive environmental governance blockchain as an example, chemical plant node A invokes a cross-chain smart contract to construct a cross-chain data transaction, and uses Atmosphere distributed notary related interface functions to broadcast to the comprehensive environmental governance blockchain. EPA node C (miner) verifies the signature and the validity of the data, and writes it into the comprehensive environmental governance blockchain after successful confirmation, and a cross-chain data transaction is successfully completed on the chain.

[CA1 – CA2] Main chain feedback governance data on-chain process: Environmental Protection Agency C generates governance data after targeted prevention and controls are assessed according to the pollution situation, and calls cross-chain smart-contracts to build cross-chain data transactions. EPA C then broadcasts to chemical pollution monitoring entity through Atmosphere distributed notary related interface functions. On the traceability blockchain, chemical plant node A verifies the signature and the validity of the data, and writes it into the chemical pollution monitoring traceability blockchain after successful confirmation, and a cross-chain feedback governance data transaction is successfully completed on the chain.

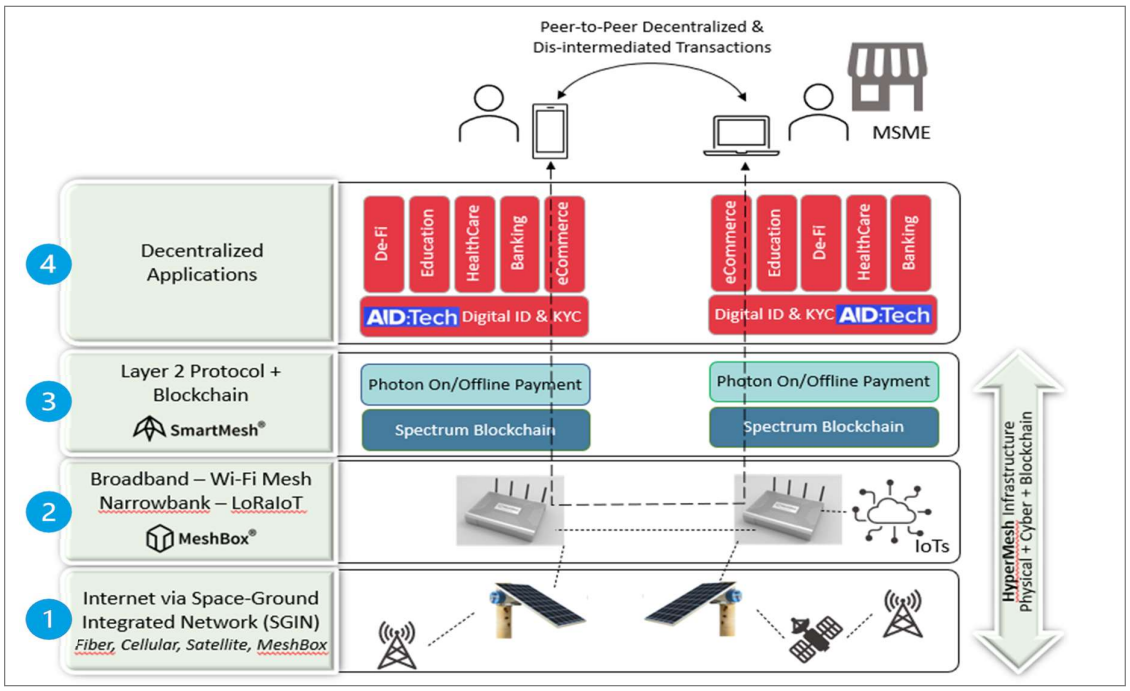
Main-chain logging of events and governance decisions: the main-chain performs correlation fusion analysis on the data from the chemical and industrial gas pollution monitoring and traceability blockchains and makes comprehensive governance decisions, The decision-making and implementation plans are written into the comprehensive management smart contract, providing a query interface for external inquiries and inquiries, and providing a reference for the subsequent evaluation of the effectiveness of comprehensive management of environmental protection.

The above is the two-way flow of data interaction between the sub-chain and the main chain.

[BD1 – BD2 – DE1 – DE2] In addition, if there is a need for interaction between sub-chains, the intermediary role of the Spectrum main-chain can also be used to implement cross-chain smart contract function calls and achieve a wider range of cross-chain functional application scenarios. In this example, a TransacKet from MS2 is conveyed, through the Cross-chain smart contract to

Atmosphere Distributed Notary (BD1), to the Cross-chain smart-contract on Spectrum (BD2) and finally to the Cross-chain smart-contract on MS3 (DE1 – DE2).

6.8 Decentralized Identity for People and IoT Devices



6.8.1 Decentralized Identification for all people

AID:Tech's Decentralized Identification solution creates digital identity for everyone and enables them to prove who they are. An integrated solution of AID:Tech's Decentralized Identity, built on the SmartMesh Spectrum public blockchain is being considered.

Such a solution is transparent and immutable, with the identity being owned by the rightful owner, the user themselves. There will no longer be a trust issue, with the identity being accessible anywhere in the world, 24/7/365. Digital identify is the foundation to bring financial inclusion to the Unbanked, as well as social inclusion such as welfare, aid and donations for the needy. Together with AID:Tech's other applications like KYC, healthcare, and aid, the solution provides a unified identification service for users to access other DApps seamlessly on the HyperMesh.

7 IoT Security Who, What, When, Where, and How

The Who, What, When, Where, and How the Hypermesh Architecture achieves IoT security is discussed in this paper.

A comprehensive security architecture is a critical aspect of the HyperMesh architecture. All HyperMesh layers (Interconnect, Storage, and Execution) must be secure individually, and also synergistically.

7.1 WHO to Secure

People, machines, and objects (including IoT devices and eventually sentient robots) require an immutable identification, which is controlled by themselves.

Decentralized Identities (DIDs) fulfill such a requirement, and are needed in order for them to participate in finance, health, insurance, and welfare transactions. IoT devices generate valuable and at-times, mission critical data, and must be authenticated.

The HyperMesh Architecture may use the standardized decentralized identity method from the W3C Credentials Community Group to give a decentralized identifier (DID) to each device. An example format of a DID is:

did:methodname:123456789abcdefghi

The DIDs of IoT devices can be registered and accessed anytime on the Spectrum blockchain. For scalability, the DIDs and Verifiable Credentials may be stored on the Mesh Sub-chain covering the group of local Edge-Network of IoT devices.

In order to prevent malicious registration of DIDs for many bogus IoT devices, constituting a denial of service (DOS) attack, the philosophy of Token-switching is applied, in which a token payment is needed in order to register each new DID.

7.2 WHAT to Secure

The data which are generated by people and IoT machines must be secure against tampering.

In supply chains, goods and services must be secured against theft or damage (such as food getting spoiled).

For instance, in smart supply-chain and smart home IoT applications, it is helpful collect video recordings of business processes, such as for supply-chains. Such videos can be viewed in real-time, processed for anomalies, and stored, or flagged. According to the required privacy levels the following are needed.

- Two-channel Video data collection and upload (save to cloud services and mesh sub-chain registration) are to be simultaneously supported. Via one channel, the video can be directly sent to the cloud for storage. In the second channel, the video can be hashed, with the encrypted version being stored on Storage Layer and the hash being recorded on the Spectrum blockchain.

- Privacy-encrypted two-channel video collection and upload (using TEE encryption and storage to cloud services). This option supports the encryption and sharing of video clips, and supports authentication via the Mesh sub-chain.

7.3 WHEN Security Occurs

The time associated with key events is of critical importance. An event may be captured by a Blockchain transaction, which are kept in an immutable, timestamped ledger for all participants to query, and cannot be tampered with.

7.4 WHERE Security Occurs

Being able to securely determine the location of people and goods is highly important for several user and IoT applications. This can be done with IoT devices which can be tracked with GPS, and also through cryptographic consensus mechanisms, such as Proof of HyperMesh Connectivity.

IoT devices which claim to be at a certain location, and performing certain tasks should be checked and verified through Proof of HyperMesh Connectivity.

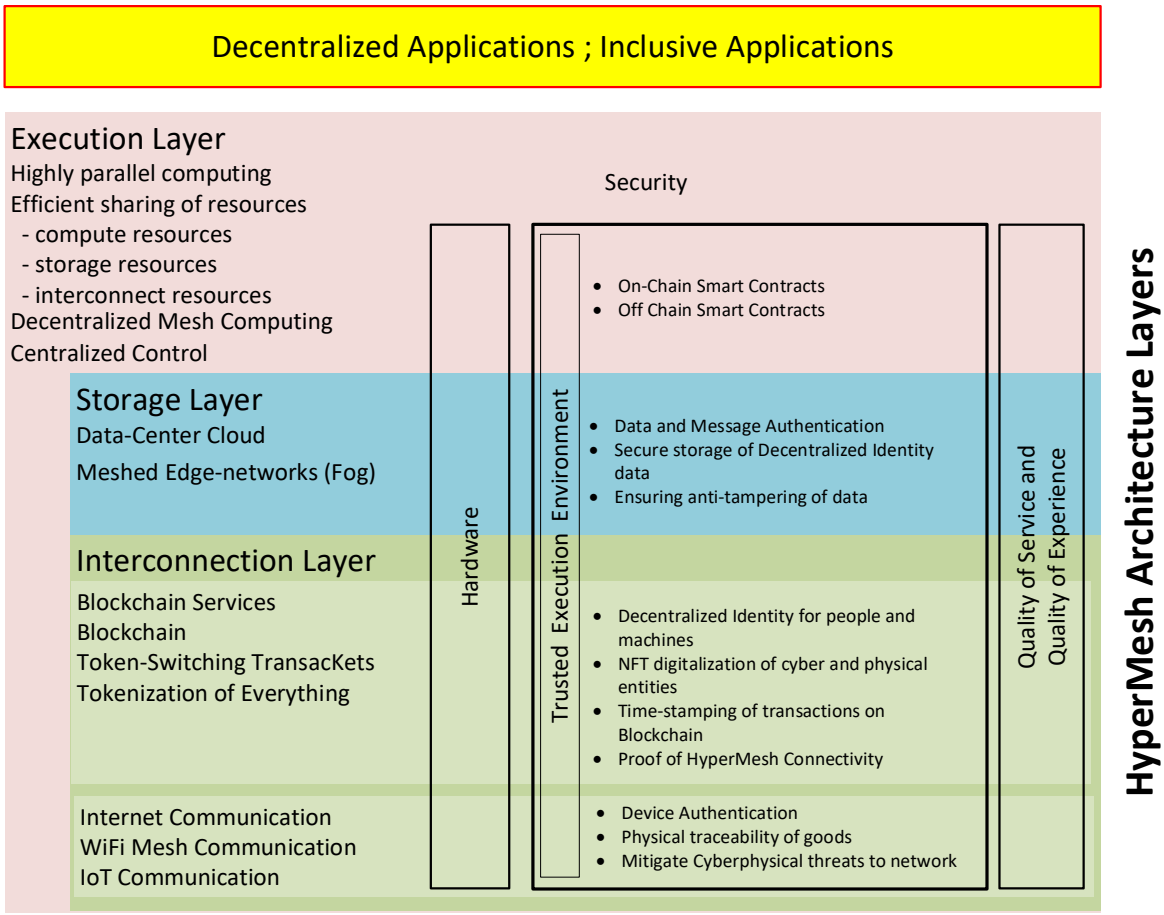
The Cyber-physical aspect of security is also considered, in which secure transactions must take place, even with unstable internet connection. The HyperMesh is designed such that local services, via the MeshBox WiFi mesh network, are still available, even when the connection to the Internet is lost.

Within the mesh network, one can imagine cyber-physical attacks, such as theft or destruction of MeshBox nodes, or introducing impediments in the air or physical surroundings which greatly attenuate the wireless signals.

The MeshBox++ nodes are designed with high Ingress Protection 67 and are also robust to withstand physical attacks, due to the tough casing and antennas being protected inside the node. The mesh network also reconfigure automatically when any nodes are taken offline, in order to maximize the throughput, even in the presence of node failures (such as due to attacks). Also, the MeshBox++ nodes are self sufficient in power with solar and batteries, and will continue to operate through power outages.

