SCTE-ISBE CABLE-TEC EXPO 2018 ATLANTA, GA OCTOBER 22-25

2018 Fall Technical Forum SCTE·ISBE • NCTA • CABLELABS

Navigating IoT Technologies, Standards and Frameworks for Managed IoT Service

A Technical Paper prepared for SCTE•ISBE by

Gary Gutknecht
CTO Connected Home
Technicolor
Sugarloaf Pkwy, Lawrenceville, GA
+1-317-809-2417
gary.gutknecht@technicolor.com

Table of Contents

Title **Page Number**

List of Figures

List of Tables

Introduction

Network Service Providers (NSPs) have a major opportunity and advantage in offering Managed IoT Services. They have the organizational and business structure to successfully build all the necessary IoT layers of Connectivity & Networking, Core IoT Platform and Services. However, IoT technologies and ecosystems are complex and fragmented making it challenging for Service Providers to formulate a winning strategy. A significant number of complex heterogeneous IoT options for sensor connectivity, networking and application layers make it challenging to understand the best solution for targeted use cases. Functional overlap is pervasive when considering networking and IoT sensor application layer options such as Thread, Open Connectivity Forum and Dotdot, which makes it difficult to pick the best approach and understand how they will work together. IoT connectivity protocols and standards such as Wi-Fi, Zigbee, Z-wave, BLE, NB-IoT, LoRa and SigFox can be confusing without an understanding of their technical features, optimizations and use cases. In addition to these challenges, IoT solutions connect to their closed service layers using different messaging protocols (CoAP, MQTT, HTTP, AQMP), data-models and proprietary APIs, which make service integration difficult. This paper will provide an overview of e2e IoT network layers and make comparison of different IoT technologies in each layer with an emphasis on use case alignment. In addition to topics above, the paper will also include Service Provider e2e considerations such as security, privacy, reliability and scale. A review of harmonization efforts among standards at each layer, and a brief introduction to IoT data-model normalization efforts in the industry (e.g. Semantic Web of Things) will be covered. The reader will gain a clear understanding of the current e2e IoT technology landscape in a structured taxonomy and have a current and practical view of how to apply this understanding to their IoT decisions.

Content

1. Introduction

By 2023, there will be over 50 Billion devices connected to the Internet and much of the device growth over the next 5 years comes from Internet of Things (IoT) for consumer, enterprise and government applications. Of these use cases, the Smart Home IoT device and services market worldwide is estimated to be $138 Billion by 2023 according to MarketsandMarkets (July 2017) and grow at a 13.61% CAGR between 2017 and 2020. The current evolution in IoT which focuses on making devices inter-connected and smarter, builds on the technology evolutions and disruptions we have witnessed in the last decennia around cloud computing and Big Data. Like those technology evolutions, the Smart Home IoT evolution will perform the same paradigm shift from closed ecosystems with lots of industry specific and fragmented standards of today, to an open and common framework to interconnect Smart Home IoT devices and services. Until this point, compliance efforts and integration issues for NSP planning to offer IoT managed services will lead to long development and deployment cycles. These means that Technicolor needs to participate in the market evolution today to gain a leadership role within our NSP customer base. Missing the initial innovation cycle could result in revenue opportunity impact in 2019/20.

1.1 Smart Home Trend

Historically, the Smart Home market started with point solutions (security, home automation etc.) that where closed platforms providing some portal or simple mobile application User experience (UX) for the

consumer to manage the service. The first evolution transition came with the introduction of NEST which was a producer of programmable, self-learning, sensor-driven, Wi-Fi-enabled thermostats (2011), smoke detectors (2013), security cameras (acquired Dropcam), and other security systems.

Table 1 : Smart Home IoT Evolution

	Before 2010	2010 - 2015	2016 – 2020	2020+ (what do we believe)
Breadth	• Point Solution	• Solution Sets	• Broader and Multifunction	• Universal Multivendor Plug-n-play
Sensor Types Introduced	• Fire/Smoke/CO • Motion/Glass Break/Locks • Thermostat • Lighting	• Consumer Video Cams • Smart Speakers • Smart Lighting	• Intelligent Video Cam/Mic • Smart Speaker/Voice Assist • Smart Display • Facial Recognition • Air Quality Awareness • Sound Recognition • Point Function Robots	• Super-Sensors • Augmented Reality • Virtual Reality • Multi-function Robots
Openness	• Closed System	• Simple Open API • Some multi-vendor • Closed hardware	• Robust API • Multi-vendor Automation which is Cloud-to-Cloud	• Open Hardware • Virtual Service Orchestration
UI/UX	• Simple UX	• Complex (Techie) UX	• Consumer UX, CUI?	• AI
Intelligence	• Simple Sensor Intelligence	• Introduction of ML, Analytics and Voice Assist, and Cloud Management	• Pervasive Voice Assist • Advanced ML and AI • Data Analytics	• Predictive • Cognitive
Market Delivery	• Fragmented Network of Distributors, Reseller and VARs	• Emergence of OTT direct to consumer, • Continued Dist./Reseller/VAR	• OTT direct to consumer, • Emergence of Managed IoT SaaS for Service Provider Managed offerings	• Service Provider Managed become significant • OTT direct to consumer decrease

Today, the Smart Home solution space consists largely of silo solutions but with open interfaces for multi-vendor solution to be automated. All IoT solutions for Smart Home have some cloud capability to self-provision and manage devices, often via user friendly Mobile application. Although some multi-function solutions are available, these solution vendors tend to focus on a specific solution area (Lighting, Home Security, Smart Speaker etc.). The IoT industry has started to address multi-vendor interoperability and automation of IoT devices and applications. For example, many if not all smart home IoT solutions have open published APIs for cloud-to-cloud interaction, and common connectivity interfaces to allow more integration of vendor solutions. Due to the success of voice assistant technology from Amazon Alexa and Google Assistant many IoT solution vendors are integrating Amazon and Google into their solution offering. Amazon Echo appeared on the market in 2015 and now dominates with 69% of US Smart Speaker market share in 2017 (source: voicebot.ai) and has an estimated installed base of 20+ million echo units shipped. Globally, this is a small installed base and the voice assistant market is in early market adoption phase but the estimated to grow is significant in consumer (home, car, mobile) and enterprise/commercial applications.

Going forward the emergences of Machine Learning (ML) and Artificial Intelligence (AI) has positively impacted scaling and service value in many categories such as Voice and Video recognition. Many more

innovations will emerge in the coming years as these technologies are advanced and implemented in real-world applications.

NSP's have a major opportunity to participate in the Smart Home IoT value-chain because they bring several advantages in a managed service context. Unlike over the top IoT providers, NSP's have an installed base of connected home subscribers to market and manage IoT solutions.

1.2 Network Service Provider Opportunity for Managed IoT Service

Network Service Providers (NSPs) have strategic assets, operational and business models that, if leveraged, can create key differentiation compared to Cloud IoT managed service player like Google Cloud Platform (GCP) and AWS. NSPs are in the best position to curate a cohesive managed service offering for Smart Home with multiple direct and indirect service models that can be utilized to generate revenue and value. This requires an e2e IoT framework that integrates with NSP network infrastructure, device life cycle management and BSS/OSS systems. The table in this section provides a summary of these potential differentiators.

Despite NSPs having broad capabilities, given the enormous opportunity and the incredible number and variety of competing companies, widespread success in the IoT market will not be easily achieved. NSPs face the persistent threat of disintermediation by over the top (OTT) challengers as well as increasing difficulty deploying new technologies among ageing and diverse internal systems. Technicolor faces on-going challenges as well. Popular OTT services and a general focus shift from raw internet access to multiprotocol connectivity and higher-level services has created an environment where in-home routers and gateways must evolve to remain differentiated and resist commoditization. Additionally, and as is reasonable given their size and maturity, Technicolor and NSPs both move less nimbly than many emerging companies in the space which creates time-to-market challenges that are difficult to overcome.