7.5 HOW Security is Guaranteed

In the HyperMesh Architecture, Security spans all of the Layers.



7.5.1 Interconnect Layer

The Interconnect Layer, which includes the communication network and the blockchain, must provide security services such as IoT Device Authentication (Who), physically tracking the location of goods (Where), and mitigating Cyberphysical threats (Where).

Token-Switching version 1.0, implemented in Photon, is a new Value-Internet protocol, which adds a new dimension to the information internet by integrating value transfers in the same protocol as the data transfer. This allows billing and payments to be transacted between peers (tokens paid in return for data transfers), without traditional overhead (accounting and billing), inefficiencies, potentially errors and possible corruption.

Security is enhanced in that DDOS and Spam attacks can be mitigated, due to the (optional) cost associated with the sending of data.

7.5.2 Storage Layer

The vast amounts of IoT data may overwhelm blockchain TPS and storage limitations, necessitating wise tradeoffs for storing data on-blockchain, versus off-blockchain. Highly important

data, such as legal documents, decentralized identifiers, and IoT authentication information may be stored on the blockchain in order to guarantee immutability. Data, which is transitory, such as streaming data, or large amounts of data are better suited to be stored on decentralized Storage Layer, with perhaps a hash being stored on the blockchain in order to provide a time-stamp of a transaction, and guard against tampering of such data.

7.5.3 Execution Layer

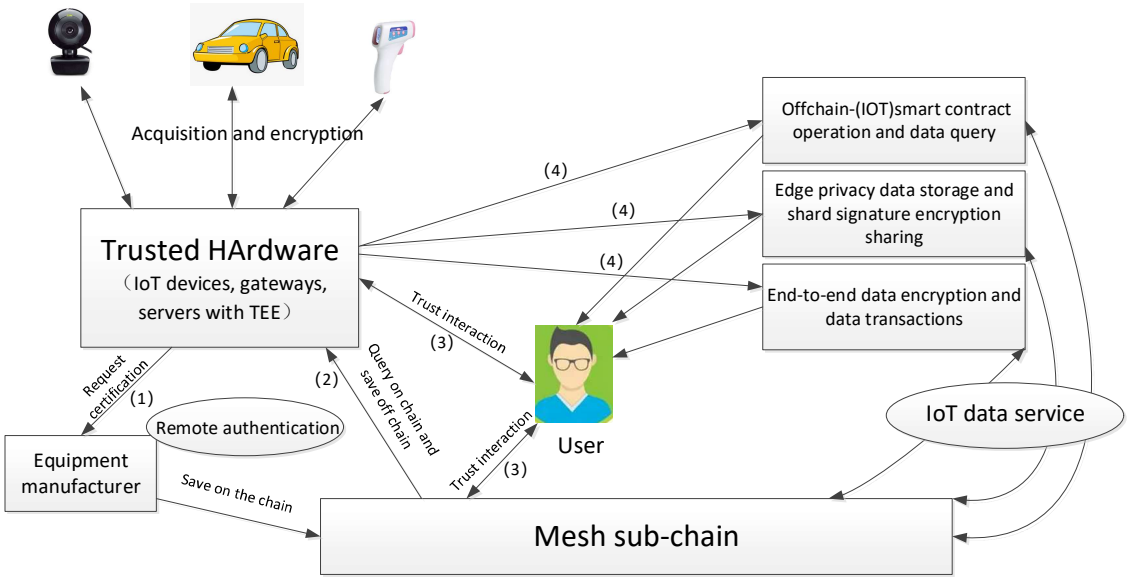
Token-switching, used in on-chain and off-chain smart-contracts, enables applications in which users control how their data is shared, and the monetization of such sharing. Token-switching crypto-graphically simplifies the management of synchronization between information transfer and a token payment, associated with the information transfer.

7.6 Trusted Execution Environment (TEE)

IoT data is generated by various sensors and devices. To ensure the credibility of such devices, and the authenticity and privacy of data which are generated, TEE is considered as a technology solution. TEE may be more efficient as a secured solution for massive data processing requirements compared to zero-knowledge proofs and other multi-party computing security measures. TEE is implemented as a combination of software and hardware, usually as part of a CPU, or a separate Micro-controller Unit (MCU), and is a secure hardware platform on which some code (such as critical kernels) can be executed. TEE hardware, strictly isolated from other areas in the CPU, is secured hardware logic where programs execute.

At the same time, because of the strong isolation, it is very difficult for outsiders to steal the program or data inside TEE, Thus, processes executed in TEE cannot be tampered with.

This is very similar to the concept of a smart contract. In the IoT architecture within the SmartMesh ecosystem, TEE has a wide range of use cases. TEE can be used in the authentication of IOT devices remotely, providing tamper-proofing for data storage, operating off-chain smart contracts and query services, providing edge private data storage, and [threshold signature ???] fragment signature encryption sharing, etc. TEE also supports end-to-end data encryption and data transaction service through Elliptic-curve Diffie-Hellman (ECDH) anonymous key agreement scheme. Thus, TEE is able to provide security guarantees for users and Mesh sub-chains via trusted interactions, as well as for data authentication, equipment confirmation, and trusted interactions for edge devices.



TEE Interactive Scene Description

The figure above illustrates the interaction among TEE, users and blockchain. The process for establishing trust and providing services is as follows:

(1) TEE generates a remote authentication request and sends it to the equipment manufacturer. After equipment manufacturer verified its identity, certificate is issued and saved in the mesh sub-chain via the bridge program (at the same time, TEE equipment public key is written to the chain)

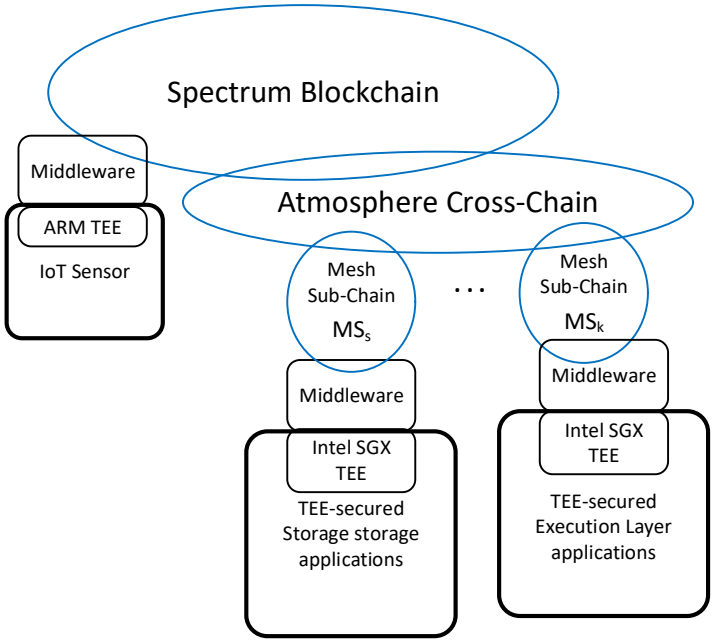
(2) TEE obtains relevant data from the chain, verifies and saves it (input verification will be carried out)

(3) User verifies the authenticity of TEE through certificate on the Mesh chain (TEE can also use a signature to verify user's identity), and after bi-directional verification is successful, it can interact with the TEE securely and trustfully (using ECDH key negotiation to establish end-to-end secured communication)

(4) For IoT application, different TEEs will execute anti-tampering data solutions (or off-chain smart contracts), and upload signatures of the collected sensor data (encrypted or unencrypted, depending on requirements) to cloud server or edge server for metadata on-chain processing and data authenticity verification, and provides further confirmation and interconnection services for upper-level applications (peer-machine interconnection, machine-machine interconnection, business-business interconnection, and physical object and program interconnection).

Other examples are shown below.

(1) The RA4M2 chip by Renesas integrates a 100 MHz Arm Cortex-M33, with TrustZone (analogous to TEE). The chip includes a secure crypto engine, offering secure element functionality [RA4M2].
(2) Some decentralized applications use TEE functionality, implemented as Software Guard Extensions (SGX) in implementing secured decentralized data storage.
(3) For the Execution Layer, TEE can be used to execute some critical portions of application code for enhanced security.



8 Interconnection Layer Architecture

The following details the architecture of the Interconnection Layer, which includes the following hardware for communication

- IoT devices, such as LoRaWAN devices from GTI
- IoT Access Points
- Edge-Computing nodes (such as MeshBox)
- Modems and Antennas for connection to Internet backhaul

Related to the W3C DID method, each IoT device should be equipped with a unique public/private key pair, which can be used for identity verification. For security, IoT devices also integrate anti-tampering technology and a Trusted Execution Environment to mitigate against malicious attacks.

The Interconnection Layer also includes

- IoT Device and Data Security
- Spectrum blockchain, Mesh sub-chain, and Atmosphere cross-chain
- Smart contracts and services

8.1 IoT Device and Data Security

8.1.1 Device Connection Model

For each application scenario, s, a group of IoT devices D_s, is allocated to support the application. For each specific IoT application scenario, a Mesh sub-chain MS_s is allocated to support that scenario, with the corresponding D_s IoT devices being served by sub-chain MS_s.

The sub-chain is jointly maintained by multiple blockchain nodes $n_{s,b}$ which run the consensus algorithm . Such nodes can run in the nearby MeshBoxes, working as edge servers. In order to facilitate communication operations, the Mesh sub-chain nodes for MS_s are usually deployed in the geographic vicinity of IoT devices D_s, which they serve. Meanwhile, the blockchain nodes belonging the Spectrum main-chain are located anywhere, world-wide, such as in cloud servers, MeshBoxes, and other common computers. While the various sub-chains and the main-chain are connected via the Atmosphere cross-chain, the sub-chains are not directly connected to each other.

8.1.2 Certification Management Model

Each of the devices of D_s, denoted $d_{s,j}$, has a certificate $Ct_{s,j}$, which is generated by the Certificate Authority (CA) and registered on the sub-chain MS_s. When device $d_{s,j}$ uses its private key to sign a message (containing IoT data), blockchain nodes in MS_s can use the corresponding public key to prove validity, verify the message, and write a transaction that contains the certificate of $d_{s,j}$ into a block to be added to the blockchain. Considering the large number of IoT devices, MS_s does not share each certificate in its registry with other sub and main chains.

Each blockchain node $n_{s,j}$ in MS_s also has its own certificate to support decentralized ledger operations, where digital signature and stored certificate are used to verify each other. Since the number of blockchain nodes relatively low (compared to the number of IoT devices), blockchain node certificates are stored both in the sub-chain M_s and Spectrum main-chain.

Certificate revocation can be completed within the corresponding sub-chain ledger. The revocation of an IoT device certificate is managed by the sub-chain nodes and broadcasted to all nodes in the system. Cross-sub-chain authentication, if required, can be achieved through the Atmosphere cross-chain protocol.

The HyperMesh Architecture may use the standardized decentralized identity method from the W3C Credentials Community Group to give a decentralized identifier (DID) to each device.

8.1.2.1 IoT Device Certificate

Device certificate is a prerequisite for data authentication. IoT devices can be certified using private key or digital signature in the following ways:

1) Signature authentication – The IoT device $d_{s,j}$ submits a digital signature to the connected sub-chain ledger MS_s. Nodes in MS_s can then use the certificate stored in the ledger to verify the signature.
2) Crosschain Certificate check – In case the IoT device $d_{s,j}$ migrates to a new ledger MS_s', blockchain nodes in MS_s' can obtain that device's certificate from ledger MS_s via Spectrum main-chain through the Atmosphere cross-chain protocol.

8.1.2.2 IoT Data Authentication

In order to ensure the authenticity of IoT data transmission, taking into account the limitations of the energy capacity of IoT devices, a Schema, used by [IoTeX] is considered to be used for the authenticity of IoT data.

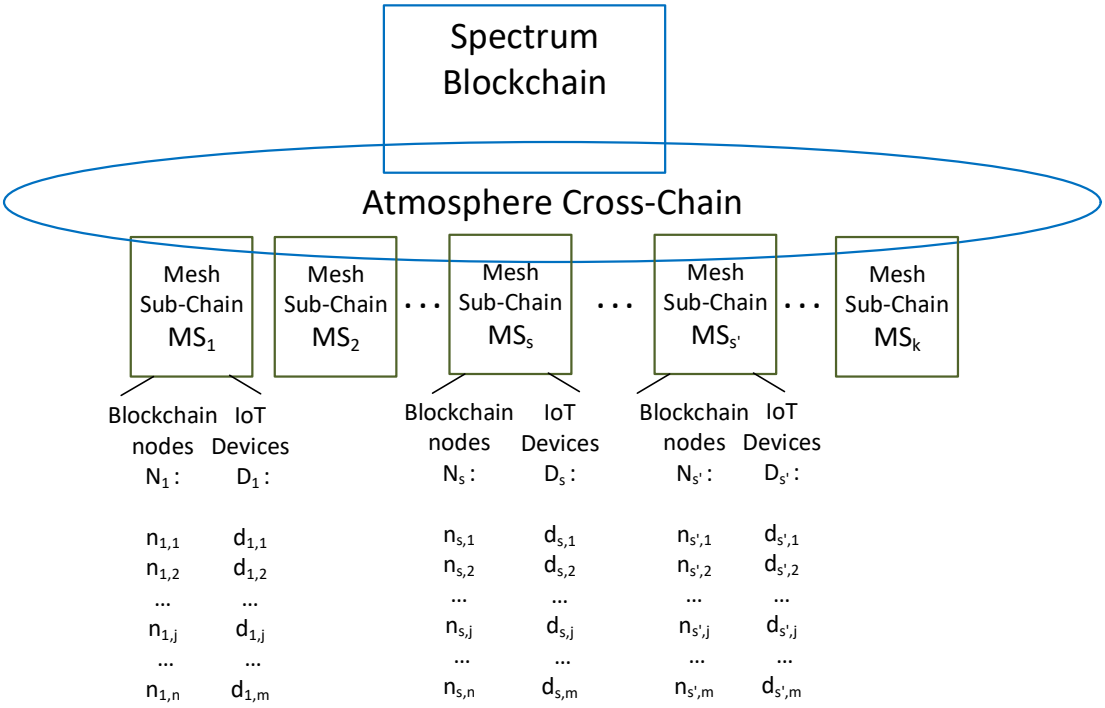
8.2 Spectrum Blockchain and Mesh Sub-chain

Blockchain and the associated services are used to interconnect nodes via transactions. The Spectrum blockchain and Mesh sub-chain work together to maintain a secure ledger for IoT, which is used to provide authenticity confirmation of IoT devices, data and related services.

8.2.1 Blockchain, Sub-Chains, and Cross-Chain Architecture

From the beginning, SmartMesh® blockchain technology, consisting of the Spectrum blockchain, and Photon Payment Network (Layer-2, Smart-Contract) has been carefully designed in order to meet HyperMesh and IoT requirements. A two-Layer ledger structure is used, consisting of the Spectrum Main-blockchain (main-chain) and multiple Sub-blockchains (sub-chains).

Due to the large number of IoT devices, such devices are partitioned based on geographic location, or another logical sharding method, to be served by their assigned sub-chains as needed. Each sub-chain is connected to the Spectrum main-chain through the Atmosphere cross-chain.



Smart contracts and services on Spectrum include Photon payment network, Atmosphere cross-chain service, and Mesh-related services to streamline interaction between IoT devices (such as MQTT communication services).

8.2.2 Photon Payment Network and Services

Due to Blockchain's low TPS, and the high volume of IoT transactions, Blockchain TPS scalability is usually addressed with Layer 2 scalability solutions. Photon is such a Layer 2 TPS scaling solution supporting a payment network, which is secured by the Spectrum Blockchain, but conducts most transfers Off-blockchain. Photon thus offloads transactions off of the blockchain and enhances the interactive functions between IoT devices due to high TPS scalability and low-latency transfers.

For areas with Intermittent Internet, Photon is designed to support transfers even when unconnected (to the Internet, and therefore disconnected from Spectrum). Immediate Token-Economics incentives through Photon accelerate adoption.

However, Photon suffers from the requirement to lock deposits on Channels, which can be challenging for inclusive applications. This will be addressed with HyperMesh Architecture support for DeFi applications, including lending/borrowing, investment with interest payments, NFT tokenization of documents and assets are discussed.

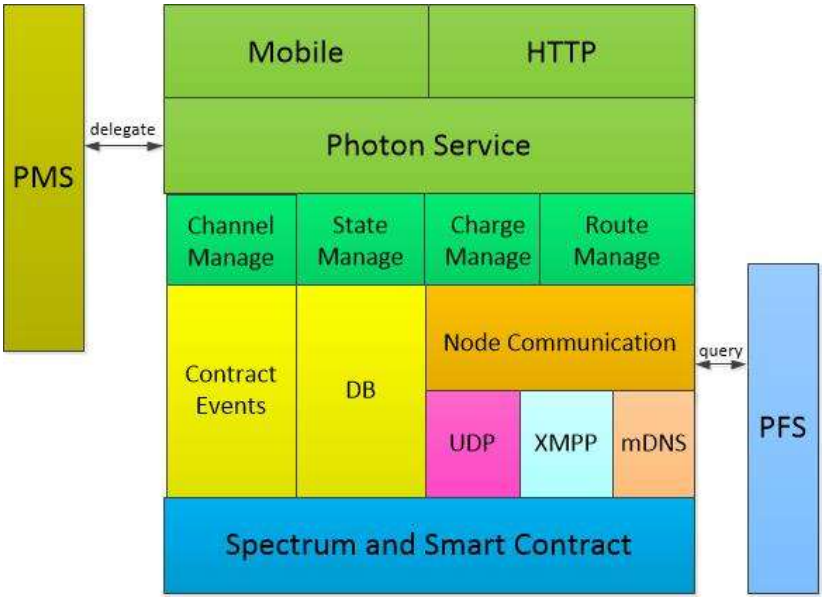
For IoT related processes, such as supply-chains, which need to meet QoS guarantees, Spectrum smart contracts can be converted to Photon Job-Graphs for enhanced performance and lower fees. This is due to Photon support for highly scalable TPS and online/offline payments.

Photon network currently supports Token-Switching version 1.0, in which TransacKets supports the simultaneous transfer of both In-band data, with tokens which carry value. TransacKets can be used by applications to trigger smart-contracts for Execution Layer. Atmophere cross-chain supports interaction between applications running on other industry blockchains (Bitcoin, Ethereum), Mesh sub-chains, and the Spectrum main blockchain.

For details on the Photon architecture, please refer to [Photon] and the Appendix.

8.2.3 Photon Architecture

Photon uses the following protocol stack to support off-chain token transfers.



There are five layers in the protocol stack, starting from the bottom to top:

- Smart contract layer: Based on the Spectrum public chain, which secures transfers and TransacKets being transferred over the Photon payment network.
- Communication layer: Including smart-contract events, database, and node communication components. Communication is based on UDP and XMPP. XMPP supports interactive query routing with Path-Finding-Service (PFS) and charging. For offline node discovery, mDNS is specially designed to interwork with UDP for offline communication.
- Management layer: Includes channel management, device status management, charging management and routing management to handle channel functions, status changes, rate setting and revenue, routing query, etc.
- Service layer: Responsible for processing and distributing all user requests to the lower layers, and supporting interactions with MSs for security.
- Application layer: Including mobile smartphone and HTTP applications.

8.2.4 Photon Unique Features

Photon is unique terms of fault-tolerance and security features, including the following.

- Off-chain direct and mediated payments (data transfer)
- HTLC mechanism

- Offline (without internet) direct payment (data transfer)
- Crash recovery security
- Supported on multiple Operating Systems, including windows, linux, android, IOS, etc.
- Cross-chain support
- Third-party monitoring services and routing support, etc.

Based on the above framework and functions, Photon can realize trustless, real-time and off-chain peer-to-peer payments and data transfers.

8.3 Atmosphere Cross-chain Architecture

Existing centralized IoT solutions focus on data processing and analysis at the cloud data-center, so costs and bottlenecks are also concentrated at the cloud processing, and which presents scalability issues for IoT applications. While the IoT ecosystem HyperMesh Architecture mitigates the IoT device cost (through rewards for those who deploy sensors and MeshBoxes), the cost and processing bottleneck of the centralized processing paradigm is now addressed.

The HyperMesh Architecture, leveraging the Atmosphere cross-chain design is able to offload the bottleneck at the centralized cloud server. In the HyperMesh approach, initial data processing requirements for a local area are handled by the edge-servers. Thereafter, further processing (data analytics) involving multiple IoT areas is mapped to cloud servers and the main chain.

IoT sub-chains M_s and Spectrum main-chain support IoT-related TransacKets (data and token transfer) through the Atmosphere cross-chain. NFTs such as DNET, which are supported over Spectrum and Atmosphere, can also carry both value, and data. The Spectrum main chain is used as a bridge to design cross-sub-chain function calls (supporting a wider range of IoT data applications).

8.3.1 Atmosphere Features

Atmosphere is designed to realize a cross-chain interconnection of value. The design idea is realized by the main-sub-chain and two-way anchoring method, with Spectrum as the main chain, and other IoT sharded (such as locality-based) sub-chains M_s. The sub-chain data is mapped to Spectrum through two-way anchoring. With the help of Spectrum's intermediary role, functions such as cross-chain data exchange, cross-chain payment, and cross-chain data access payment across multiple sub-chains can be further realized.

The Atmosphere cross-chain solution uses distributed keys, HTLC atomic swaps, threshold signatures, homomorphic encryption, zero-knowledge proof, PBFT and other technologies to build a safe and stable cross-chain design.

8.3.2 Atmosphere Cross-Chain Process

Refer to the whitepaper [Atmosphere] for details of the Atmosphere Lock-in (transferring from a sub-chain into Spectrum) and Lock-out (transferring from Spectrum to a sub-chain) processes.

8.4 Potential Interconnection Ecosystem Collaborations

GTI IoT Technologies collaborates with MeshBox and SmartMesh on IoT, by providing a variety of IoT devices, including sensors and actuators.

8.4.1 GTI IoT Technologies Introduction

An example of IoT devices which can be used for the Interconnection Layer are those using the universal sensor development board of the STM32F103RT6 chip developed by GTI IoT Technologies. The component has a variety of sensor interfaces, used to collect data parameters of up to 15 sensors, and further uses the HyperMesh Architecture to store and analyze the sensor data.

GTI offers a variety of sensors for plug and play, enabling sensors and transmitters to be quickly and seamlessly connected to the IoT application system. In order to adapt to the demonstration application scenarios of the SmartMesh ecological IoT, the following conceptual models of IoT devices with blockchain attributes are planned.

8.4.2 GTI environmental monitoring sensor devices

According to the needs of smart agriculture IoT application scenarios, the environmental conditions of crops (soil temperature and humidity, air temperature and humidity, illuminance, etc.) need to be regularly monitored, saved with timestamps, and processed for potential control equipment actions.

For the convenience of monitoring and unified management, GTI has developed a sensor unit which measures: soil temperature, soil humidity, air temperature, light intensity. Such data can be aggregated and saved on a MeshBox Edge-Storage server. The HASH value of the aggregated data is computed and sent as a transaction, to be recorded on a Mesh sub-chain. The Hash value can be used to store, authenticate, and access data via a query. Data values passing pre-determined thresholds can also be used by Execution Layer to construct early warning transactions on the chain for query and control actions.

8.4.3 GTI sensor tracking component model

According to the needs of the smart home or supply chain, sensors are needed for quality control and traceability. GTI sensors measure temperature, humidity, pressure, and location (latitude and longitude), which can be leveraged by mesh networking to support Photon network, off-chain data sharing and actuator control.