The Smart Home market is not without major competition for NSPs primarily coming from Cloud IoT Players (Google, Apple, Amazon etc.) which have aggressively position both home networking products (OnHub, Echo) with integrated IoT capabilities as well as Cloud IoT platforms for service delivery. In addition to players like Google and AWS which are directly competing for subscriber in the home, these vendors have also enabled new entrants because they have opened their Cloud platforms and infrastructure as a service (GCP, AWS IoT etc.). The first wave of Cloud IoT platform solutions (2016-2018) that have targeted scaling problems have allowed new entrants to build managed services that compete in traditional telco/cable market space. Cloud IoT solution providers have been able to apply intelligence to complement traditional networking protocols resulting in an augmented user experience and operational scale of these new products. Despite this development, the embedded devices in the home still poses a real challenge since they require specialized skills in embedded and real time development due to their constraints in CPU power and memory budget on top of the required domain knowledge necessary for every specific industry. NSPs have the broad set of capabilities and processes to solve this challenge, that Technicolor can help them achieve this goal.

Table 2 : NSP's Differentiation

IoT Advantage	Details
Customer Base	• NSP have trusted customer base delivering Broadband, Video, Voice and other Home Automation Services. • NSPs have most experience with scaling networks and customers. • Have the marketing and sales channel to help consumers deal with complex choices.

IoT Advantage	Details
Connectivity	• NSP are most experienced in connecting millions of devices, enforcing quality-of-service guarantees and offering pricing plan granularity (per device, usage based), and ability to run analytics engines in the network to manage data flows more effectively and to accelerate response time.
CPE Life Cycle Management	• NSP are most experienced in device enablement, authentication, management, maintenance and replacement in a way that ensures network security, device directory maintenance and rights management.
Customer Management	• NSP have major advantage over non NSP based IoT SPs when it comes to customers care, man power for truck rolls, automation system to measure and optimize customer experience, billing and service pricing infrastructure. Additionally, regulatory requirements at scale can be implemented across products and services.
Vertical Service Creation or Customization	• NSP have key advantage in leveraging common infrastructure to address different vertical market.
Drive Standards	• NSPs can drive vendors to implement standards Helping and supporting standardization efforts in areas like semantic web to overcome the current gaps of device centric communication standards. • NSP needs to standardize on a framework which cover multiple layers and coordinate with SP community but allow device manufacturers to differentiate on algorithms and services that improve the current products in a model that still allows competition.

1.3 NSP IoT End-to-End Layers

NSP's must determine which layer of the end-to-end service they want to add-value or own. At a high-level there are three distinct layers that can be evaluated by the NSP to add value. Starting at the bottom we have **Devices**, making up the **Connectivity** layer, which deals with devices discovery, network connectivity and control (QoS, Security etc.). Above this layer is the **Platform** layer, which deals with many critical functions such as device management, service discovery, messaging data management and network and service integration into operational and business systems. Next is the **Service** layer which provides the consumer application or services being offered, and all related systems that enable the applications (AI, Machine Learning, Cloud Infrastructure etc.). Below is diagram and a table that summarizes these important layers, key function and examples.

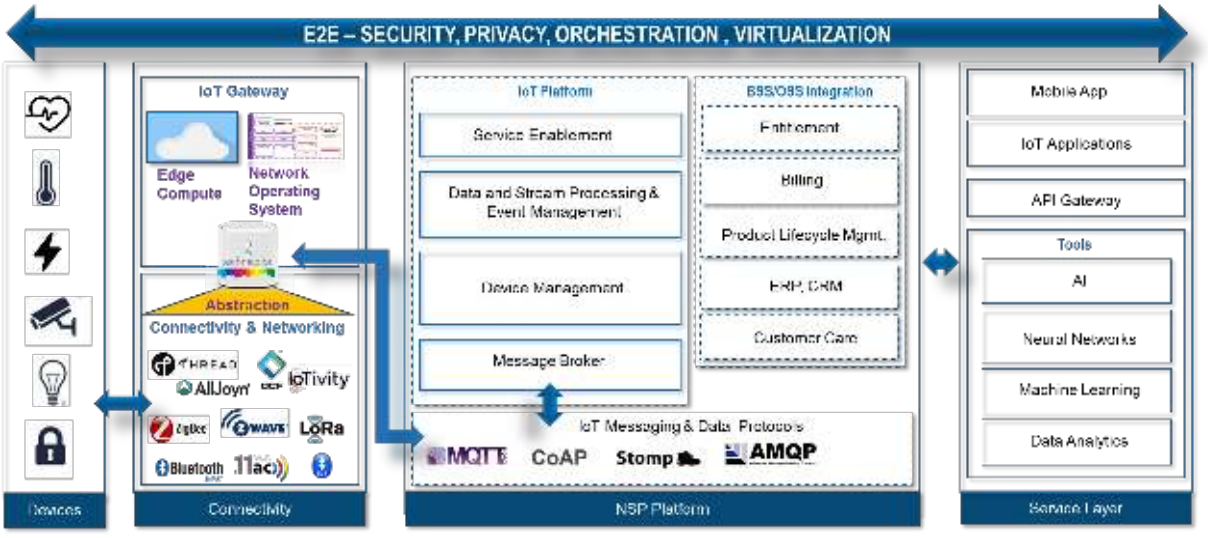

Figure 1: NSP End-to-End IoT Service Layers

1.3.1 Devices

The number and diversity of IoT sensing devices is vast and fragmented and will not be discussed in detail in this paper. However, from an NSP view, it is important to mention the complexity of managing SKUs and kits for curated IoT services will be a major challenge. There is a new opportunity to provide general-purpose sensors which eliminate the need for discrete number of sensors. Many sensors are product specific and limited in one functional area (glass break sensor, motion sensor etc.). Since many smart devices and sensors are silo managed products it makes it extremely difficult for the NSP to kit many sensors, and it is equally challenging for the consumer to manage. Additionally, many homes have devices that are not smart but provide some alerting/alarming capabilities (smoke and CO detector), and to upgrade these to Smart IoT is expensive and time consuming. Furthermore, these discrete sensors may have intelligence or awareness of other sensor devices.

Against this backdrop is a new concept for general-purpose sensors that can integrate many sensor capabilities and are low cost. These sensors can be placed in home in key locations and providing a panopticon of sensor awareness and capabilities "super sensor". A super sensor can eliminate many SKUs for NSP, provide contextual awareness with multiple sensor inputs and make non IoT device smart (e.g. non-smart Fire Alarm). The super sensor utilizes Wi-Fi connectivity to the IoT cloud application which eliminates the need for multiple IoT radios to interconnect appliances, alarms and sensors.

A recent Carnegie-Mellon research project and paper "Synthetic Sensors: Towards General-Purpose Sensing" demonstrates that a super sensor can eliminate many sensors in the home. (Gierad Laput, 2017) In the project a super sensor integrated many sensors except video to keep the cost low. The super sensor is capable of sensing; sounds, vibrations, ambient temperature, air pressure, humidity, illumination, color, motion, magnetism, EMI and RSSI.

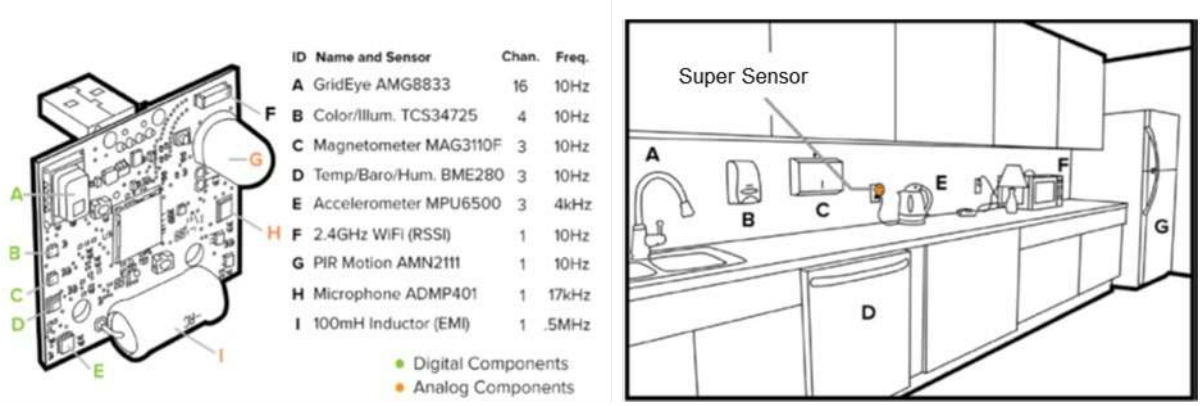

Figure 2 : Super Sensor (Gierad Laput, 2017)

This research showed that a single super sensor could provide a whole room awareness which would require 10s of devices in a single room. Furthermore, combinations of sensor inputs provide more accuracy in detecting events than purpose designed sensors and eliminated need for heterogeneous IoT protocols and frameworks. It is important to note this sensor does not cover video and IoT use cases such as remote door locks, doorbell etc.