8.4.4 GTI smart home sensor component model

In the smart home IoT ecosystem, there are two types of sensor component models for the environment and the human body.

- Environmental sensing components include: temperature, humidity, light, smoke, odors, infrared, etc.
- Human body sensing components include: sound, blood pressure, blood sugar, body temperature, pulse, etc.

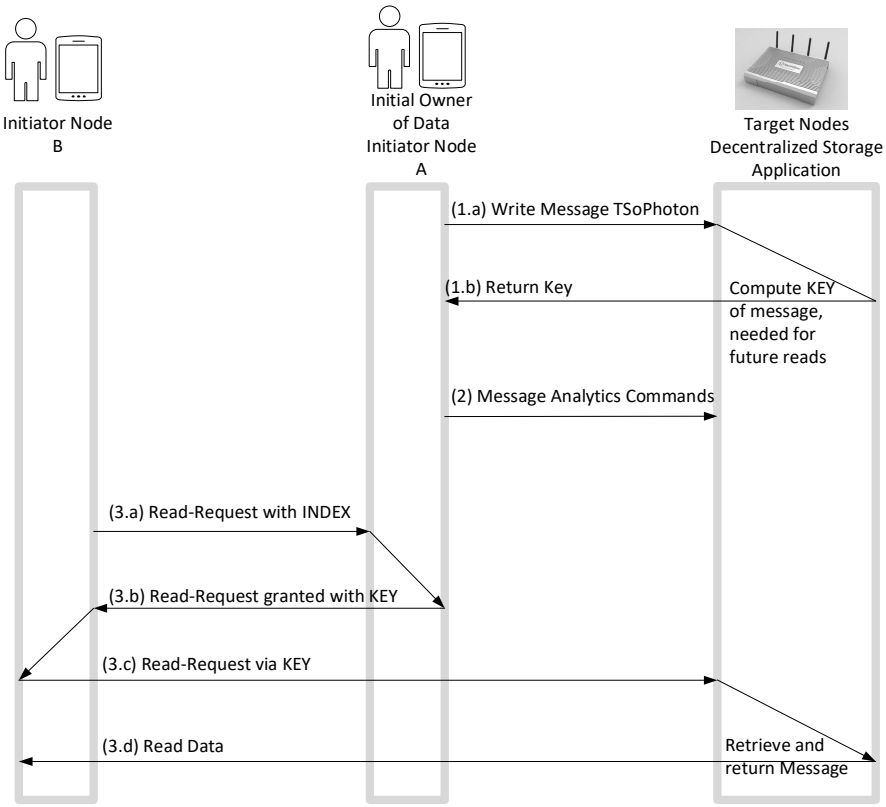
The two types of sensor components form a sensor data network, and may regularly upload data to edge-servers and/or cloud servers to enable subsequent intelligent analysis and actions.

9 Storage Layer Architecture

Storage Layer provides storage services, which can be implemented both in the cloud, and within the Edge-Network, which can consist of a WiFi mesh network composed of MeshBoxes. Storage Layer is used to store data generated by connected IoT devices, to be accessed by Execution Layer related applications. The following are included in Storage Layer.

- Data-Center Cloud
- MeshBox decentralized storage and Edge-Computing
- Peer-to-peer Content Delivery Networks (PCDN)

Data belonging to users can be monetized via Token-Switching. The following illustrates how Token-Switching works in terms of the programming model to access data maintained by Storage Layer. Initiator Node A creates the original message and stores the message on Target nodes (Tnodes), which can be implemented on a MeshBox mesh network. Tnodes run Decentralized Storage Application, which is built on Storage Layer.



Token-switching version 1 consists of transfers via Photon and has the following functionality.

(1) Write Message:

- (1.a) Using the Token-switching-over-Photon (TSoPhoton), Initiator-node-A (Inode-A) pays tokens to Target-node(s) (Tnode(s)) to store Message.

- (1.b) Tnode computes and returns a secret KEY (hash), and a public INDEX needed for future Reads (retrieval) of the message.

(2) Message Analytics commands: Commands sent in Message (along with KEY, and optional small token value) to Tnodes in order to perform computation/re-factoring/analytics of Messages stored at Tnodes.

(3) Read Message:

- (3.a) Inode-B, is running a Task, defined in Execution Layer, which has the INDEX as an input event, broadcasts a Read Request with the INDEX.
- (3.b) Inode-A (which has exclusive access (cache coherency) to the Message) recognizes this request and sends KEY to Inode-B in a Message,
- (3.c) Inode-B sends Read command to Tnode(s) with KEY in the Message and the appropriate amount of tokens (non-zero amount helps to prevent DDOS attacks).
- (3.d) Tnode (running decentralized storage protocol), sends Message corresponding to KEY to Inode-B.

TS version 2 is defined as General token switching, whose architecture is to-be-determined.

9.1 IoT Data Storage Needs

There are no one-size-fits-all solutions for IoT data storage, since requirements can be different depending on the use cases, deployment locations and data type (static vs streaming). The requirements are assessed by first breaking down the IoT data process as follows:

IoT Data Processes	Tasks	Deployment
Data Ingestion	collect and store real-time data and messages	Edge
Edge Analytics	▪ Scribe, translate, classify and aggregate all incoming data ▪ provide real-time data for decision making	Edge
Device Management	▪ store device status data ▪ send/receive messages to/from devices.	Edge
System-wide Analytics	▪ Scribe, translate, classify and aggregate all data from edge data storage ▪ provide data for decision making ▪ provide data for AI & Big data analytics	Centralized
Big Data and AI	▪ Normalize data ▪ data presentation and visualization	Centralized

The following considers the data storage needs for edge deployment. The key factors for selecting the right edge data storage for IoT solution are:

1. **High Write rate** – IoT can generate millions of messages at high rate. While some IoT devices create data periodically, others may have unpredictable bursts of data. The database needs to be able to support high write rate and highly bursty data receive (ingress) rates.

2. **Real-time/Edge Analytics** – the ability to perform real-time analytics on high-volume, time-sensitive data is often critical in the IoT solution for providing real-time decision-making to sound alerts for emergency management or controlling sensors and devices. Hence the database needs to provide timely access to the data.
3. **Synchronization for Fault Tolerance** – Connectivity in the environment where IoT is operating in may be intermittent or slow and hence can create challenges for data transfer. Database therefore needs to be fault-tolerant and perform read/write queries in such conditions. Distributed databases with synchronization and publish-and-subscribe features are much desired.
4. **ACID** – (atomicity, consistency, isolation, durability) compliant database ensure that a database transaction is completed in a timely manner such that no data (or a very small amount of data) is lost and assuring data integrity.
5. **Flexibility** – Different IoT devices produce data in different formats and may include structured and unstructured data like sensor readings, digital and analogue signals, metadata, images, sound and video. A flexible IoT data storage solution needs to be able to store any type of incoming data without the need to redefine data schemas.
6. **Data Security** – IoT data are usually highly sensitive and privately owned. Data storage solutions should therefore apply security techniques, protocols and encryption at all architecture levels, during transmission and storage in order to maintain integrity and privacy of the data.
7. **Scalability** – With the ever-increasing proliferation of IoT devices, it is important to have a sustainable scaling strategy for storage. In addition to scaling vertically (increasing memory size, disk space or server CPU), horizontal scaling (sharding, and adding new servers) is also helpful.
8. **Small Footprint** – Since IoT databases are mostly deployed at Edge devices, where resources like space, CPU and memory are limited, databases should have a small footprint.
9. **Low Maintenance** – IoT solutions may be deployed across geographical regions. Data storage and databases should operate autonomously and unattended without the need for a database administrator (DBA).

9.2 Database Storage Technology Alternatives

Database storage can be categorized, and their respective Pros and Cons with respect to IoT application are as follow:

9.2.1 Non-SQL Databases

Non-SQL databases include (e.g., MongoDB, Couchbase, Apache Cassandra, Apache CouchDB etc).

NoSQL databases are schema-less or non-relational, allowing new data types or formats to be added dynamically, and are therefore the most flexible among the choices of database type. NoSQL databases are originally designed to run on cluster of servers and have built in capability to replicate data across nodes. Therefore they scale more cost efficiently and have superb fault

tolerance compared to other database types. Schema-less database stores data as JSON documents and are typically write-optimized out of the box, and generally exhibit high throughput and low latency. They have the ability to store small snippets of data and retrieve them quickly.

Pros	Cons
• Flexible • High throughput and low latency • Scale Horizontally (sharding) by adding nodes • Ability to handle large data size • Fault tolerant High availability via data synchronization across nodes	• Conflicting read & write architectures • Scalability issues

9.2.2 SQL Databases

SQL-type Databases include SQLite, MySQL etc

SQL databases are relational, and require that schemas, that describe how information is organized, to be defined prior to implementation. This makes them highly manageable. However, they run into issues scaling effectively. While most SQL databases were built for client/server architecture, there are purpose-built SQL databases for Edge-Storage (e.g., SQLite). Tradeoffs are:

Pros	Cons
• Small footprint • Good data integrity with built in validation capabilities • Strong analytics for data with complex relationship • ACID compliant • Stable	• Fixed data schema • Scale Vertically by increasing memory size, disk size and CPU • Performance deteriorates with large data size • Limitation in database size

9.2.3 Time-Series Databases

Time-Series Databases include InfluxDB, Prometheus, Riak, OpenTSB, etc.

Time-series databases, as the name suggests, are designed specifically for indexing and optimized for querying time-series data, making them perfect for storing most IoT sensor data or capturing change events in the devices. Most of these time-series databases come with built-in continuous query and calculate rolling metrics, and are ideal for real-time analytics for time-series data.

Pros	Cons
• Index time-series data • Built-in continuous queries to compute data aggregation	• Relatively new databases

- Built-in linear interpolation for missing data
- Supports automatic data down sampling
- Easy to learn – SQL-like query language

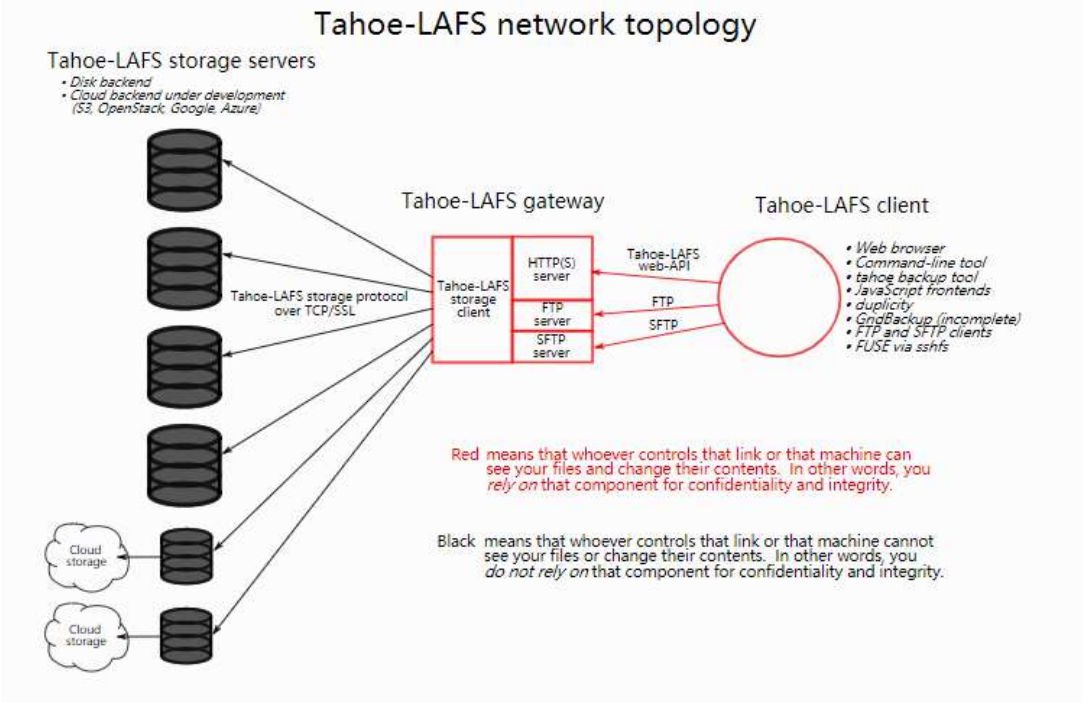
9.3 Potential Storage Ecosystem Collaborations

As a mesh network node, MeshBox not only has the function of supporting network connection, but also has the characteristics of storage. The widespread distributed deployment of MeshBoxes can be leveraged for distributed storage and Edge-Computing. Storage Layer supports the development of applications for IoT data storage. MeshBox users can simply and safely use the distributed storage system deployed in MeshBox for encrypted storage and sharing of files.

9.3.1 Tahoe-LAFS Distributed Storage

MeshBox Foundation has done some interoperability tests with the Tahoe distributed Storage.

Tahoe-LAFS is a free and open decentralized cloud storage system. It can distribute data across multiple devices. Even if some equipment fails or is taken over by an attacker, the entire file storage will continue to operate normally, thereby protecting data privacy and security.



The main technical characteristics of Tahoe are as follows:

(1) Tahoe encrypts the data (backup, slice data) before it reaches each storage device

(2) Tahoe adopts distributed file storage, that is, the content of the file cannot be viewed and reviewed without authorization

(3) Tahoe replaces traditional file access authorization verification with variable-length strings, which has high security, basically cannot be cracked online, and is completely anonymous, without registration and identity verification processes.

(4) The Tahoe file storage process includes: uploading a file through the client, the file is divided into file fragments before uploading to some storage devices, and uploaded to T storage devices; each device only retains a part of the file, which is not complete. The file is backed up; and the file name is replaced by a string of encrypted strings; after the upload is complete, Tahoe returns a string of XXX through the client, that is, XXX is the only credential to access the file, for example, URI: SSK:234hv34x1t43a6t7ft76iaa3oa:35633ebsf2yrfghn55xo7c5oh3we2rvgi32da930r23sbr7t2rvgi32da930r23sbr7t2.

(5) For the storage service, only a part of the file is encrypted, and the file name, source and content of the file cannot be known, and "a certain file" cannot be deleted.

(6) Tahoe distributes files and retrieves (restores) files according to the set threshold. A file is split into T shares, which are randomly (evenly) distributed among the available storage nodes. If one needs to rebuild files, one only need Return the T'(T'<T) shares. That is, even if a T-T' storage device fails to retrieve its file share, the file can still be recovered. Since more storage nodes mean better resistance to failures or attacks, this sharing mechanism is suitable for remote and secure storage of sensitive data while reducing the risk of data loss.

MeshBox Foundation and Tahoe-LAFS have successfully built a Tahoe operating environment on MeshBox, and conducted performance testing and joint development of a simple application. The application can realize the functions of uploading and saving encrypted files, exporting private key of the folder, file import and retrieval via different device, file sharing, etc.

9.3.2 IPFS

The Inter-Planetary File System (IPFS) is a peer-to-peer distributed file system that seeks to connect all computing devices with a common system of files. In some ways, IPFS is similar to the Web, but IPFS could be seen as a single BitTorrent swarm, exchanging objects within one Git repository. In other words, IPFS provides a high throughput content-addressed block storage model, with content addressed via hyper-links. This forms a generalized Merkle DAG, a data structure upon which one can build versioned file systems, blockchains, and even a Permanent Web. IPFS combines a distributed hashtable, an incentivized block exchange, and a self-certifying namespace.

9.3.3 Swarm

Swarm is a system of peer-to-peer networked nodes that create a decentralized storage and communication service. The system is economically self-sustaining due to a built-in incentive system enforced through smart contracts on the Ethereum blockchain.

Swarm provides continuity of service and resilience against network outages or targeted denial of service attacks. As a platform for permissionless publication, Swarm fosters freedom of information. With its exceptional privacy features like anonymous browsing, deniable storage,

untraceable messaging and file representation formats that leak no metadata, Swarm responds to the growing demand for security on the web.

Built-in incentives seek to optimize the allocation of bandwidth and storage resources and render Swarm economically self-sustaining. Swarm nodes track their relative bandwidth contribution on each peer connection, and excess debt due to unequal consumption can be settled in BZZ. Publishers in Swarm must spend BZZ to purchase the right to write data to Swarm and prepay rental fees for long term storage.

9.3.4 IPFS and TEE

Some decentralized storage architectures for Web3.0 support multiple storage layer protocols such as IPFS, and exposes storage interfaces to the application layer, which values data privacy and ownership. Various consensus paradigms can be used, such as Guaranteed Proof of Stake (GPoS), which requires nodes to provide storage resources as a guarantee to obtain staking quota, and encourages users to stake their tokens to high quality nodes via a guarantee operation to obtain staking income.

Such an architecture enables users to backup their data, sharded, coded, and distributed on many nodes of the network. The coding overhead is a configurable parameter, such that if some maximum percentage of nodes fail (based on the coding overhead), the entire data can still be recovered from the remaining working nodes.

TEE (Trusted Execution Environment) technology may also be used, such as using a MPoW (Meaningful Proof of Work) consensus to quantify meaningful storage resource usage and generate a corresponding work report reliably.

10 Execution Layer Architecture

To support the IoT Use-cases described, generalize, and optimize business processes, the HyperMesh Architecture supports the following Interconnection, Storage, and Execution functions. including

- Fractal Task Graphs with both internal (inside on-chain smart-contracts) and external transfers (between on-chain smart-contracts)
- Fractal Job Graphs with both internal (inside off-chain smart-contracts) and external transfers (between off-chain smart-contracts)
- General triggering of ready Tasks/Jobs based on the Digital-Twin state (including Storage Layer Data and State variables)
- Scheduling of Task/Job execution, with associated Storage Layer data transfers needed for execution.
- Cyber-physical resource sharing for supply-chain type of business processes
- QoS and Real-time guarantees on completion of N executions of a Job Graph
- AI and ML on Storage data, when permitted by the user(s) who own such data.

10.1 Business Processes as Task Graphs, Implemented on Spectrum

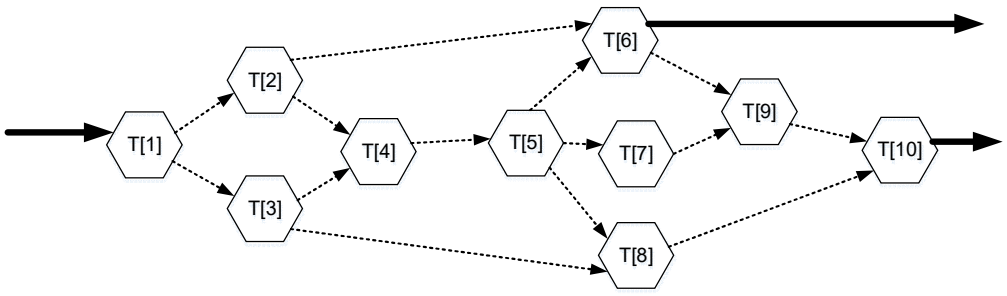
Task-Graphs (TGs), are used to model and implement secure business processes, which can be executed an infinite number of times. Task-Graphs, implemented as Spectrum Smart-Contracts, are adequate when performance can be low (TPS, latency requirements)

Task Graphs can represent

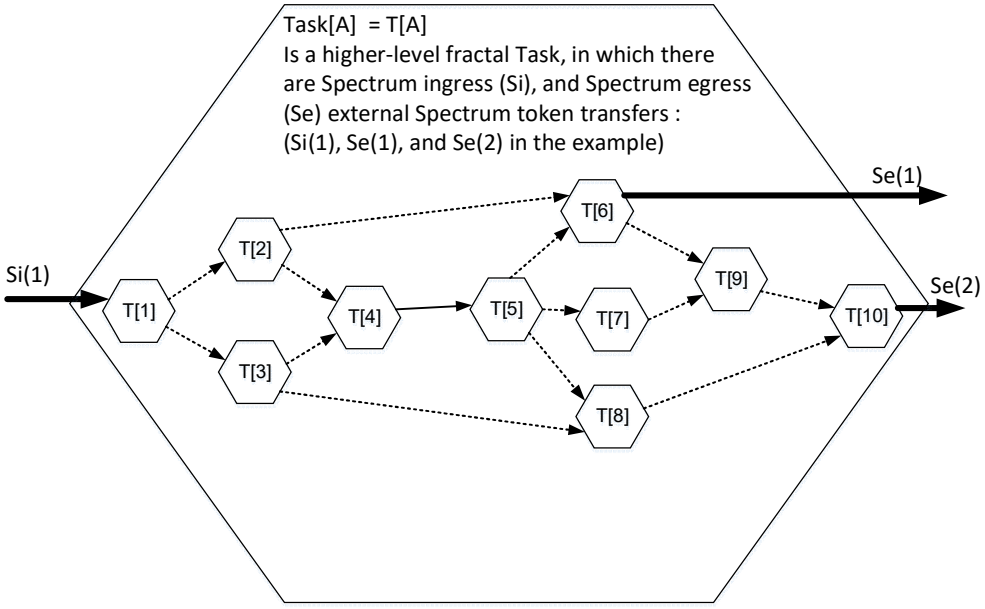
- Supply-chain goods transfers
- Business processes (buying real-estate)
- Aid Distribution
- Smart-Farm management
- Smart-Home management
- Etc.

Spectrum Task-Graphs are implemented with Spectrum nodes (hexagons) and Spectrum transactions (arrows)

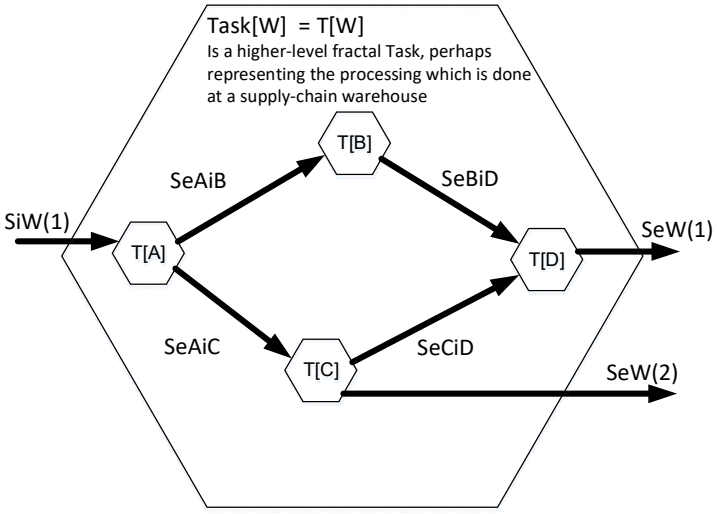
The following shows a Smart-Contract Task Graph, Task[A], in which all dotted line arrows represent internal transfers inside a Smart-Contract.



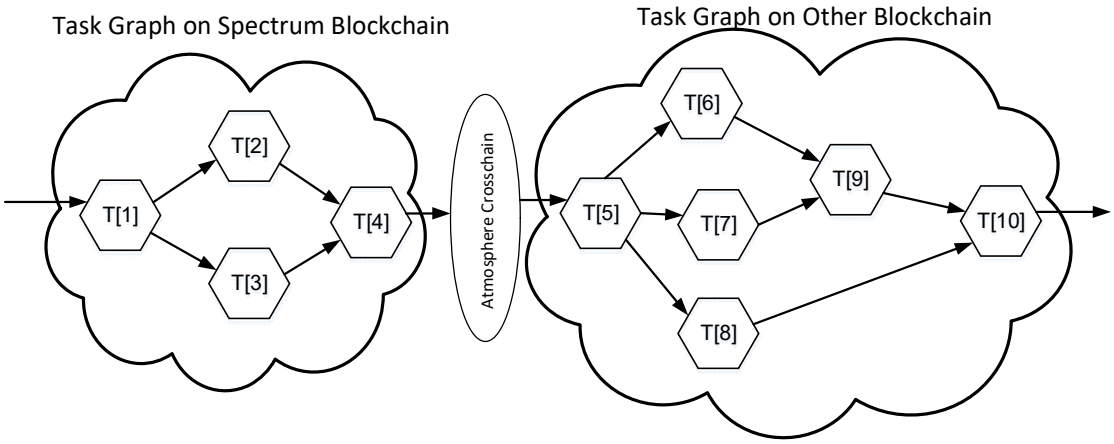
The following further shows the internal and external transfers of Task[A], and how Task[A] is represented as a higher-level Fractal Task.