1.3.2 Connectivity Layer

Today there exist a very fragmented and heterogeneous world of different competing in-home device connectivity solutions (Zigbee, Z-wave, Thread, etc.) that are all defined with different device constraints,

different wireless protocols and use cases in mind. These industry initiatives are solutions that are focusing on the effort of device manufacturers and standardization committees to make them interoperable in their own walled garden (a.m. Silo). The impact on the market viability for both mass market and NSP trying to package an IoT offering are very challenging. Imagine today a home with 20 smart devices with some joined to third-party IoT hubs and all these being connected to each vendor's cloud service, each with its own proprietary API. These silos are then linked to some other cloud service used by a controller like an Amazon Echo, Google Home, or smart home app. One can imagine the challenge for both consumer and NSP to manage this complexity.

The connectivity layer consists of IoT device connectivity and networking of IoT devices in the Smart Home. The IoT connectivity layer has universally moved to wireless technology. To an end user these are invisible network connections to their IoT devices in the home, and they don't want to be troubled with installing or diagnosing these connections. Many established and emerging wireless technologies are available in the market, each addressing some combination of optimization (power, performance, range, bandwidth, latency and cost). The leading established wireless networking standards for IoT are Wi-Fi, Bluetooth, Zigbee and Z-wave however, newer generation of technologies such as 6lowpan, WeMo and Thread exist. A comparison of IoT wireless connectivity standards should be viewed based on use case requirements such as power consumption, throughput, latency and range. The following diagram shows three distinct groupings based on Data Rate, Range and Power consumption.

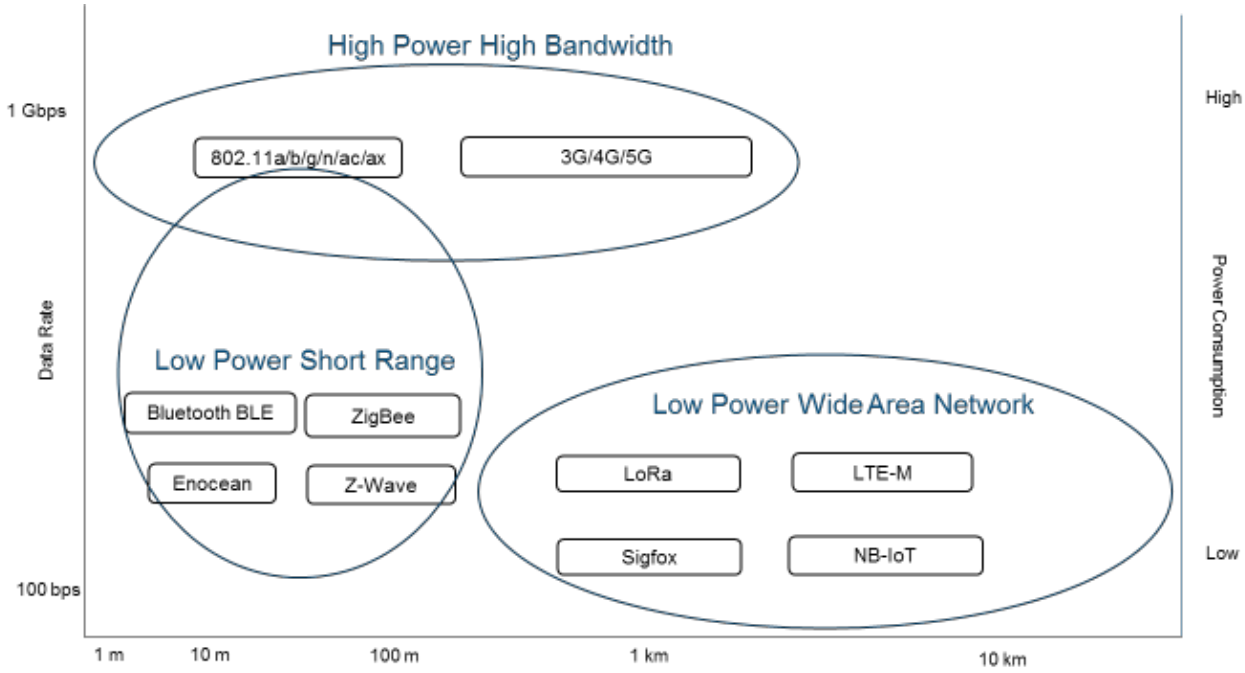

Figure 3: Wireless Connectivity for IoT

Wireless connectivity technologies can be grouped into 3 general segments. There is the low power short range technologies such as Bluetooth, ZigBee and Z-Wave. There is the lower power long-range or wide are network technologies like LoRa, Sigfox and NB-IoT. Lastly there are the high power wireless broadband protocols such as WiFi and 4G/5G, although WiFi is more of a short range wireless broadband solution.

While the new generation of transport protocols are achieving ipv6 using 6LoWPan (Thread, BLE) and are transport agnostic, most of them don't provide a clean separation between connectivity and application

specific functionality and hence need complex transformations between them to make them interoperable. The myriad of wire protocols, data formats and data models have given rise the current complex landscape in IoT connectivity. The complexity is exacerbated by the fact that many vendors create walled gardens with their cloud support functions to operate these solutions.

Smart Home IoT connectivity is mostly concerned with range of 100m or less and depending on the use case low power or high bandwidth. Therefore, the predominate technologies are Short Range. A more detailed comparison of short range connectivity options is provided in the following chart. It is important to note that range of different technologies can depend on indoor, outdoor obstructions and some are subject to interference. For example, technologies using 2.4GHz frequency band could be subject to significant radio performance degradation to do interference. Equally variable is the power consumption comparisons since different IoT devices and or use case may utilize power than others. Therefore, this chart should be view as a general range and power consumptions numbers.

Table 3: Short Range IoT Connectivity Comparison

Feature	**Wi-Fi**	**Bluetooth**	**ZigBee** (Alliance, n.d.)	**Z-wave**	**Enocean**	**Thread**	**6lowPan**
Open	Yes IEEE 802.11	IEEE 802.15.1	Yes IEEE 802.15.4	No, Proprietary based on IEEE 802.15.4	Yes, Enocean Alliance.	Uses IEEE 802.15.4, IETF 6LowPan	Yes, IETF RFC 6282, uses 802.15.4
Range	100-150 feet	v4 = 300 feet v5 = 600 feet	30-100 feet	50-100 feet	100-300 feet	100 feet	100 feet
Frequency	2.4/5 GHz	2.4GHz	2.4GHz	908/915 MHz	315, 868 MHz and 2.4 GHz	2.4 GHz	2.4 GHz
Data Rate	300 – 1300 Mb/s	v4 = 1 Mbps v5 = 2 Mbps	40-250 Kbps	9.6-100 Kbps	125 Kbps	250 kbps	250 kbps
No. Devices	Router dependent	7	65,000	232		250-300	250-300
Topology	Star	P-to-P Mesh	Mesh	Mesh		Mesh	Mesh
Hub Required	No	Yes	Yes	Yes		Yes	No
Security	WPA2	AES-CMAC encryption ECDHE (Elliptic Curve Diffie-Hellman)	AES-128 symmetric encryption	AES-128 symmetric encryption	AES-CBC and variable AES (VAES)		
Power Consumption	High Power	Low Power	Low Power	Low Power	No Power	Low Power	
Cost							
Good For	High bandwidth Mid-range	Short Range PAN and LAN	Lower Power Short Range	Low Power Medium Range			

It is evident in this evolutionary phase of IoT that the primary focus of solutions and technology are more driven by IoT use case and user experience versus universal interoperability. Therefore, several connectivity technologies will exist for the foreseeable future. The NSP will need to make key decisions on which combination of connectivity technologies they need to deploy if they want to add value in this layer. Another factor is cost of technology that impact device costs to consumer or network related costs.