The following Task Graph is composed of Tasks A, B, C, and D, in which the solid line arrows are part of the Interconnection Layer, and represent Token transfers between Smart-Contracts.



SmartMesh Atmosphere bridges from Spectrum to other blockchains (Mesh Sub-chains, Ethereum, Bitcoin, Hyperledger, etc) in order to support interoperability between various business applications (running on different blockchains) together.



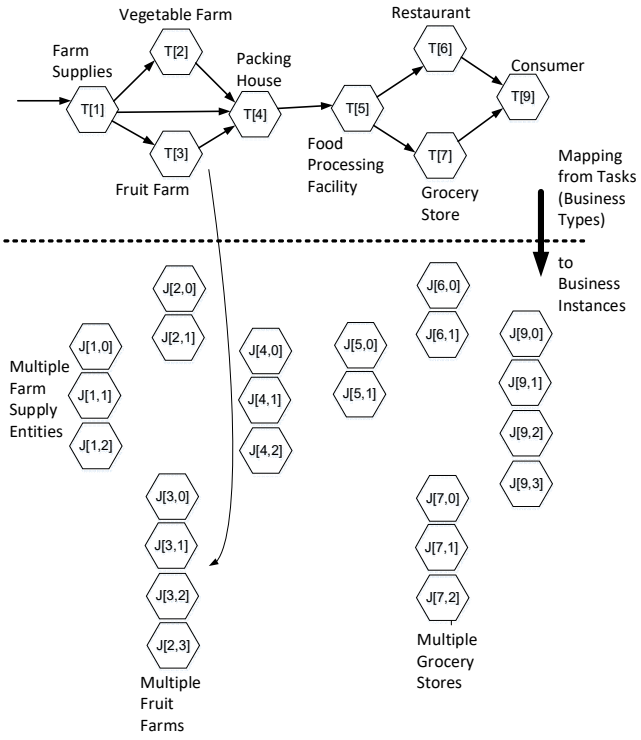
Sub-Chains may be geographically sharded, or Sub-graphs which have much higher internal transfer rates than external transfer rates. Sub-Chains from the Spectrum main blockchain can be used for scalability.

10.2 Job Graphs and Application Example

If a TG (corresponding to a Spectrum Smart-Contract) should be executed at high frequency, and/or sometimes offline, the TG can be converted to an analogous Job-Graph, which is associated with a finite number, N, of executions.

To attain certain performance goals (TPS, load balancing of shared resources, real-time deadlines) in a shared-resource environment, JGs are optimally mapped to the physical world nodes IoT and Decentralized HyperMesh Architecture nodes (e.g. MeshBoxes in a mesh network for supply-chains). Job-Graphs are implemented with Photon nodes (hexagons) and Photon channels (arrows)

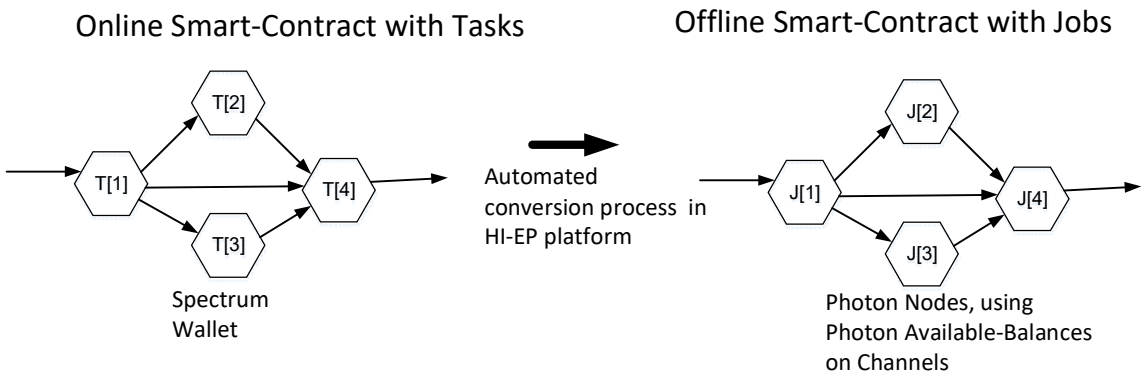
The following is an example of a supply-chain, which is represented as a Job-Graph. Each node of a JG has the potential to be mapped to one or more physical nodes, which are potentially different business sites. Each business site holds various machinery, equipment, or communication nodes, such as MeshBoxes. Leveraging real-time systems and resource allocation theory, it is possible to optimize the Supply-Chain to meet various performance (such as load-balancing and deadline) guarantees.



10.3 On-Chain to Off-Chain Smart-Contract Mapping

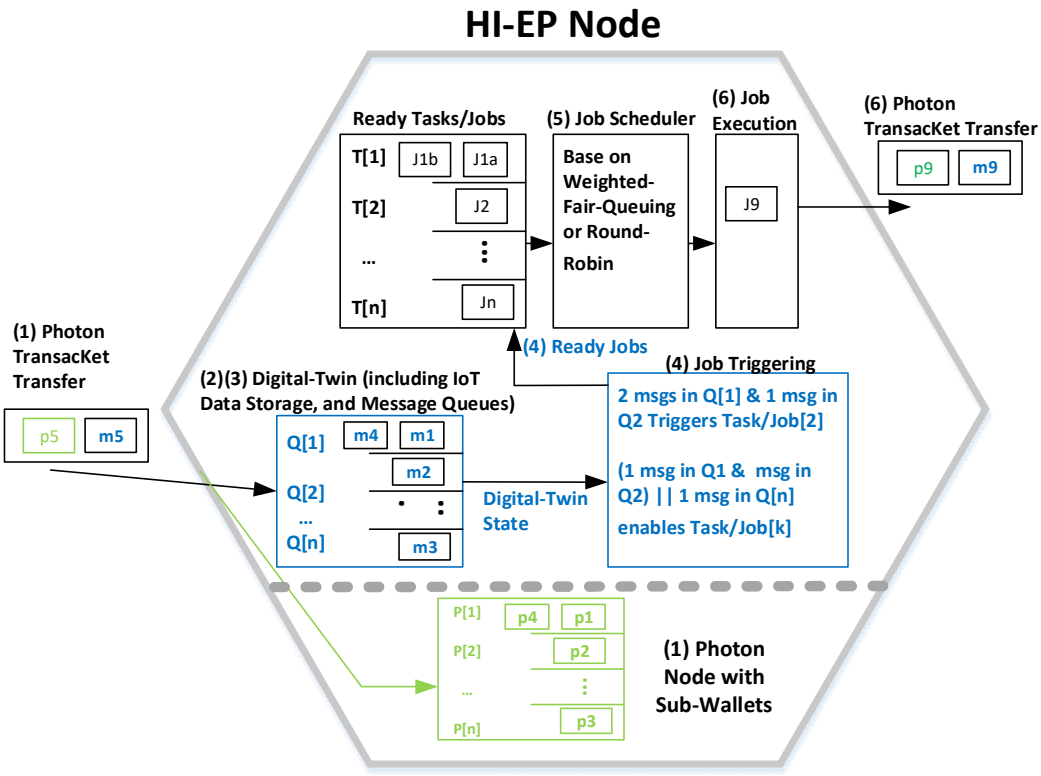
A business process, such as Transactive IoT, may initially be executed as a Spectrum smart-contract. However, if such a smart-contract is executed at high frequency, it may be better to run the process as a Photon-enabled Off-line Smart Contract.

The goal is to auto-convert an online smart-contract to an offline smart-contract, which also requires mapping to the physical world, along with QoS guarantees.

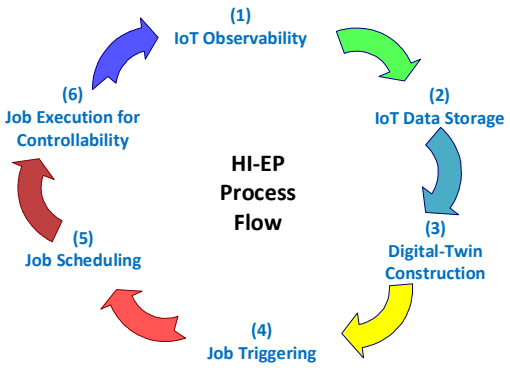


10.4 Offchain-Smart-Contract Execution Layer Architecture

In the Execution Layer Job-Graph node below, all data structures, such as the Message Queues and Ready Tasks/Jobs Queues, as well as the Data-bases holding all IoT and related data, are all part of the Storage Layer.



An Execution Layer node, which can run on a MeshBox, has the above architecture and supports the following functionality, as previously defined in Section 2.5. The generalized process flow diagram is mapped to the Node architecture.



(1) IoT Observability: IoT Devices gather data; transmit to LPWAN Access Points. This is done via a LPWAN wireless protocol such as LoRaWAN. Execution Layer node receives a

TransacKet (consisting of a payment amount and data message) via a Photon transfer. The Photon transfer can be directed to a Photon-Sub-Wallet, indicated in the Execution node.

(2) IoT Data Storage: Edge-Storage nodes store data and may perform simple aggregation of such data, if so instructed. Messages, representing events, can be either In-band (entire message is present) or Out-band (e.g. Hash pointer to message, stored in Storage Layer)

(3) Digital-Twin Construction: A digital-twin of the Mesh-Subchain area-of-interest is constructed using Internet information, measured IoT data, and pre-determined triggering conditions. Received Messages stored in Message-Queues are yet another aspect of the Digital-Twin state.

(4) Job Triggering: When state variables in the Digital-Twin meet predefined triggering conditions, the associated Jobs become Triggered. Programmable triggering conditions on the state of the Message-Queues determine when Jobs are triggered, becoming ready to be executed, i.e. Ready Jobs.

(5) Job Scheduler (for Off-chain smart-contracts) determines which Ready Jobs should be executed, based on pre-determined QoS guarantees.

(6) Job Execution for Controllability: the execution of the Jobs is performed, and results in Control actions (such as charging an electric-vehicle, or adjusting the temperature). Execution can involve processing of the Message contents (e.g. Database operations, Message management, big-data Machine Learning, and analytics), as well as the generation of output TransacKets.

10.5 Decentralized Programming Model

In conventional parallel-processing on multi-core CPU environments, multiple levels of private and shared memory are used to store the data for processing by tasks/jobs. Usually, a globally shared-memory space is assumed, which requires cache-coherency protocols for each memory cache-line, such as MESI (Modified, Exclusive, Shared, Invalidate).

HyperMesh considers a message-passing programming model, in which data is not globally-addressed by default. For instance, in decentralized data-storage architectures, such as Tahoe-LAFS, and IPFS, data is accessed by a secret hash. Only the owner of the data, and those entities to which the owner has given access, will possess that secret hash. The owner can also monetize the sharing of the hash to others. This adds another layer of security, and mitigates Machine-learning AI from being able to access vast amounts of data for good or ill.

On the other hand, blockchain is used to ensure data immutability and transparency. The owner of data can also choose to share that secret hash to the entire blockchain, in which case all nodes are able to query the blockchain to get the most recent value, through timestamped transactions on the blockchain.

10.6 Potential Execution Ecosystem Collaborations

As a mesh network node, MeshBox not only has the function of supporting network connectivity, but also has the characteristics of storage and computing. The widespread distributed deployment

of MeshBoxes with Execution Layer can be leveraged to build Edge-Computing applications. MeshBox Foundation is considering collaboration with the following technologies.