Market penetration of various wireless connectivity technologies among sensor and appliance equipment providers varies significantly. Wi-Fi and Bluetooth dominate home due to their maturity and pervasiveness, others depend on maturity and vendor solution support. It is hard to find good analysis of which wireless technologies dominate the Smart Home beyond Wi-Fi and Bluetooth, but Z-wave appears to have the most significant ecosystem support followed by ZigBee, and then a large gap in terms of adoption exist with new generation technologies such as Thread, NB-IoT and so on. We believe that Wi-Fi will dominate applications that do not require low power or some other constraint. In the low power solution area there is a lot of competition with Z-wave having a largest ecosystem. It is worth noting that Z-wave which was developed by Sigma Designs has been acquired by Silicon Labs (here). This could widen the adoption of Z-wave since Silicon Labs is a major player in IoT chips. More than 2,400 certified, interoperable Z-Wave devices are available from the Z-Wave Alliance of more than 700 manufacturers and service providers worldwide.

The official Bluetooth marketing material from the Bluetooth standard organization advertises that Bluetooth 5.0 has four times the range, two times the speed, and eight times the broadcasting message capacity of older versions of Bluetooth. Again, these improvements apply to Bluetooth Low Energy, ensuring devices can take advantage of them while saving power.

With Bluetooth 5.0, devices can use data transfer speeds of up to 2 Mbps, which is double what Bluetooth 4.2 supports. Devices can also communicate over distances of up to 800 feet (or 240 meters), which is four times the 200 feet (or 60 meters) allowed by Bluetooth 4.2. However, walls and other obstacles will weaken the signal, as they do with Wi-Fi.

Looking at their functionalities, following picture shows the overlap of these connectivity stacks when applied to OSI model.

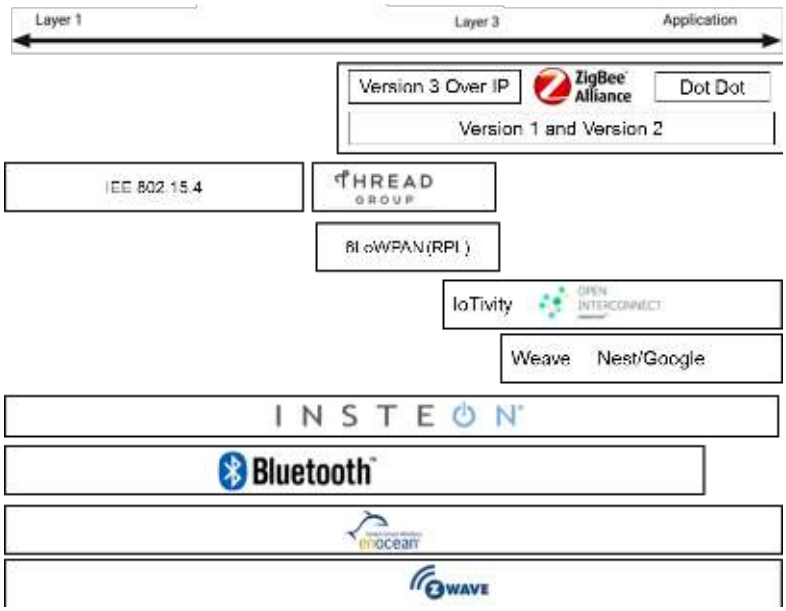

Figure 4: Connectivity Stack Comparison

Albeit these connectivity solutions will continue to exist and evolve within their silo, there is a clear need to come up with an "abstraction layer" that is protocol agnostic and provides an industry standard Data Model to enable those silos to interact with each other, and this is analogous to traditional IT normalization initiative in data models and applications. Having a normalize and coherent model to allow the creation of IoT applications that are agnostic to the underling protocols and frameworks within the Smart Home. Like the efforts around "Semantic Web & Technologies" standardization in the last few years, IoT could leverage similar concepts to further improve its capacity to understand things' data and facilitate their interoperability.

There is a clear trend of moving the network layer to an all IP layer based on 6LowPan. Thread is a good example of that and device constraints of the latest version of Zigbee followed that trend with Zigbee/IP. This opens possibilities for easier integration with this layer in container technology, since the network stack is typically embedded in the firmware seen the close relation with the hardware drivers.

1.3.2.1 Messaging Protocols

There are many messaging protocols standards each with advantages and disadvantages. There is no panacea protocol to address a universal solution for IoT, however we are seeing some of these protocols dominate specific use cases. The IoT Platform will need to flexibly support multiple protocols to address many use cases. Below is a comparison of the messaging protocols being used by IoT solutions providers.

Messaging protocols are unique from other protocols since IoT device are resource constrained, are data centric, always on and have security considerations like getting through firewalls. Messaging protocols differ on several criteria such as communication model, message size, syntax and QoS mechanisms. The Hyper Text Transport Protocol (HTTP) which the most widely use internet protocol is a Request/Response protocol versus a more model data centric protocol such as Message Queue Telemetry Transport Protocol (MQTT). (R. Fielding, 2014) More recent introductions of HTTP-like protocols such as Constrained Application Protocol (CoAP) constrained nodes and networks are still request response but address size.

Table 4 : Messaging Protocols Comparison

	MQTT (OASIS, 2015)	HTTP/S (R. Fielding, 2014)	CoAP (C. Bormann Universitaet Bremen TZI, 2018) (Z. Shelby of ARM, 2014)	AQMP	XMPP	STOMP
Standard	IISO/OASIS	IETF	IETF rfc8323	IETF rcf2119	IETF rfc6120	
IP Type	TCP-based	TCP-based	UDP-based	TCP-based	TCP-based	TCP-based
Message Type	Publish/Subscribe	Request/Response	Request/Response	Transactional	Transactional/PubSub	Request/Receipt
Syntax	Simple Noun/Verb?	Verbs/Status Codes	HTTP Like		HTTP like	HTTP like
Size	Small: 2 Bytes	Large: ASCII	ASCII	8 Byte Header, Variable Ext Header and Variable Frame Body.	Verbose, XML	1KB - 10KB
QoS	3 levels	No mechanism	Confirmable requests			No
Reliability	Avoids packet loss on client disconnect via keep alive	No mechanism	"Observer", "Response back"			No
Security	SSL/TLS, user/password in connect message	SSL/TLS	Datagram TLS	TLS/SASL	SSL/TLS	SSL
Other	MQTT 5 - Enhancements		rfc7252 Resource discovery			
Good For	Cloud Scale & Small Footprint	Non-event based applications	Improves simplicity of HTTP		Messaging, presence detection, signaling plane	

Conclusion there is no solution that fits all and thus as an NSP must deal with heterogeneous network of IoT protocols. However, effort should be made to minimize the number of protocols in the network.

1.3.2.2 Message Broker/Gateway Function

A key function is the message broker/gateway in the platform layer for the Service Provider. This function needs to support multi-protocol messaging interfaces, message load balancing, high-availability and management. A few open source message brokers are available in the market (RabbitMQ) but the Carrier Grade requirements for this critical function will require optimized and hardened platforms.

Messaging infrastructure which typically provides publish/subscribe semantics have been used for a long time and complement the more traditional Request/Response semantics of the current internet

communication paradigm. They have been used to decouple monolithic solutions towards a more decouple system that is easier to evolve. Additionally, these networks are always on and require real-time capabilities to enable a more event driven architecture.

With the advent of the cloud a new generation of cloud-native solutions have emerged for IoT which hence are more capable of scaling to a global infrastructure, while still providing resilience, High availability and guaranteed message delivery.

These new cloud-based solutions have followed the same evolution as their database counterparts moving from SQL to NoSQL solutions, but are therefore also lacking some maturity and have been simplified in terms of requirements in favor of their scalability requirements.

Typically, IoT platforms support different transport protocols like MQTT, Websockets, or proprietary protocols, which are chosen because there are lightweight and hence can easily be integrated in resource constrained devices. Also, the CoAP protocol starts to follow this trend with the recent extension of the protocol towards pub/sub capabilities, but this is still in a very early stage.

Today's solutions are using this infrastructure layer as a control plane for use cases such as command/control, notifications at large scale, configuration management, presence detection and even for the signaling plane for communication protocols like WebRTC. They typically also providing the necessary adaptors to other systems as queuing systems, streaming systems and rule engines.

Today all major cloud providers are offering this messaging services as part of their IoT offering, with the caveat that they provide out of the box integration with their own solutions and therefore create a risk of locking in their ecosystem. Some other companies have focused on providing alternative solutions in this space, even with capabilities of running the solution on premise. PubNub is a notable example of that evolution.