10.6.1 Decentralized Computing (Edge-Computing)

Decentralized edge-computing networks aim to use idle computing capacity from various computers to process tasks. Such approaches can be used to build asset-light, low-cost, high-availability computing architecture. A verifiable Edge-Computing model may also be based on TEE, which can run on ARM CPUs, X86 servers, and edge devices. Parallelizable tasks such as MapReduce are a good fit for such an approach.

The main features include:

(1) High availability: Heterogeneous P2P resource management, with large-scale distributed scheduling network and several redundant nodes

(2) Large-scale data processing: Based on the public network, supporting heterogeneous MapReduce computing frameworks including Android phones

(3) Trusted computing: Big data can be processed based on TEE, while solving the problems of computing scale and privacy security

(4) Low cost: A fair incentive system that makes full use of idle social resources to realize asset-light cloud computing in a sharing economy model.

Application scenarios mainly include

- Low-cost big data analysis services,
- Edge-Computing scenarios,
- Trusted data exchange networks,
- Privacy-protected joint computing, etc.

MeshBox has successfully run a decentralized edge-computing application. Algorithms executed on such a system can be used to implement subsequent IoT Edge-Computing applications.

10.6.2 Data Privacy

Some decentralized storage solutions utilize a distributed network with the goal of supporting privacy focused applications and use cases.

Whether data is being shared data externally, or controlled for use internally, such solutions provide an immutable, tamper-proof record of all actions taken against a dataset. This log acts as a public ledger that parties can access to ensure that such access is as expected and compliant.

One such solution combines confidential compute with blockchain technology to create a secure and privacy-preserving trust layer for user data. Features include

- Automatic data ownership and access policies；
- Analyzing data in a privacy-preserving environment, such as based on the TEE (Trusted Execution Environment) technology；
- Unlocking data related applicatiosn which were previously too regulated or risky to use.

10.6.3 Distributed Services on Smart Devices

Some decentralized solutions offer a hosted market for distributed applications. Some offer an open-source framework for peer-to-peer applications, which enables distributed hosting services provided by peers.

One such solution uses a cryptographic fabric to maintain data integrity across many peers without requiring consensus. Integrity without consensus means:

- immediate and efficient processing,
- no proof-of-work,
- no proof-of-stake,
- no energy wasted on mining,
- no bottlenecks nor global delays.

Scalable crypto-accounting allows to build new generations of asset-backed and value-stable cryptocurrencies. Programmers build P2P web applications designed to operate on the scale of Twitter or Facebook with no centralized data centers or infrastructure. Each user just brings their own device and shares a small amount of computing and storage.

Suc a solution enables people to

- own their own data,
- control their identity,
- have automatic backups,
- customize their user experience,
- choose how to connect their applications,
- decide with whom to share their private information,
- and transact without dependence on banks or governments.

11 Conclusion

According to IDC, by 2025 42 billion IoT devices will be connected to internet generating 73 zettabytes of data. IoT data are used to make potentially costly business and/or mission-critical decisions. Thus, certifying and authenticating the devices and the data which they produce and consume is a must. IoT solutions are also fragmented and do not interoperate.

To address such issues, a highly vertically-integrated HyperMesh Architecture is proposed, on top of which IoT applications are built. HyperMesh Architecture leverages MeshBox Edge-Networking nodes, and enables the Web 3.0 paradigm -- a decentralized, open and trustless network for peer-to-peer interactions without intermediaries.

Critical security requirements for IoT devices and data are covered by the SmartMesh Spectrum blockchain, specifically designed for Edge-Networking and IoT applications. SmartMesh Photon payment network, a Layer-2 scalability solution for Spectrum, supports Token-switching. Here, users are the rightful owners of their data; and both tokens (holding value) and data are conveyed in the same protocol.

SmartMesh DeFi offers monetization of IoT observability and controllability functions, which incentivize deployment, leading to wide-scale proliferation of IoT applications.

The proposed HyperMesh enables blockchain based inclusive finance, high-throughput wifi-based internet connectivity, and a distributed content storage system, with high-availability and reliability, operating

- with or without the Internet
- with or without a blockchain; with Photon state-channel secondary architecture,
- with or without an electrical grid; with renewable battery and solar technology.

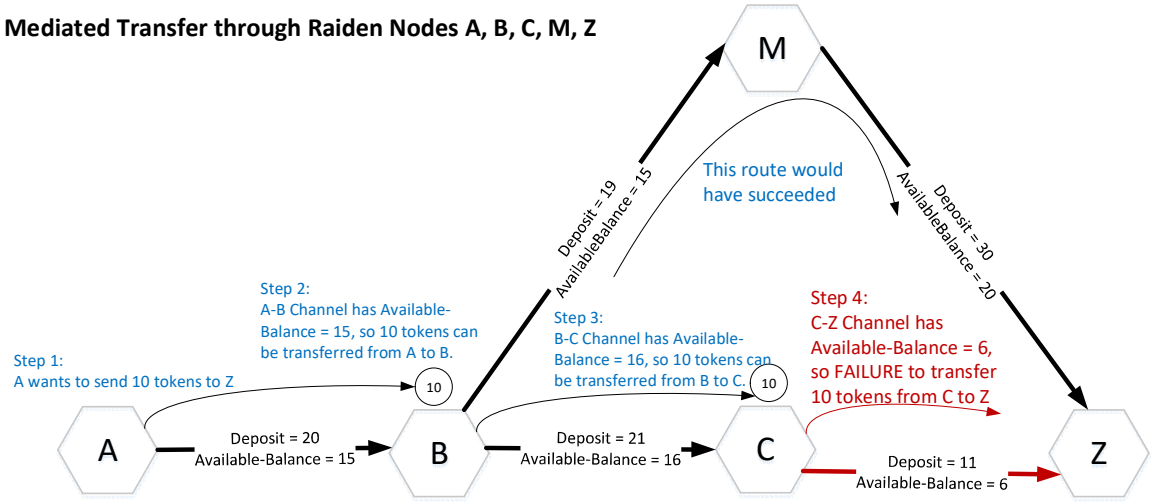
SmartMesh®, MeshBox® and ecosystem partners such as Satellite operators, LoRaWAN technology providers, and application builders work towards the Web 3.0 paradigm shift, which incentivizes deployment with token rewards. Instead of a traditional service providers determining when and where to roll out service, the residents of a community decide when and where to deploy infrastructure such as MeshBox® HyperMesh nodes.

Stay tuned for future articles which detail how the HyperMesh™ Architecture, enabled with SmartMesh® and MeshBox® technology, can speed up the realization of the Value Internet.

We aim to bring dignity and a sustainable livelihood for the 3.7 billion people without internet access and the 1.7 billion people who are unbanked.

12 APPENDIX: Photon Architecture Details

Before discussing Photon, a basic State-Channel Payment-Network architecture (such as Ethereum Raiden or Bitcoin Lightening) is first described. The following shows an arbitrary payment network, composed of an interconnection of State-Channels (Channels) between Raiden nodes, which are required to be connected to the Internet.



A transfer is successful when a Route can be found from Initiator (A, the payer) to Target (Z, the payee). Payment (Transfers) can be either Direct-Transfers, which take place directly between two Photon nodes, or Mediated-Transfers, which are routed through intermediate payment nodes.

One issue is that, since Channels are setup by users, the associated Deposits (tokens which users must lock into the system) will be minimal, and sometimes, routes cannot be found between Initiator and Target. Routing from Payer to Payee has high chance to fail, due to

- Complicated Topology
- Inadequate Available-Balance in possible routes

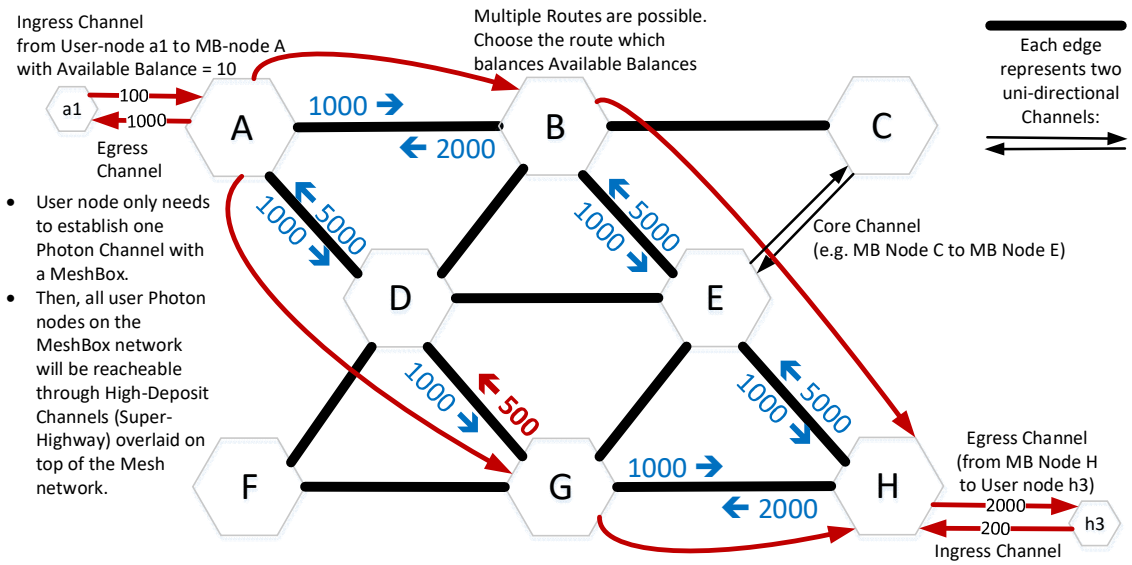
Furthermore, only small Payments are supported due to small user-defined Channels.

Also, conventional payment networks cannot work without continuous connection to the Internet.

To solve the above limitations with conventional payment networks, SmartMesh® has developed Photon, which is also a payment network, but which is optimized to work on MeshBoxes, which, in-turn, do not require continuous Internet connectivity.

> Using MeshBoxes as the “super-highway” for mediated transfers resolves many of the issues of conventional, user defined payment networks. Merchants or local operators who own the MeshBoxes will be incentivized to support high-deposit channels in order to facilitate e-commerce in the community.

Using statistical multiplexing and economy of scale benefits, high-deposit Core Channels between MeshBox® Photon nodes, with the AI/ML based routing, ensures that routes are quickly found between payer and payee.



The benefits include:

- MeshBoxes are Photon nodes with large deposits on Channels
- Routing through topology is simplified
- Large deposits reduces routing failures
- Large transfers are supported

Photon is extended through Wormhole Universal Channels, and works in the Atmosphere architecture to enable transfers between various tokens, with highly scalable Transactions per Second (TPS) performance with the number of Photon nodes. Thousands of Photon nodes work together to support thousands to millions of TPS.

The Spectrum, Atmosphere, Photon, and Wormhole Universal Channels architectures are directly applicable to inclusive financial projects to connect numerous blockchains and tokens together and enable interoperability.

Thus, the SmartMesh Atmosphere architecture brings about the “Token Switching” era, to differentiate against the previous “Packet Switching” and “Circuit Switching” eras.

Here is how the accounting works for the Channels.