In short, careful consideration need to be made in the selection of these technology, with respect of locking-in, cost, global availability, size of the ecosystem and developer's community, protocols supported, adaptors, and the messaging semantics they provide.

1.3.2.3 IoT Device Management

An IoT Platform is the lowest layer at which the IoT devices connected to the system can be viewed and managed on a system-wide basis. This makes the platform the ideal place to manage those devices. Device management activities can be classified into three distinct groups:

- Hardware Management
  - Inventory
- Software and Configuration Management
  - Device Modeling
  - Authentication of cloud/backends
  - Registration
  - Entitlement
  - Software upgrades, OTA updates
  - Configuration management: determination, verification, backup, reset
  - Policy creation and application
  - Off-boarding and device retirement
- Monitoring

- Centralized log collection and management
- Fault tolerance, failing safely
- Issue alerting
- Troubleshooting, diagnostics and remote reboot

Much of IoT Device Management overlaps with broadband device management that NSPs excel at. There are two areas in which IoT Device Management may expand on existing broadband device management functions in an NSP:

- Scale: It is likely the number of connected devices and sensors across the customer base has or soon will exceed the number of broadband gateways in the NSP network. This will require additional scaling and automation beyond what exists today.
- Device Modeling and Offline Representations: The requirement to support smaller embedded devices, devices with lower power, and intermittent internet connectivity will force IoT management platforms to seamlessly manage devices whether those devices are online at the time of a change or management action. Leading platforms are supporting this through a cloud-resident abstraction of a remote physical device. These abstractions are commonly called device shadows or device twins, which allow the IoT device management platform to execute changes and actions that are cached centrally until the device becomes online.

1.3.2.4 BSS/OSS Integration

Most NSPs have OSS/BSS systems that have evolved over decades. In such a brownfield environment, it is critical that any new service introduced by the NSP (e.g. IOT service in this case), integrates seamlessly into the NSPs current OSS/BSS systems.

For OSS integration, NSPs have mediation platforms that act as brokers for OSS integration. The mediation platforms map the operational workflows via a standardized interface within the NSP domain to provide configuration and provisioning of customer data, operational parameters /bounds for the service, health monitoring and analytics associated with the service. Typical integration technologies used are SOAP/XML, RMI/RPC ORB, EAI/CORBA.

For BSS integration, NSPs have customer management systems that provide billing & charging, NSP CRM systems (Product Catalog, Order Management, service order fulfillment etc.) as well as end user presentation systems e.g. customer account management portal, service management portal as well as a set of associated NSP branded mobile apps. Like OSS integration, the BSS integration on the backend is also done via mediation platforms that use integration technologies like SOAP/XML and RPC. However, the customer presentation systems are usually bespoke to the service being offered e.g. a set of mobile apps dedicated to the service to provide customized UX for the service.

1.3.3 Service Layer

The service layer deals with the end user application and service offering that is consumed by the end user. It deals with all the service logic, intelligence (ML, AI, Analytics) application interaction and services experience by the end consumer. The service layer deals with presentation layer (portal or mobile app) to the end customer and manages all the service logic for the end user or IoT devices. This layer must integrate with service activation, billing and customer care BSS systems in the NSP network.

Service or application execution environment have been virtualized and thus can run in Cloud in a centralized model or distributed to Edge Compute.

1.3.3.1 Edge Compute

Many challenges have emerged as the number of IoT solutions in the Smart Home market grow. Privacy, latency, bandwidth constraints, and reliability, among others, present challenges that cannot easily be overcome in cloud-only models.

Edge compute is a term that generally refers to the ability to perform enhanced or additional processing in the CPE. This processing uses higher-performance CPUs and additional RAM and may utilize virtualization or containerization technologies or may take place directly on the existing device operating system. Because processing via edge compute takes place in close-proximity and well-connected to the consumer, it is an attractive and useful tool to combat the challenges created by many Smart Home service offerings. For example, processing data in the consumer's home without sending data to the cloud is more private and less reliant on fast and reliable internet connectivity.

Privacy – which is only one of the concerns reduced by successfully leveraging edge compute – has become a major concern for consumers in IoT markets like US and Europe. Regionally, European markets have instituted greater regulatory protections over privacy than the US. This is best exemplified by the implementation of the General Data Protection Regulation (GDPR) which takes effect May 2018. The GDPR significantly changes how companies handle EU citizen data privacy. GDPR places requirements for managing EU citizen data, such as a 72-hour notice to citizens after first having become aware of a data breach. Other aspects of the regulation require data erasure, data portability, and reporting. In general, personal data about identity, habits, speech and video will become a growing concern. Edge compute can enable new service architectures where personal data is processed locally to minimize the exposure of personal data.

Edge compute can also maintain service operation during a network failure. On-device, service-specific processing that is enabled by edge compute can act as a buffer during a network failure and then synchronize data and state with cloud-based processes when internet connectivity is restored. Some sensor applications have additional redundancy requirements that may include using a battery backup to operate during a power outage. Edge compute, coupled with a battery backup, creates a robust platform for offline data processing.

Edge compute is also a useful tool to service providers looking to reduce operating costs by shifting processing away from expensive cloud providers. Utilizing edge compute to relocate data processing from data centers into consumer's homes decreases the cost associated with cloud provider compute resource consumption.

Over the last two years we have witnessed the shift towards edge computing to complement the first generations of cloud centralized IoT solutions, mainly for cost and latency sensitive solutions and allowing for autonomous intelligence at the edge in absence of internet connectivity. The concept of intelligence at the edge is not new. Similar concepts include mobile cloud computing (MCC), mobile edge computing (MEC), mist computing, and cloudlets (fog nodes). Although the edge ranges from private cloud, to NSP core network and the to endpoints, they contain similar architectural building blocks that designed for their respective resource environments. In the context of Edge Compute for IoT there will not be one solution fits all, but the true challenge is to provide a Network Operating System that can support an application execution environment to run IoT application and services on the Edge of the

Network. Fog compute can be confused with edge computing, but they are different. For example, Fog Node may be a service that runs on Edge Compute on the Homeware CPE.

One of the most important elements of this shift is the focus on offering a developer friendly environment for NSP that enables self-service and e2e control of a software workloads using container technologies. Containerization enables a leap forward in productivity with a modern workflow, and cloud-like agility to offer a new service, collecting user feedback and a fail fast attitude that eventually will disrupt the current status quo w.r.t. Embedded development.

1.3.3.2 Containers for Embedded Implementations

The introduction of containers in the open source community have their origin in Linux development projects. Linux began providing containers (LXC) in Release 4.x and OpenWrt projects began around 2015 (Rel. 15.x) for both LXC and Docker. Docker is a superset of LXC and will be discussed below in more detail. Containers are self-contained execution environment with their own, isolated CPU, memory, block I/O, and network resource which are share the kernel of the host operating system. This is different from virtual machines which are running many duplicate instances of the same OS and thus heavier weight. Another form of virtualization is process containers a.k.a serverless programming.

In this case a runtime binding is mapped into a container that offers an execution environment that contains the basic set of libraries necessary for a specific language (Python, JavaScript, Java, Go, etc). This offers a more lightweight option w.r.t. the container size, but creates some additional language specific dependency management problems. The key advantage here is that the need to manage all the security aspects and operational burden is taken away from the internal container environment and hence the name serverless was coined.

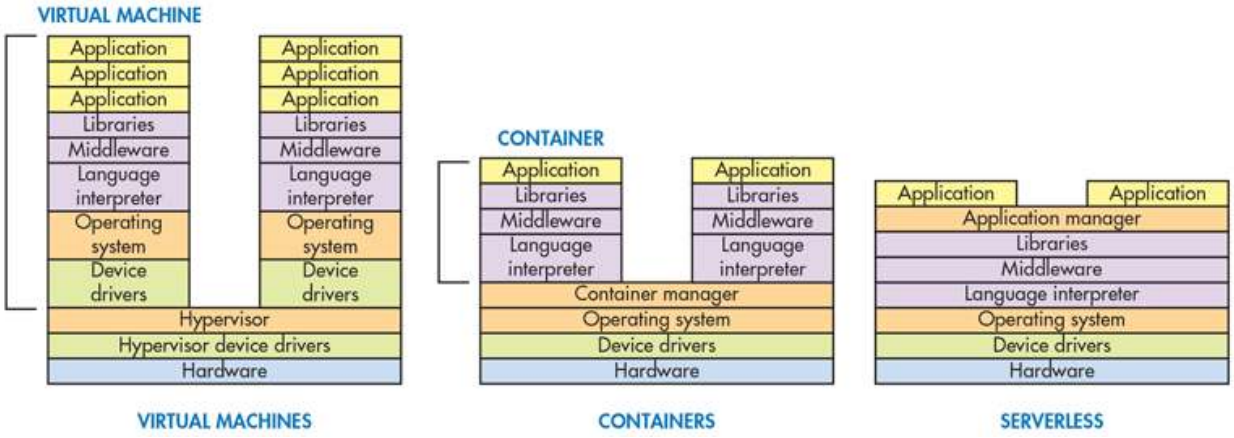

Figure 5:Virtualization of IOT services

The Linux Container (LXC) features and capabilities continue to evolve for embedded systems with limited resources and different IoT use cases. Containerization and light weight sandboxing tools are available in open source to develop a framework for secure application execution environments. This is particularly important in residential or connected home CPE, Wi-Fi APs and Extenders offered by service providers.

Orchestration and Portability of containerized applications is an important design requirement for Service Providers to move applications across different device hardware and resource capabilities. Running containerized or serverless application on customer premise equipment that the NSP curates will be a

challenge but represents and innovation that may allow them to have better service delivery than OTT solutions competing for the same consumer in the Smart Home market.

1.3.3.3 API Gateway Function

As discussed in earlier sections, many (if not all) Smart Home IoT solutions have leveraged the Application Programming Interfaces (API) based on Representational State Transfer (REST) style resource-oriented architecture (ROA). REST APIs allows application integration between IoT product silos and/or 3rd part application development and automation between these products or to integrate into other systems (eg. (backend DB, Analytics, BSS/OSS etc.). In addition to REST API which leverages HTTP protocol, an increasing number of IoT solution can support IoT protocols (MQTT, etc.,) over Websockets. This shows that APIs are critical aspect of the Service Layer and NSP need to understand key aspects of API Management, Security, Privacy and Consumer Experience.

1.3.3.4 IOT visualization

Due to large number of datasets available with IOT application, the task of aggregating and rendering the datasets to the end user in an intuitive way is key for good user experience (UX). As an example, the figure below shows a smart farm IOT application that has LoRA based wireless IOT sensors that gather all the farm sensor data and aggregate it in a useful way that it can be layered upon a real time video feed and rendered to the end user. The screenshot below shows the temperature and humidity in the grain silos, soil humidity as well as health related information on the farm animals.

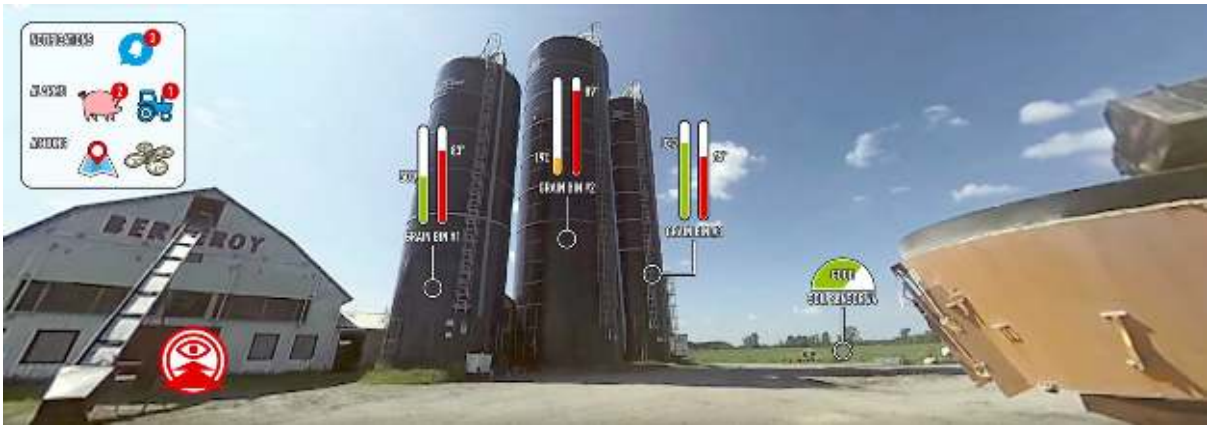

Figure 6 : IOT smart farm Augumented Reality

1.4 IoT Architecture Approaches

IoT architectures have traditionally been public or private cloud based, where all the IoT devices sent actionable telemetry directly to a virtualized cloud IoT gateway or via a physical IoT gateway on-prem. IoT gateways aggregate all the IoT events and feed it to an event processor which would correlate the events to actions and triggers. Triggers might be to send dynamic control messages to IoT devices and/or to notify the end subscriber of the events. Such a traditional architecture is shown below:

Tenets of a such an architecture as shown above are:

- IoT devices at premise

- Broadband gateways at premise that support Wi-Fi and other IoT specific low power 802.15 radios (Zigbee, BLE, Z-Wave, etc.)
- Virtualized cloud gateway (Public or Private Cloud)
- Stream Event processors: To process all the incoming IoT telemetry
- Scalable databases: To manage IoT device identities and states.
- Control systems: To analyze incoming IoT telemetry, map it business logic, and send control signals to actuators in IoT devices if needed
- Analytics: Create Actionable insights from IoT data and state for OSS and/or end user presentation.
- Subscriber Portal / Mobile UX: Cloud based portals and mobile apps to provide a GUI to end user to interact with the subscribed IoT services and manage notifications

However, in the last 18-24 months, users have been lot more concerned about privacy and security related to IoT services. Interaction with NSPs echoes those concerns as well. In recent discussions with a leading US MSO, this topic was front and center for their IoT platform requirements. This has led to evolution of a hybrid architecture that allows for user data to kept private and on premise while doing other non-sensitive data processing in the cloud.

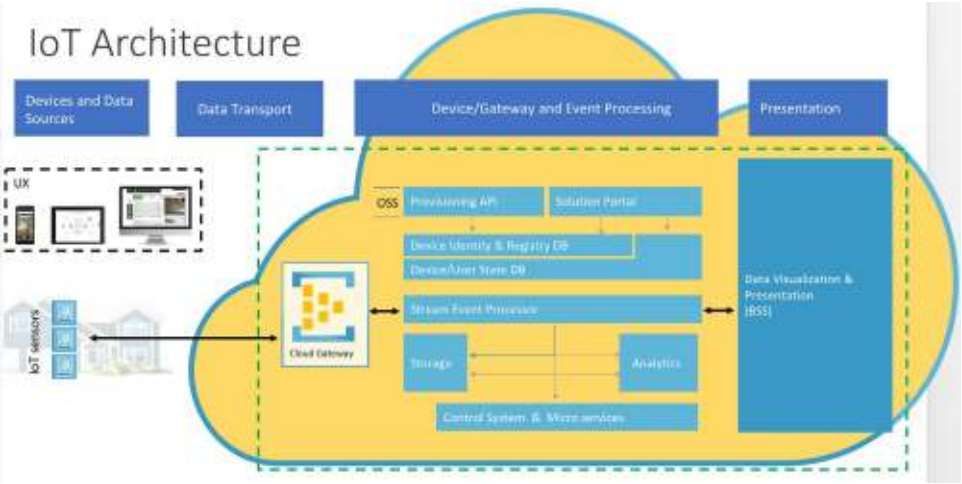

Figure 7: IoT Platform Architecture – Cloud Approach

Such an architecture uses a Broadband/IoT Gateway device at premise that has edge-compute capabilities to store privacy sensitive user data (telemetry) as well as be able to do local processing of such IoT telemetry. Once local processing is performed, only anonymized triggers are sent to the cloud backend for further processing. Based on the intent of the service, IoT user data may or may not be sent to the cloud. As an example, a 'peace of mind' security service could have business logic where a motion sensor at home triggers a video capture device (IP camera) on motion detection. Then the video clip is sent to the user's mobile app as a notification; for user privacy the captured video clip is sent directly from the edge-cloud capable IoT gateway to user's mobile app using a secure IP transport from Hybrid GW to the mobile app. Video clips are never stored in the cloud backend thus mitigating any privacy related fears (as well as addressing regulation like GDPR) related to cloud based storage.