At Creation, each Channel has a Deposit (=D), which is recorded on the Spectrum blockchain

When a Channel is settled, each side gets what is due

- A gets AvailableBalance(A_B)
- B gets AvailableBalance(B_A)

The AvailableBalance of the Channel from A to B (A_B) is a function of

- X(A_B) = Sum of all DirectTransfer amounts (historically), which is the amount which A owes to B directly. Initial value = zero.
- X(B_A) = Sum of all DirectTransfers (historically) which B owes to A directly. Initial value = zero.
- Y(A_B) = Sum of all LockedTransfer amounts for all pending mediated transfers passing through the A to B Channel (e.g. the mediated transfer a1—A—B—E—H—h3), which are the locked amounts of MediatedTransfers which may or may not succeed.
- Y(B_A) = Sum of all LockedTransfer amounts for all pending mediated transfers from B to A

Then, the AvailableBalance of the channel (A_B) is given by :

AvailableBalance(A_B) = D – X(A_B) + X(B_A) - Y(A_B)

13 APPENDIX: SGIN Fractal Network Architecture

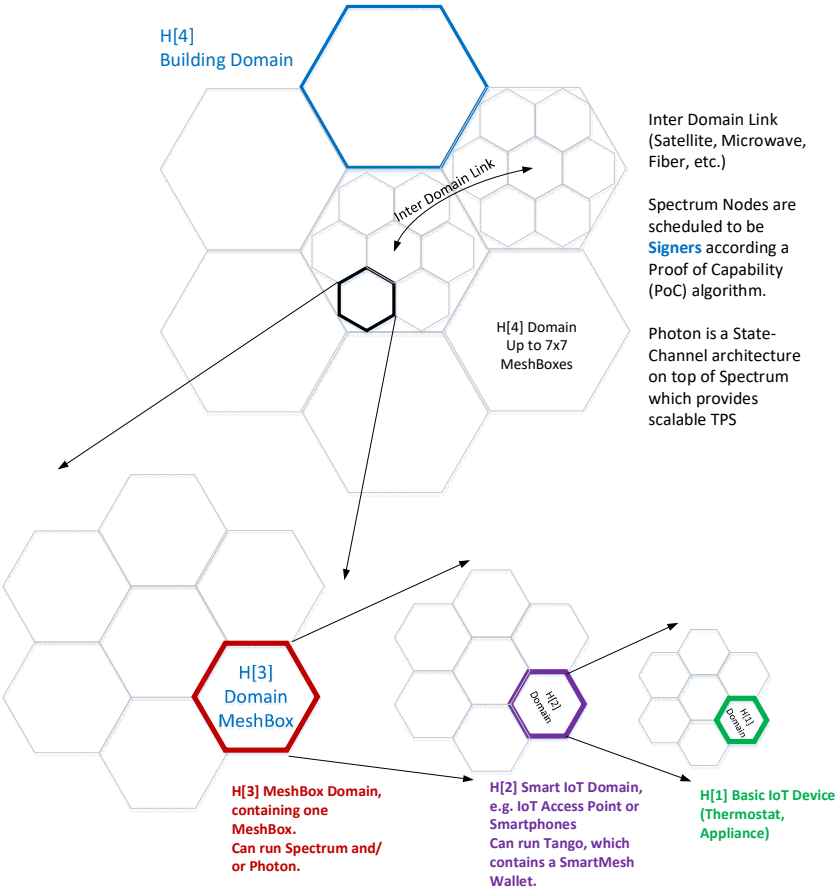
The following shows a Fractal [FRACTALS] architecture which is used to scale SGIN using Satellites and MeshBox networks. Scalability spans from the lowest-level Domain of IoT devices, all the way to a Universal Domain.

The hierarchical levels are defined as follows:

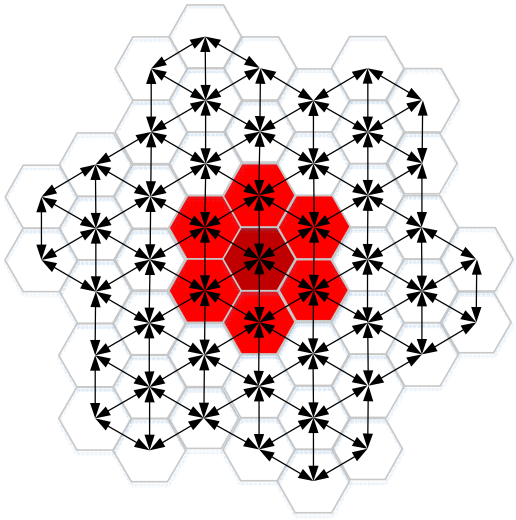
- H(11) = Universal Domain = Domain which spans to beyond the Earth, including networks to the Moon, Mars, and beyond.
- H(10) = International Domain = Span multiple countries. Initial focus is on China and the Association of South East Asian Nations (ASEAN) countries: Philippines, Indonesia, Brunei, Malaysia, Myanmar, Laos, Vietnam, Cambodia, Thailand, and Singapore.
- H(9) = Country Domain = Spans a Country, such as those in Southeast Asia.
- H(8) = Region Domain = Spans Region (State or Province)
- H(7) = City Domain = Spans a City or Village
- H(6) = District Domain = Spans a district within a City
- H(5) = Community Domain = Spans a Community, which can be a block or a group of buildings managed by one entity, or jointly owned by a community.
- H(4) = Building Domain = Spans a house or building (or several inter-connected buildings)
- H(3) = Single MeshBox Domain
- H(2) = Smart-devices Domain, such as IoT access points, and smartphones
- H(1) = IoT device Domain

Satellite Internet can be used at hierarchy levels H(11) down to H(4), while the MeshBox-based Mesh networks can cover from H(7) down to H(1). In the figure below, the Hierarchies from a

Building Domain (H[4]) down to an IoT domain (H[1]) are shown.



Each Hierarchy Domain is shown with 7 Sub-Domains. However, the number of such Sub-Domains can be up to 49 (=7x7) as shown below.



On such a Fractal network architecture, various HyperMesh Architecture networks are overlaid, including communication, Photon peer-to-peer payments, Wormhole Universal Channels for multi-blockchain token exchange, Energy Internet (Enernet), etc. The exchange of various tokens, both fungible (monetary) and non-fungible (non-monetary, cyber-physical), are supported.

14 APPENDIX: IoT Data Authentication

The following is based on the Authentication Schema from the [IoTeX] paper. Atmosphere is used to crosschain between various Subchains.

The Authenticity Schema is maintained by the sub-chain ledger MS_s instead of the IoT device itself. Therefore, an IoT device does not need to manage the session by itself. The Schema includes six fields, which can be implemented in a TEE, for instance.

(1) Sender: Proof of the sender IoT device

(2) Data unit: Depending on characteristic of the IoT device, data transmission unit can be 1KB, 1MB, or other sizes

(3) Data authenticity mechanism: IoT devices can choose different methods to authenticate the generated data, such as Hash-based Message Authentication Code (HMAC) and Cipher-based Message Authentication Code (CMAC).

(4) Key information: Key used in the selected data authenticity mechanism

(5) Key update frequency: After the authentication key is used for a specified number of data units, the IoT device sending the data will update the key and deactivate the old key when each new data unit is authenticated

(6) Life cycle of the Schema: How many times the authentication key can be updated.

Each IoT device can choose parameters in the schema that is applicable to its specific needs.

The Key field in the Schema is hidden when initially created, in order to prevent potential adversaries from generating a valid MAC to forge data. However, the Key must be exposed at a later time to allow other users to verify the MAC of the sender. Therefore, the Encryption Promise Schema is used to temporarily hide the Key. In order to reduce the computational cost in IoT devices, the HMAC-based Encryption Promise Schema is used. IoT Device $d_{s,j}$ initially submits its digital signature and data authentication Schema to the sub-chain ledger MS_s for authentication and storage. Endpoint data users can get Schema information from MS_s and if user is not connected to MS_s directly, it can request Schema information from the ledger nodes it is connected to (using Atmosphere cross-chain protocol for information exchange).

14.1.1.1 Data Transfer

After the data authentication Schema is accepted by the sub-chain MS_s, the IoT device can start sending data. The device uses two channels for transmission:

- Data channel
- Meta-data channel, containing authentication and data protection Schema fields

The data is sent to the connected MeshBox edge-server and/or a cloud-server for storage and processing. The corresponding Meta-data (including authentication protection information) is only sent to the sub-chain ledger MS_s.

The data itself is divided into one or more segments and IoT device creates a MAC for each segment using different key. When (t+1)th segment out of the total n segments is sent to the edge and cloud server, t segment of the sender's MAC key is also sent to and disclosed in the sub-chain ledger, which can be used to validate the corresponding t-th data segment. The final MAC key k_n

is selected by the sender device and the rest of the MAC key is derived from k_n using the key refreshing method.

Data from IoT devices is collected and stored by edge-servers and cloud-servers, and the main/sub chain architecture is responsible for maintaining authenticity. When data segment seg_t is received and stored by the edge/cloud servers, the corresponding edge ledger cannot immediately verify its authenticity. The sub-chain ledger waits for the corresponding authentication protection key k_t, which is provided by the IoT device at time period t+1. Then, each sub-chain node performs the following checks on k_t:

$$\text{if } \mathrm{mac}_t = MAC(seg_t, k_t), t \geq 1 \; and$$

$$\text{if } \mathrm{H}(k_t \| t-1) = k_{t-1}, t > 1$$

Here, k_{t-1} is the previous data authentication key and has been stored in the sub-chain ledger. The sub-chain ledger nodes decide whether to include k_t in the ledger through consensus. The data consumer obtains authentication information from the main/sub chains and authenticates whether the data has been tampered with. The main/sub architecture maintains a large number of keys to assist IoT data authentication and protection. Each key is embedded in a transaction. During the operation of the architecture, the node can obtain the corresponding key by querying the ledger for data authentication and IoT device authentication.

15 References

[Atmosphere] SmartMesh whitepaper, SmartMesh Spectrum Scalability and Extensibility Architecture via Atmosphere and Photon, July 23, 2021.

[Photon] SmartMesh whitepaper, SmartMesh Spectrum Scalability and Extensibility Architecture via Atmosphere and Photon, July 23, 2021.

[SGIN] MeshBox Whitepaper: *Space-Ground Integration Network for SUNSHINE* , April 07, 2020.

[RA4M2] https://www.renesas.com/us/en/products/microcontrollers-microprocessors/ra-cortex-m-mcus/ra4m2-100mhz-arm-cortex-m33-trustzone-high-integration-lowest-active-power-consumption?utm_campaign=mcu_ra4m2&utm_source=google&utm_medium=ppc&utm_content=ra4m2

[IoTeX] Lei Xu, Lin Chen, Zhimin Gao, Xinxin Fan, Taeweon Suh, and Weidong Shi, *DIoTA: Decentralized Ledger based Framework for Data Authenticity Protection in IoT Systems,* University of Texas Rio Grande Valley, Computer Science Faculty ublicaitons and Presentations, College of Engineering and Computer Science. February, 2020.

[Paygo] UN Climate Change Newsroom, Using Pay-As-You-Go Solar Home Systems in Sub-Saharan Africa, March, 2017

[MB++] MeshBox++™ is designed by Mesh++ Inc., a close partner to SmartMesh® and MeshBox®

[PLDT] 3. PLDT Wired Internet Costs https://www.imoney.ph/broadband

[WSJ] https://www.youtube.com/watch?v=9gi_0KR80TQ&t=197s. John Hodulik, UBS Media and Telecom Analyst.

[EEN] Jonathan Koomey, MIT Technology Review The Computing Trend that Will Change Everything, April 9, 2012. https://www.technologyreview.com/s/427444/the-computing-trend-that-will-change-everything/

[Seba] James Arbib and Tony Seba, Rethinking Transportation 2020-2030, The Disruption of Transportation and the Collapse of the Internal-Combustion Vehicle and Oil Industries. May 2017

[Seba] [GridWise] [PowerMatcher] [TeMix]

[GridWise] GridWise Architecture Council, US Department of Energy, GridWise Transactive Energy Framework Version 1.0, January, 2015

[PowerMatcher] Koen Kok, The PowerMatcher: Smart Coordination for the Smart Electricity Grid, Dissertation at Dutch Research School for Information and Knowledge Systems, sponsored by the Netherlands Organisation for applied scientific research TNO, May 13, 2013

[TeMix] Edward G. Cazalet (CEO, TeMix Inc), Business and Regulatory Models for Transactive Energy, GWAC Transactive Energy Conference, www.tea-web.org, December 10, 2014

[FRACTALS] Peter Yan, FRACTALS Realtime Autonomic Control Anti-fragile Layered System Architecture -- SmartMesh® and MeshBox® Ecosystem, September, 2018

[Helium] Amir Haleem, Andrew Allen, Andrew Thompson, Marc Nijdam, Rahul Garg, Helium A Decentralized Wireless Network, Helium Systems, Inc. Release 0.4.2 (2018-11-14)