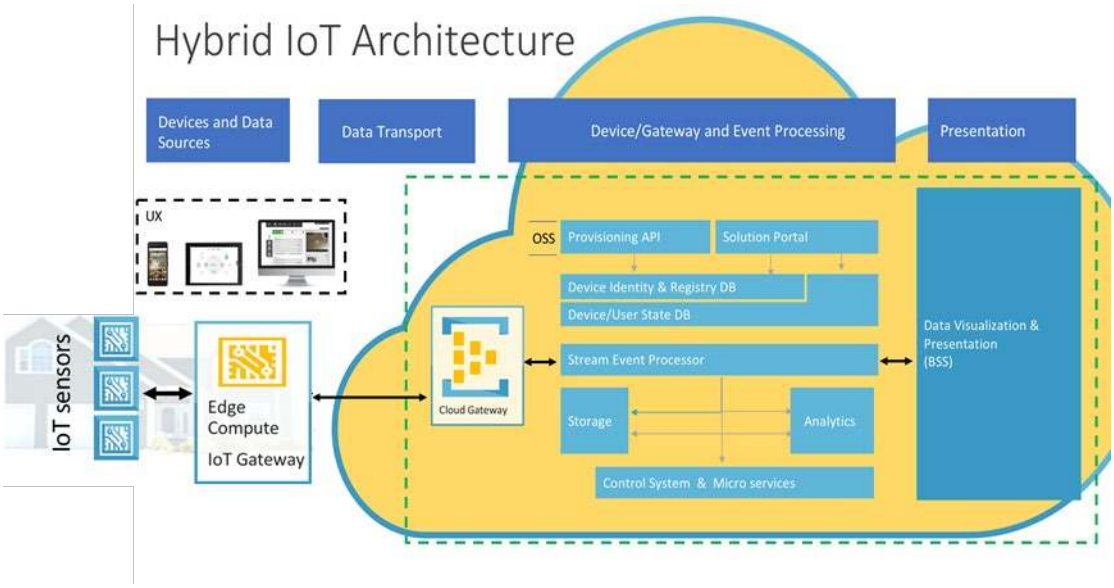

Figure 8: IoT Platform Architecture – Hybrid Approach

As shown above, the hybrid architecture utilizes an edge-compute capable Gateway on premise that is capable of localized processing of IoT events, thereby keeping user data and IoT telemetry local and private on premise. Typical Edge-Compute capable Gateways have Dual core or Quad core low power processors as well as local storage (Flash and RAM) that is more than traditional broadband gateways.

1.4.1.1 IoT Gateway Function

IoT hubs such as Logitech Harmony Hub, Samsung SmartThing Hub and others are focusing on integration at the connectivity layer to provide customers a "universal remote" experience. In addition to the capabilities of bridging a myriad of connectivity protocols, the industry has recognized the importance of a local execution environment on these Hub's referred to as Edge Compute. Edge Compute addresses several challenges related to latency, network bandwidth, reliability and security, which cannot be addressed in cloud-only models. Containerized execution environment running on Edge Compute provides agility in rapid development and deployment of new capabilities, driving cloud cost down and providing more autonomous operations of these devices without having to rely constantly on the availability of cloud services. Although this was the initial promise of OSGI, containerization and virtualization are providing a more flexible and secure alternative, and in this respect, we see quite some movement in this space with offerings from Amazon, Microsoft (Lambda's and Functions) or fully dockerized environments like what resin.io is providing.

Table 5: IoT Gateway Stack

IoT Gateway Functions	
Messaging & Data Management (MQTT, HTTP etc.,)	Network Management
Connectivity & Inter-Networking (ZigBee, Z-wave, Wi-Fi)	
Edge Compute (LXC, Docker, Serverless Programming)	
OS/RTOS (Linux, .NET)	

Being on the demarcation point between the NSP's network and the home network, we believe that a residential GW is at an ideal place to tap into Edge Compute. Edge Compute processing giving NSP's and end-to-end capability to balance network functionality and in-home service and being able to finally break the current IoT siloed solution challenge. It gives them the opportunity to improve security, home networking and fundamentally change the user experience in an agile way. For more details on Edge Compute go to section 2.2.8 Service Layer.

IoT gateways that are curated by the NSP can provide a single point of Wireless IoT connectivity interfaces, device and data management and edge compute. With edge compute resources core network functions such as the API Gateway could have a local instance and sync with core network when needed. This would address important privacy and resiliency requirements for IoT services. We have already seen the emergence of IoT API layer moved into the local devices execution environment. For example, Amazon allows you to develop application in the AWS Greengrass environment but allow it to execute local. Azure and Google are all moving to this local compute environment.

1.4.2 Platform Layer

The Platform layer deals with all the important management layers to provision, network, scale and manage IoT services as an integrated service offering. NSPs contemplating ownership of all or pieces of this layer require careful platform design of key integration points into existing provisioning, operational and business systems. The introduction of new a IoT device management framework and potential integration will be core function. Across these layers NSPs will have to consider how each layer meets security and privacy requirements, and how to facilitate by network infrastructure at scale. At the top of the stack important consideration of how the platform interacts with services and applications that are 3rd party or organic needs to be determined. The complexity of the Platform layer requires NSP's to look at vendor solutions that address these function areas, and different deployment models. The different layers of the IoT Platform from and end-to-end perspective is presented below and detailed in following sections.

Table 6: IoT platform layers

		IoT Platform Functions
Security & Privacy	Infra. Compute, Storage, Processing	**Service Enablement** Set of well-defined APIs, interfaces and tools between management layer and services layer.
		BSS/OSS Integration Run business logic for IoT service, operational integration, visualization, customer care etc.
		Entitlement Manage different services and policies including billing, bundles etc. This layer interacts with service catalogue and other BSS functions.
		Data Processing & Event Management Process data real-time via rule engine, events and notifications
		IoT Device Management Device management (on-boarding, monitoring, updating, replacing), Backup.
		IoT Messaging Management Message Broker/Gateway, Messaging, Queueing, Security…

As a point of reference, according to IoT Analytics (here) there are +450 IoT platforms on the market, of which 32% focus on industrial applications. IoT Analytics also states that "We believe, only 7% of the 450 IoT Platform companies generated revenues more than $10M with their IoT Platforms in 2016. Furthermore, more than half of all companies made less than $1M, most of them smaller startups. The firms leading the pack are mainly made up of large cloud players, legacy device management and connectivity backend platforms as well as a handful of heavily backed Silicon Valley startups that are scaling faster than most of their counterparts around the world."

1.4.2.1 Messaging Management

This layer interacts with IoT devices and is typically referred to as a Messaging Broker or Gateway. The messaging must be securely transported from IoT device to the management layer, must be scalable messaging protocol to support millions of devices, must support maintain resiliency and integrity of the

messages, and must intelligently distribute message to the correct processes in the management core layer.

1.4.3 Cloud IoT Platform

Amazon, Microsoft, Google and IBM all have IoT infrastructure and services that are similar but with some key differences. They all are essentially offering their Platform-as-a-Service. For the most part these vendors focus on infrastructure and service to facilitate IoT applications and leverage other cloud services (compute, storage, analytics etc.) they offer. They are largely cloud-based platforms, but they do provide device level components on premise with local processing (e.g. AWS Greengrass, Azure IoT Edge). AWS and Google are most aggressive in enabling IoT platforms to complement their smart home devices push (Google Home, Alexa Dot), but AWS and Azure are more mature than Google. IBM is more focus on Artificial Intelligence (AI) as a service, and Microsoft Azure has refocused its efforts on telemetry and data collection of IoT devices. Despite the big 4 developing broad end-to-end approach and aggressive roadmaps, the market and technology is still in early phase of maturity. Both Amazon and Microsoft did not release their IoT platforms until late 2015, and Google's was made available in 2017 and is still in Beta.

Table 7: Infrastructure-as-a-Service Cloud IoT Frameworks

	Edge Products	IoT Core Product	IoT Core Services	Pricing (US)
AWS	IoT Device SDK Greengrass – Local Lambda (edge compute) Snowball – Local storage	IoT Core	Connectivity: Message Broker Device Management: State - Device Shadow Registry Rules Engine IoT Analytics -- filters, transforms, add meta-data	Connectivity: • $0.080 per million minutes of connections Messages monthly message volume: • Up to 1 billion messages $1.00 • Next 4 billion messages $0.80 • Over 5 billion messages $0.70 Device Shadow and Registry: • $1.25 per million operations Rules Engine: • Rules per million triggered $0.15 • Actions per million executed $0.15 *AWS Free Tiers (see AWS details below)
Azure	IoT Device SDK IoT Edge – local processing of Azure modules (edge compute)	IoT Hub	Connectivity: Message Broker Device Management: State - Device Twin Registry Provisioning Monitoring - IoT Suite Maintenance – IoT Suite	IoT Hub (all messages metered in 4KB blocks, max message 256KB: • S1 tier: 400,000 messages per day per IoT Hub $50 /mo. • S2 tier: 6 Million messages per day per IoT Hub $500/mo. • S3 tier: 300 Million messages per day per IoT Hub $5,000/mo. IoT Device Provisioning: • S1 tier: General Availability Price: $0.10 per 1,000 operations
Google	Google Cloud MQTT Client Opt. Brillo/Weave *	Cloud IoT Core	Connectivity: MQTT/HTTP Bridge Device Management	Don't allow flexibility with MQTT to address scale.

*Brillo became Android things, and weave got abandoned in favor of Nest weave which is a total different protocol.

**

A functional and feature comparison of the big 3 IoT frameworks is provided below. As you can see in the table AWS has the most comprehensive IoT offering, but there are many similarities between AWS, Azure and Google.

Table 8: Cloud IoT Framework Comparison

	AWS	Azure	Google
Client SDK / Language	Android-Mobile, Arduino, Embedded C, C++ , iOS-Mobile, Java, JavaScript, Python	.NET, Embedded C, Java, Node.js, Python	GCP - Java, Python, NodeJS, Ruby, Go, .NET, and PHP
Messaging Protocols	MQTT, HTTP, WebSockets	MQTT, AMQP, HTTP	MQTT and HTTP 1.1 (not 2.0)
Security Transport	TLS	TLS	TLS 1.2
Authentication	Per-device with SAS token	X.509 certificate client authentication, IAM Service, Cognito Service	Per-device public/private key (asymmetric) device authentication and JSON Web Tokens (JWTs RFC 7519)
Device Management	Registration Configuration State	Registration Provisioning State Monitoring & Maintenance	Registration Provisioning State Monitoring
Edge Compute & Services	Greengrass -	IoT Edge – Stream Analytics, Machine Learning, Azure Functions (custom code)	No
Data Ingestion & Processing	Kinesis	Event Hub	Cloud Pub/Sub
Stream Event Processing	Kinesis analytics	Stream Analytics	Cloud Data Flow
Data Storage (DB)	S3 DynamoDB RDS	Azure Blob Storage Azure Cosmos DB Azure SQL DB	Cloud Storage (object store) Cloud Bigtable BigQuery
Data Visualization	QuickSigth	PowerBI	Cloud Datalab/Data Studio
Analytics	Iot Analytics	HDInsigth	Cloud Analytics
Machine Learning	Sagemaker	Azure ML	CloudML
Notifications & Alerts	SNS	Azure Notification Hubs	Firebase Cloud Messagin

1.5 Harmonization of Standards

1.5.1 Connectivity Harmonization

Currently there are different opinions on how to tackle the connectivity challenge. Some are advocating, open data models, or a new generic opensource protocol, while we see also movements from vendors like

Amazon that is promoting de-facto API's like what happened in the cloud space for most of these services.

At connectivity layer there are several alliance/stands initiative to create common standards. Examples Alljoyn and OCF, you have half the industry standards. Wireless standards are also coordinating common standards. OCF/Alljoyn is focused on the lower layer abstraction…

At layer 3 we see consolidation around IP vs. proprietary.

At the application layer, silos remain a main issue among alliance groups. There remains a gap in the standardization of the application layer. For example, when Alljoyn was absorbed by IoTivity, They where directly competing with the established standards like Zigbee, Zwave and bluethooth, and despite all the efforts they have still not established a major footprint of compatible devices. Therefore, an abstraction layer will still be required in the architecture to stich these APIs together.

1.5.2 Data Model Harmonization

The IPSO Alliance which is now part of Open Mobile Alliance (OMA) has been one of the first standardization committee's, identifying the problem around interoperability issues with the different connectivity protocols and has originally started to create a cross industry attempt to harmonize the different data models in different industries into a more generic set of data models.

The IPSO Alliance is actively developing an entirely new approach to resolve this data representation and scalability issue. They call it the Node Metal Model. It defines a unique method that allows smart objects to interoperate with each other.

This new meta model is the only known approach that universally sets out how all things should be defined, so that each specific thing, including its objects and resources, no longer needs to be predefined and preregistered.

1.5.3 Service Layer Harmonization

At the service layer, there are standardization and best common practice initiatives to develop harmonization of IoT service layer APIs. This is analogous to traditional IT normalization initiative in data models and applications. Having a normalized and coherent model to allow the creation of IoT applications that are agnostic to the underling protocols and frameworks within the Smart Home. Like the efforts around "Semantic Web & Technologies" standardization in the last few years, IoT could leverage similar concepts to further improve its capacity to understand things' data and facilitate their interoperability and device constraints

By enhancing these standards with an abstraction layer could be leverage by Semantic Web of Things and the opening the different device ecosystems at the application layer with API's. In this framework our IoT enabled products could provide the necessary bridging capabilities between the "old world" design methodologies that guarantee standardized interop at the device communication layer and the application and services layer which is e2e. This allows an agile introduction of new functionality focused on consumer interaction and experience. This effort will only be successful with the right partnerships and the creation of the right business incentives to be able to finally unlock the business potential of the future smart home.

Conclusion

This paper covers an end-to-end view of an IoT architecture from a NSP perspective. Depending on what layer the NSP wants to own or provide value will determine the importance of conclusions drawn from this paper. The most ambitious of NSPs will want to own up to the platform layer, and to enable their own and third party services to ride on top of that platform.

In order to achieve that objective from a device layer perspective, the key is to minimize the number of SKUs to be handled and kitting to be done for curated sensor solutions. An example was given in section 1.3.1 of the synthetic "super sensor" work done at Carnegie Mellon, that could fulfill this minimization objective.

From a connectivity layer perspective, the NSP must own/drive the requirements for the IoT hub in the smart home because this is a key interworking and management point of presence in the home. The IoT hub must support multiprotocol connectivity, messaging and framework capable and have enough resources and a network operating system that can support containers. Highly coupled to the device layer, the ability to be conservative on the device type needs will allow the NSP to be focused and concrete on IoT radio requirements in the gateway or hub.

It is at the platform layer that the most critical work must be done, in order for an NSP to realize the objective of creating an open and inviting service environment for consumers and third parties. Along with a significant commitment in time and resources to realize the objective, historic tools such as TR-069 must be abandoned in favor of a careful selection of modern orchestration and device management methods which are evolving in the could native DevOps community.

Abbreviations

API	Application Programming Interface
NSP	Network Service Provider
IoT	Internet of Things
CAGR	Compound Annual Growth Rate
OTT	Over The Top
EMI	Electromagnetic Interference
RSSI	Received Signal Strength Indication
BLE	Bluetooth Low Energy
LoRA	Long Range Wireless
OTA	Over The Air
CPE	Customer Premise Equipment
GDPR	General Data Protection Regulation
LXC	Linux Containers
MQTT	Message Queuing Telemetry Transport
OCF	Open Container Forum
OMA	Open Mobile Alliance

Bibliography & References

Alliance, Z. (n.d.). *http://www.zigbee.org/*. Retrieved from Zigbee Alliance.

C. Bormann Universitaet Bremen TZI, S. L. (2018). *CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets.* Internet Engineering Task Force (IETF).

Gierad Laput, Y. Z. (2017). *Synthetic Sensors: Towards General-Purpose Sensing.* Pittsburgh, PA: Human-Computer Interaction Institute, Carnegie Mellon University.

OASIS. (2015). *Message Queuing Telemetry Transport (MQTT)* . Burlington: OASIS.

R. Fielding, E. A. (2014). *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content.* Internet Engineering Task Force.

Z. Shelby of ARM, K. H. (2014). *The Constrained Application Protocol (CoAP).* Internet Engineering Task Force.

